Automated Lung Cancer Diagnosis Using Artificial Intelligence

A Thesis submitted to Gujarat Technological University
for the Award of

Doctor of Philosophy
in

Electronics and Communication Engineering
by

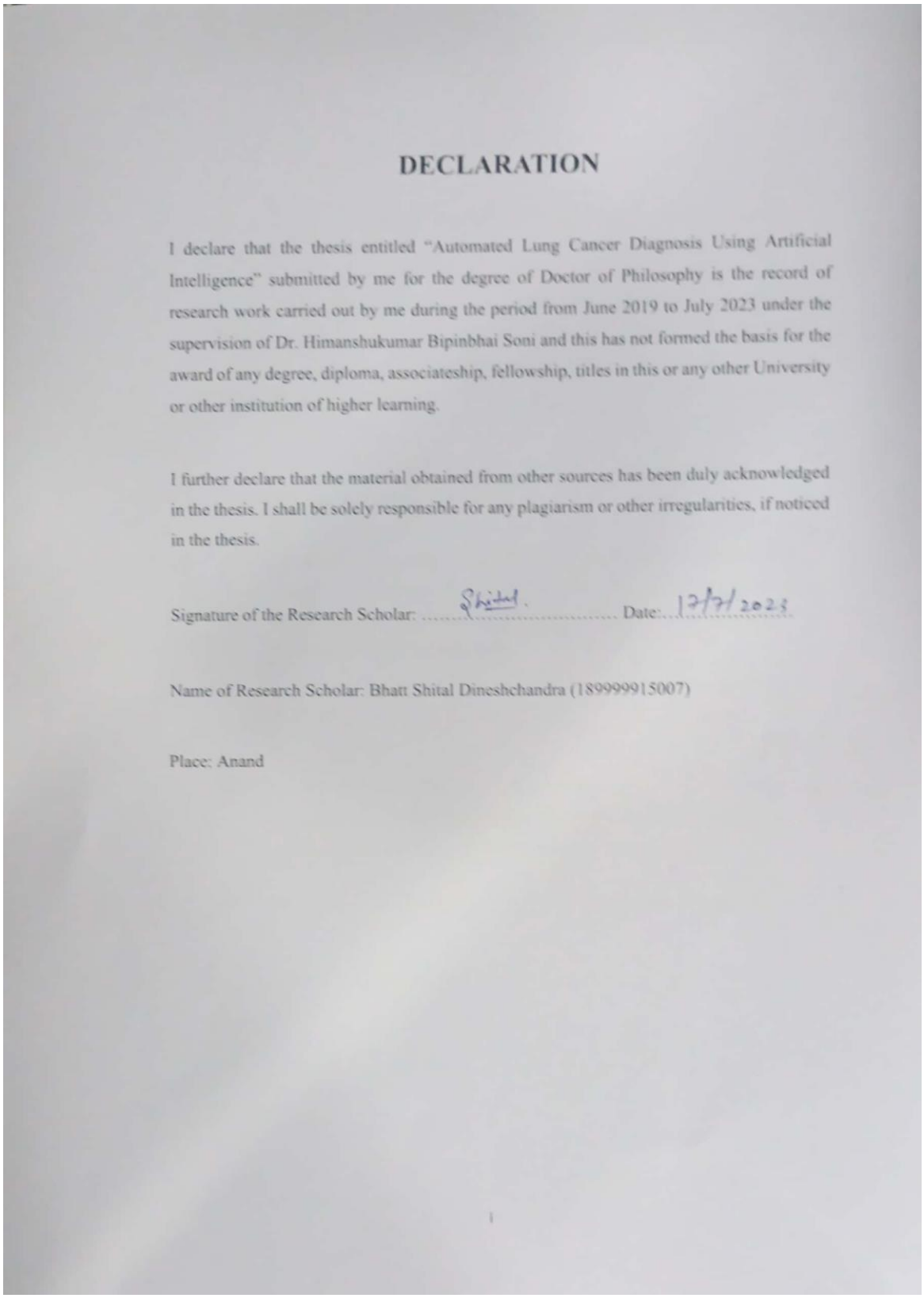
Bhatt Shital Dineshchandra
[Enrollment No.: 189999915007]

under supervision of

Dr. Himanshukumar Bipinbhai Soni

GUJARAT TECHNOLOGICAL UNIVERSITY
AHMEDABAD
[July - 2023]

DECLARATION



DECLARATION

I declare that the thesis entitled "Automated Lung Cancer Diagnosis Using Artificial Intelligence" submitted by me for the degree of Doctor of Philosophy is the record of research work carried out by me during the period from June 2019 to July 2023 under the supervision of Dr. Himanshukumar Bipinbhai Soni and this has not formed the basis for the award of any degree, diploma, associateship, fellowship, titles in this or any other University or other institution of higher learning.

I further declare that the material obtained from other sources has been duly acknowledged in the thesis. I shall be solely responsible for any plagiarism or other irregularities, if noticed in the thesis.

Signature of the Research Scholar: Shital. Date: 17/7/2023

Name of Research Scholar: Bhatt Shital Dineshchandra (189999915007)

Place: Anand

CERTIFICATE

CERTIFICATE

I certify that the work incorporated in the thesis "Automated Lung Cancer Diagnosis Using Artificial Intelligence" submitted by Ms. Bhatt Shital Dineshchandra (189999915007) was carried out by the candidate under my supervision/guidance. To the best of my knowledge: (i) the candidate has not submitted the same research work to any other institution for any degree/diploma, Associateship, Fellowship or other similar titles (ii) the thesis submitted is a record of original research work done by the Research Scholar during the period of study under my supervision and (iii) the thesis represents independent research work on the part of the Research Scholar.

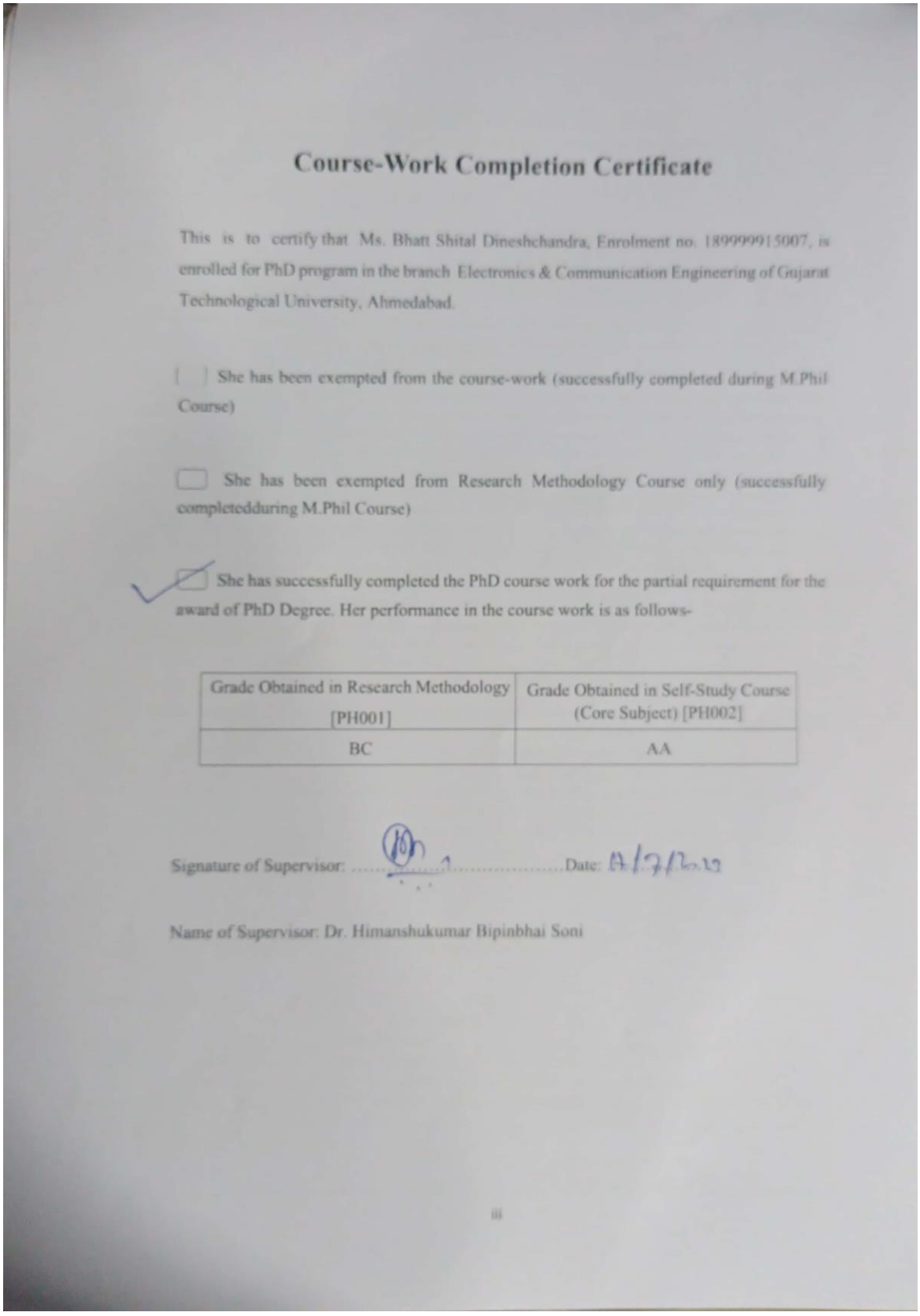
Signature of Supervisor: .....Date: 17/7/2023

Name of Supervisor: Dr. Himanshukumar Bipinbhai Soni

Place: Anand

ii

Course-Work Completion Certificate



Course-Work Completion Certificate

This is to certify that Ms. Bhatt Shital Dineshchandra, Enrolment no. 189999915007, is enrolled for PhD program in the branch Electronics & Communication Engineering of Gujarat Technological University, Ahmedabad.

[ ] She has been exempted from the course-work (successfully completed during M.Phil Course)

☐ She has been exempted from Research Methodology Course only (successfully completedduring M.Phil Course)

☑ She has successfully completed the PhD course work for the partial requirement for the award of PhD Degree. Her performance in the course work is as follows-

Grade Obtained in Research Methodology [PH001]	Grade Obtained in Self-Study Course (Core Subject) [PH002]
BC	AA

Signature of Supervisor: ..... Date: 4/7/2022

Name of Supervisor: Dr. Himanshukumar Bipinbhai Soni

iii

Originality Report Certificate

Originality Report Certificate

It is certified that PhD Thesis titled "Automated Lung Cancer Diagnosis Using Artificial Intelligence" by Bhatt Shital Dineshchandra (189999915007) has been examined by us. We undertake the following:

a. Thesis has significant new work / knowledge as compared already published or are under consideration to be published elsewhere. No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless itis placed under quotation marks and duly referenced.
b. The work presented is original and own work of the author (i.e. there is no plagiarism). No ideas, processes, results or words of others have been presented as Author own work.
c. There is no fabrication of data or results which have been compiled / analyzed.
d. There is no falsification by manipulating research materials, equipment or processes,or changing or omitting data or results such that the research is not accurately represented in the research record.
e. The thesis has been checked using < **URKUND plagiarism checksoftware** > (copy of originality report attached) and found within limits as per GTU Plagiarism Policy and instructions issued from time to time (i.e. permitted similarityindex <=10%).

Signature of the Research Scholar: ..... Date: 17/7/23

Name of Research Scholar: Bhatt Shital Dineshchandra

Signature of Supervisor: ..... Date: 17/7/2023

Name of Supervisor: Dr. Himanshukumar Bipinbhai Soni

Place: Anand

iv

Copy of Originality Report

Document Information

Analyzed document	189999915007_Bhatt_Shital_Dineshchandra_PhD Thesis.pdf (D172044097)
Submitted	7/13/2023 10:33:00 AM
Submitted by	SHITAL BHATT
Submitter email	bhatt9182@gmail.com
Similarity	3%
Analysis address	bhatt9182.gtuni@analysis.urkund.com

Sources included in the report

SA	**Final thesis 26-3-22 without ref.pdf** Document Final thesis 26-3-22 without ref.pdf (D131892050)	4
SA	**Rajasenbagam Final Thesis-1424469834-Rev1.pdf** Document Rajasenbagam Final Thesis-1424469834-Rev1.pdf (D123811484)	2
SA	**Lung Cancer_LR.docx** Document Lung Cancer_LR.docx (D152740165)	2
SA	**Lung Cancer Prediction using DL.docx** Document Lung Cancer Prediction using DL.docx (D163986148)	2
W	URL: https://www.researchgate.net/publication/363120588_The_Improved_Framework_for_Traffic_Sign_Rec... Fetched: 12/19/2022 5:46:39 PM	1
SA	**1424469834-TS.pdf** Document 1424469834-TS.pdf (D132003854)	8
SA	**final_report_lungcancer.doc** Document final_report_lungcancer.doc (D169075688)	10
SA	**AI_Lung_Cancer_Diagnostics___Thesis (1).pdf** Document AI_Lung_Cancer_Diagnostics___Thesis (1).pdf (D112039065)	10
SA	**usma9Jan2018.pdf** Document usma9Jan2018.pdf (D35469310)	4
SA	**Deep_learning_in_CV (1).pdf** Document Deep_learning_in_CV (1).pdf (D144130164)	12
SA	**applsci-1997957-peer-review-v1 (1).pdf** Document applsci-1997957-peer-review-v1 (1).pdf (D146700125)	5

Ph.D. Thesis Non-Exclusive License to GUJARAT TECHNOLOGICAL UNIVERSITY

In consideration of being a Research Scholar at Gujarat Technological University and in the interests of the facilitation of research at the University and elsewhere, I, Bhatt Shital Dineshchandra having Enrollment No. 189999915007 hereby grant a non-exclusive, royalty free and perpetual license to the University on the following terms:

a. The University is permitted to archive, reproduce and distribute my thesis, in whole orin part and/or my abstract, in whole or in part (referred to collectively as the "Work")anywhere in the world, for non-commercial purposes, in all forms of media;
b. The University is permitted to authorize, sub-lease, sub-contract or procure any of theacts mentioned in paragraph (a);
c. The University is authorized to submit the Work at any National / International Library,under the authority of their "Thesis Non-Exclusive License";
d. The Universal Copyright Notice (©) shall appear on all copies made under the authorityof this license;
e. I undertake to submit my thesis, through my University, to any Library and Archives. Any abstract submitted with the thesis will be considered to form part of the thesis.
f. I represent that my thesis is my original work, does not infringe any rights of others, including privacy rights and that I have the right to make the grant conferred by this non-exclusive license.
g. If third party copyrighted material was included in my thesis for which, under the terms of the Copyright Act, written permission from the copyright owners is required, I haveobtained such permission from the copyright owners to do the acts mentioned in paragraph (a) above for the full term of copyright protection.
h. I understand that the responsibility for the matter as mentioned in the paragraph (g) rests with the authors / me solely. In no case shall GTU have any liability for any acts / omissions / errors / copyright infringement from the publication of the said thesis or otherwise.

i. I retain copyright ownership and moral rights in my thesis and may deal with the copyright in my thesis, in any way consistent with rights granted by me to my University in this non-exclusive license.

j. GTU logo shall not be used /printed in the book (in any manner whatsoever) being published or any promotional or marketing materials or any such similar documents.

k. The following statement shall be included appropriately and displayed prominently inthe book or any material being published anywhere: "The content of the published work is part of the thesis submitted in partial fulfilment for the award of the degree of Ph.D.in Electronics & Communication Engineering of the Gujarat Technological University".

l. I further promise to inform any person to whom I may here after assign or license my copyright in my thesis of the rights granted by me to my University in this non-exclusivelicense. I shall keep GTU indemnified from any and all claims from the Publisher(s) orany third parties at all times resulting or arising from the publishing or use or intendeduse of the book / such similar document or its contents.

m. I am aware of and agree to accept the conditions and regulations of Ph.D. including allpolicy matters related to authorship and plagiarism.

Date: 17/7/2021

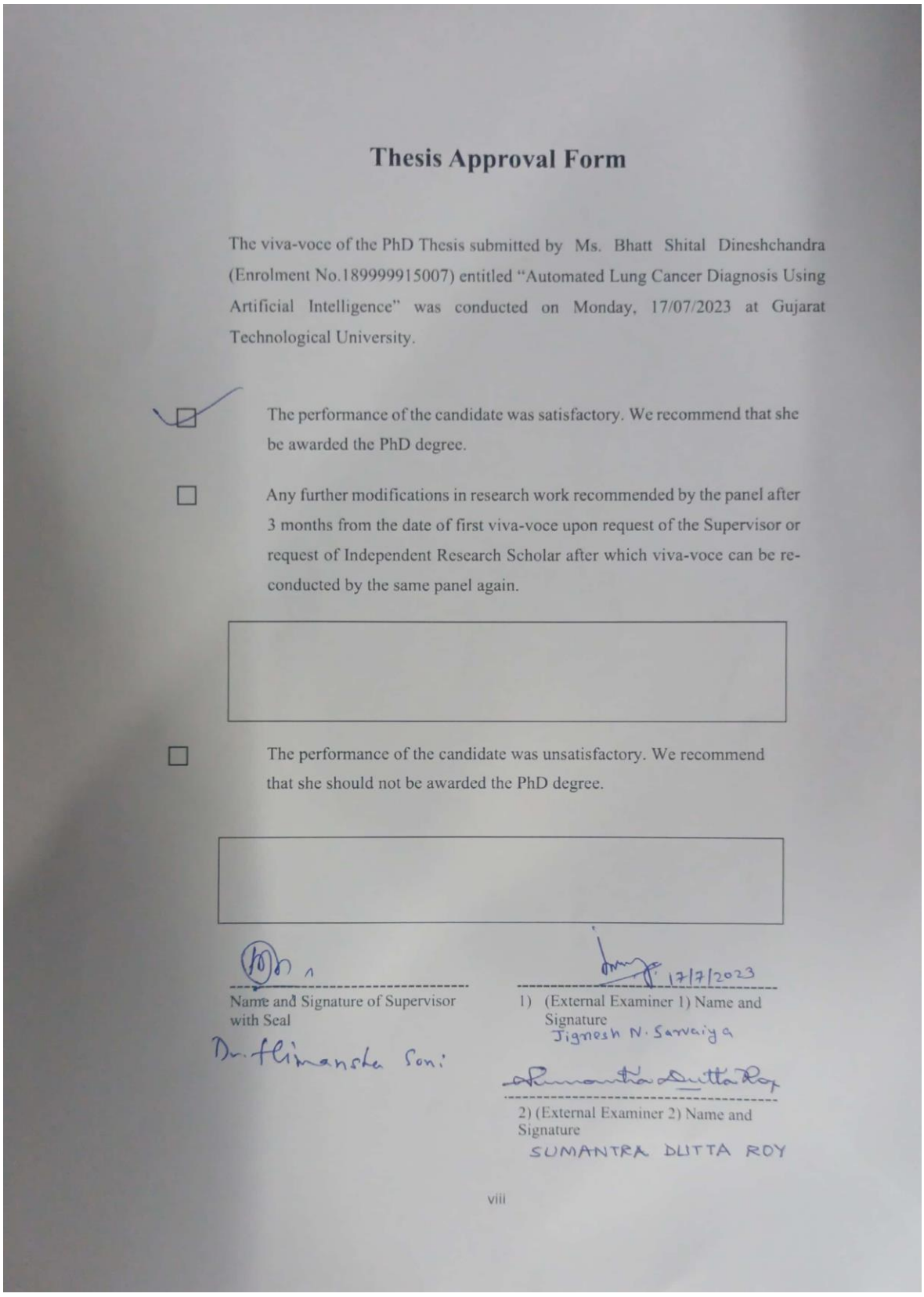
Place: Anand

Signature of the Research Scholar

Recommendation of the Supervisor: Recommended

Signature of Supervisor:

Thesis Approval Form



Thesis Approval Form

The viva-voce of the PhD Thesis submitted by Ms. Bhatt Shital Dineshchandra (Enrolment No.189999915007) entitled "Automated Lung Cancer Diagnosis Using Artificial Intelligence" was conducted on Monday, 17/07/2023 at Gujarat Technological University.

☑ The performance of the candidate was satisfactory. We recommend that she be awarded the PhD degree.

☐ Any further modifications in research work recommended by the panel after 3 months from the date of first viva-voce upon request of the Supervisor or request of Independent Research Scholar after which viva-voce can be re-conducted by the same panel again.

☐ The performance of the candidate was unsatisfactory. We recommend that she should not be awarded the PhD degree.

Name and Signature of Supervisor with Seal

Dr. Himanshu Soni

1) (External Examiner 1) Name and Signature

17/7/2023

Jignesh N. Sarvaiya

2) (External Examiner 2) Name and Signature

SUMANTRA DUTTA ROY

viii

ABSTRACT

Medical images are regarded as a rich source of data which could be used in ingenious ways to discover new knowledge. Using this knowledge there has been advancements in the field of medical imaging technologies which offers a non-invasive method for diagnosing various diseases and suggests effective treatments for them.

This work is about developing a Computer-Aided Diagnosis (CAD) system for lung cancer diagnosis, which is the second most frequent cancer in males after prostate cancer and the second-most common disease in women after breast cancer. The proportion of lung cancer patients has recently climbed substantially over the globe. However, early diagnosis doubles a patient's chances of survival. A pulmonary nodule is the first sign of lung cancer. Lung nodules may be benign or malignant. Lung nodules are roughly spherical areas of predominantly high-density tissue evident in Computed Tomography (CT) images of the lung. Traditional CT scanning techniques can readily identify large (usually considered as bigger than 2 cm in diameter) malignant nodules. Detecting pulmonary nodules on a Low Dose Computed Tomography (LDCT) scan may help to identify early symptoms of lung cancer. Clinical studies have demonstrated that the LDCT reduces lung cancer mortality by 20% when compared to plain chest radiography. Radiologists classify these nodules as benign or malignant based on features such as shape, size, margin, texture, Hounsfield Unit (HU) and so on. However, existing LDCT screening programmes have obstacles such as missed cases, high over-diagnosis rates, high costs and higher radiation exposure. Such missed cases and over-diagnosis may have major consequences and raise the mortality rate. New technologies, notably artificial intelligence, have recently sparked considerable interest in the medical industry due to their ability to automate or provide new information to medical professionals. Thus, designing CAD systems that may be used in clinical practise and assist radiologists in making diagnoses and spotting possible anomalies is crucial.

As a result, the purpose of this study is to help radiologist by developing an automated lung cancer diagnosis model using machine learning and deep learning frameworks.

In this work, Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI) dataset is utilized for the Content Based Image Retrieval (CBIR) approach. This approach combines the Speeded-Up Robust Features (SURF) algorithm for image feature extraction and the k-means clustering algorithm for feature classification. This method achieved an accuracy of 98.56%. In another technique, a subset of the LIDC-IDRI

dataset, LUng Nodule Analysis 2016 (LUNA16), is utilized to train a hybrid model that combines Residual Network-50 (ResNet-50) for feature extraction and Support Vector Machine (SVM) for classification. This hybrid model has an accuracy of 97.53%. Both these classification approaches have been implemented in MATLAB 2020a.

The two approaches described above include two steps: feature extraction and classification. Thus, in order to have a single step end-to-end models for lung cancer classification, eight distinct Deep Neural Networks (DNNs) are trained namely MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, Visual Geometry Group-16 (VGG-16) and VGG-19 and a new model is proposed for better performance. This proposed model is known as Simplified VGG (SVGG) and it has an accuracy of 96.26%. Above eight DNNs are trained on the LUNA16 and Data Science Bowl 2017 (DSB2017) datasets. These classifiers are implemented using python programming and the models are designed using Keras libraries.

The aforementioned three approaches, i.e., CBIR, hybrid approach and DNNs are used to identify the category of the greatest component in an image using a classification strategy. While lung cancer diagnosis involves a detection strategy to determine the specific location of the abnormality and its likelihood that it is cancerous or non-cancerous. Five object detectors are trained for nodule detection in this study are Faster Region-based Convolutional Neural Networks (Faster R-CNN), Single Shot Detection (SSD), You Only Look Once version 3 (YOLOv3), YOLOv4 and YOLOv5. The datasets used to train these nodule detectors are Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI), LUNA16, DSB2017, National Lung Screening Trial (NLST), Non-Small Cell Lung Cancer (NSCLC)-Radiogenomics, NSCLC-Radiomics, Lung CT Segmentation Challenge (LCTSC), Quantitative Imaging Network (QIN), The Cancer Genome Atlas Lung Squamous Cell Carcinoma (TCGA-LUSC), TCGA-Lung Adenocarcinoma (TCGA-LUAD), 4D-Lung, LungCT-Diagnosis, The Reference Image Database to Evaluate Response (RIDER), European Collaboration on Low-Dose Aspirin in Pblycythemia (ECLAP), Anti-Programmed cell death protein 1 (Anti PD-1). All the datasets are open access except NLST which is received through the grant from National Institutes of Health (NIH).

Among the above mentioned five nodule detectors, YOLOv5 outperformed and it achieved precision of 90%, recall of 80%, mean Average Precision (mAP) of 85%, F1 Score of 0.85 and Giga Floating-Point Operations (GFLOPs) of 217. Thus, it faced few challenges like complexity, have high computation time and low performance. So, to overcome these

problems, two automated lung cancer diagnosis models have been proposed. The first proposed model which is a modification of YOLOv5 where few layers are replaced by the Focus and Cross Stage Partial (CSP) layers. This model achieved 85% precision, 82% recall, 85% mAP, F1 score of 0.84 and GFLOPs 35.2. In order to reduce the computation time of Proposed Model-1, the second model is proposed which is a modification of first one whose backbone convolution layers are replaced by Depth Wise Separable Convolution (DWSC). This Proposed Model-2 outperformed the first one, with precision of 95%, recall of 90%, mAP of 95%, F1 score of 0.92 and 30.6 GFLOPs. All above mentioned nodule detection models are trained using above mentioned 14 lung CT datasets and implemented using python programming and the detectors are designed using TensorFlow library.

Among all the seven trained nodule detectors, the performance of the Proposed Model-2 is better and so it is used for validation. For validation of the Proposed Model-2, 100 NLST CT scans are used. The validation datasets consist of equal amount i.e., 50 for cancer and 50 for non-cancer CT scans. The prediction is compared with the ground truth nodule location, margin and size provided in annotation file and it was found that 90% for non-cancer and 92% for cancer predictions are correct as per actual label.

Finally, a Graphical User Interface (GUI) is built to make the nodule detection more user-friendly so that radiologists can use it for clinical diagnosis.

Dedication

This thesis is dedicated to "Bhagwan Swaminarayan and Pramukh Swami Maharaj"

Acknowledgement

Firstly, I would like to thank whole heartedly and devotionally Almighty God for providing me the opportunity and ability to complete this research work. The journey of doctoral work cannot be possible without moral and mental support. There are several people to whom I would like to dedicate my doctoral research work and express my sincere gratitude to them.

At the beginning, I would like to convey my sincere thanks to my research supervisor, Dr. Himanshukumar Bipinbhai Soni, Provost, Charutar Vidya Mandal University (CVMU) who have always been instrumental and motivational throughout my research work. His astounding direction and noteworthy contribution in ensuring quality work is highly appreciable. He has always shown readiness to extend support as and when required and whenever I stuck up.

I would never have possessed the capacity to complete the thesis without the directions shown by Doctoral Progress Committee members Dr. Tanmay Pawar, Professor and Head, Electronics Engineering Department, BVM Engineering College and Dr. Heena Rahul Kher, Former Assistant Professor, Electronics and Communication Engineering Department, A. D. Patel Institute of Technology. Remarkable suggestions and comments given by them helped me a lot to improve my research work to a great extent.

I am grateful to National Cancer Institute for allowing me to access of NCI's data collected by National Lung Screening Trial (NLST).

I wish to express my sincere gratitude and humble thanks to Dr. Deepak Mehta, Professor and Head, Department of Radio-diagnosis, Pramukhswami Medical College, Shree Krishna Hospital, Karamsad, Dr. Drushi Patel, Senior Consultant Radiologist, Gujarat Imaging Centre, Ahmedabad, Dr. Rahul Mishra, Oncology, Therapist, HCG Vadodara, Dr. Himani Patel, Radiologist, HCG, Vadodra, Dr. Abhishek Mahajan, Associate Professor, Department of Radio-diagnosis and Imaging, Tata Memorial Centre Mumbai, Dr. Kamlesh Sajnani, Radiologist, USA, Dr. Atit Kantawala, Specialist, Department of Joint Replacement Surgery, Anand Orthopaedic Hospital and Dr. Sonal Patel, Dentist, Snehi Dental Clinic and Implant Center, Anand whose support has been most commendable in the journey of this research.

I am obliged to the GTU Ph.D. section for supporting me in doctoral research work.

I am especially thankful to Er. Bhikhubhai B. Patel, Chairman, Charutar Vidya Mandal and Dr. S. G. Patel, Hon. Secretary, Charutar Vidya Mandal for their blessings. I would also like to express my sincere thanks to Dr. Sudhir Vegad, Principal, M.B. Patel Institute of Technology for his constant support and motivation.

I feel a deep sense of gratitude of my parents, brothers and sister who formed part of my vision and taught me the good things that really matter in life. I am thankful to my in-law parents for their support. At last, I would like to gratefully acknowledge the support and understanding of my husband Mitesh and my daughter Mishti throughout my research work. I am thankful to all who have directly or indirectly given support during my work.

Bhatt Shital Dineshchandra (189999915007)

Table of Contents

List of Figures

List of Tables

CHAPTER-1
Introduction

Each lung, a vital component of the respiratory system, has both conducting and respiratory zone structures. The primary job of the lungs is to execute the process of exchanging oxygen and carbon dioxide with atmospheric air.

The lungs are pyramid-shaped, paired organs linked to the trachea via the right and left bronchi; the lungs are bordered on the inferior surface by the diaphragm. The diaphragm is a flat, dome-shaped muscle found at the base of the lungs and the thoracic cavity. The pleurae, which are linked to the mediastinum, surround the lungs. The right lung is shorter and broader than the left, and it has a lesser volume than the right. The cardiac notch is an indentation on the left lung's surface that gives room for the heart. The upper area of the lung is the apex, whereas the base is the opposite region near the diaphragm. The ribs are bordered by the lung's costal surface. The mediastinal surface is parallel to the midline.

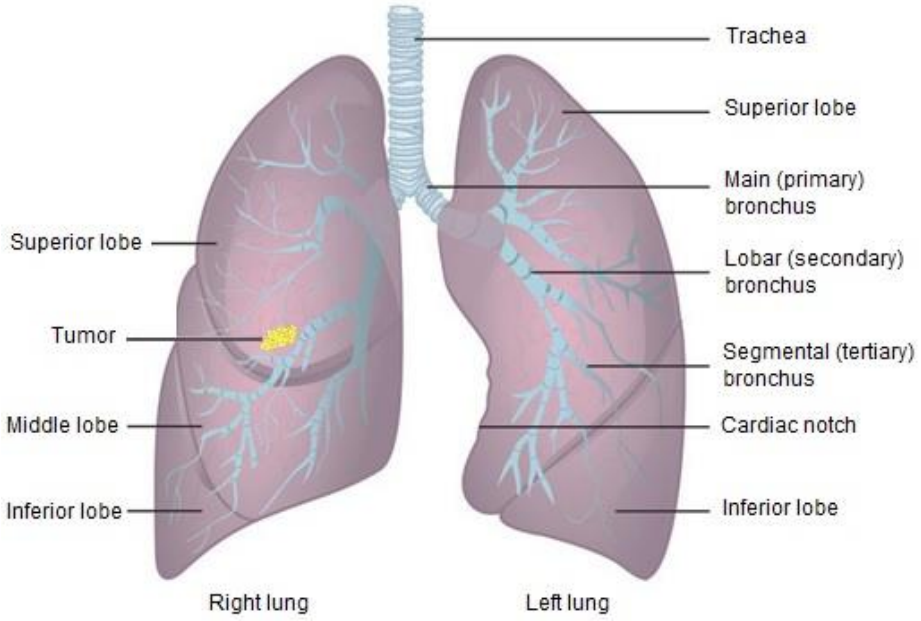


Fig. 1.1 Gross Anatomy of the Lungs [1]

Each lung is made up of smaller components known as lobes. These lobes are separated by fissures. The right lung is divided into three lobes: superior, middle, and inferior. The left lung

is divided into two lobes: superior and inferior. A bronchopulmonary segment is a lobe division, and each lobe has many bronchopulmonary segments. Each segment gets air via its own tertiary bronchus and blood from its own artery. Some lung disorders often affect one or more bronchopulmonary segments, and in some situations, the affected segments may be surgically removed with little impact on neighbouring segments. A pulmonary lobule is generated when the bronchi branch into bronchioles. Each lobule has its own big bronchiole with several branches. An interlobular septum is a connective tissue wall that divides lobules from one another.

Usually, the cells within our lungs as well as other parts of our body undergo a growth and death cycle that maintains a constant cell count. Cancer of any type develops when a series of specific alterations, known as mutations, occur in a cell that was once healthy. When a collection of mutations alters the normal growth and death cycles of cells, uncontrolled cell division can result [2].

The mass of mutated and abnormally expanding cells is known as a tumour, neoplasm, or lesion. On a chest X-ray or Computed Tomography (CT) scan, this mass may appear as a nodule that is malignant (cancer) or benign in the case of lung cancer.

1.1 Background of Lung Cancer Diagnosis

According to a global study, Lung cancer is the most common diagnosed cancer (11.6% of all cancer cases) and the leading cause of cancer mortality (18.4% of all cancer deaths) in both sexes combined [3-6]. In recent decades, lung cancer has been responsible for more fatalities than breast, prostate, colorectal and brain malignancies combined [7]. A total of 228,820 new cases of lung cancer were recorded in 2020 and projected 135,720 deaths in the US alone [8]. There were more than 2.2 million new cases of lung cancer in 2020. This translates to approximately 107,870 smoking-attributable lung cancer deaths in 2021, with an additional 3590 due to second-hand smoke exposure, leaving a residual 20,420 lung cancer deaths [9]. Approximately 2,206,771 new cases of the lung cancer out of which 1,796,144 deaths had been caused in 2022 [10]. The World Health Organization (WHO) research estimates that by 2040, there would be around 36,00,000 new instances of lung cancer globally and there will be 31,00,000 fatalities [11]. With such fatal effects, early detection is extremely important in

ensuring the patient's chance of survival. Early detection of lung cancer is rare, with advanced stage disease (stage III or IV) occurring in 70% of lung cancer patients [12].

The state-of-the art methods for lung cancer detection and diagnosis are largely observation based. One of the primary steps in diagnosis is performing a chest radiograph, also known as a chest X-ray [13, 14]. Difficulties and issues with this approach is that the diagnosis is largely based on the experience and discretion of the individual doctor and is conflicted with the various biases that they've encountered throughout their career. If the patient's condition matches an unseen exception, it has the potential to be missed/under-diagnosed, thereby heavily impacting the patient's chance of survival. It is common for two doctors to differently diagnose the exact same condition. It's this variability that heavily hinders the existing method of detection. This utilizes the transmission of radiation through the patient's body to obtain a 2-D snapshot of their anatomy. Early-stage lung cancer also can be viewed as visible pulmonary nodules in CT scans. Such lesions can be detected despite being just a few millimetres in diameter on CT scans [15].

A more recent development in this field is Low-Dose helical Computed Tomography (LDCT). It has been demonstrated that LDCT can detect smaller lung nodules [16]. This technology is fairly new and unproven. Its primary advantage is that it is able to detect much smaller tumours (the size of a grain of sand), thereby leading to earlier detection and higher survival rates. This comes at the cost of higher false positive rates, which can lead to unnecessary invasive procedures and therefore, serious clinical complications. Studies also found that radiologists are still unable to recognize all the nodules seen in CT scans. This led to the development of Computer-Aided Diagnosis (CAD) systems [17, 18].

At the beginning in 1950s, there was a thought that the CAD systems may take place of the radiologists in determining anomalous tissues, since computers perform given tasks much better than human. However, this thought could not find ground because of the ineligible computer sources and unsophisticated image-processing methods and techniques. In 1980s, systematic and large-scale development and research about various CAD systems was started. The notion of taking computer outputs as a second opinion rather than taking place of radiologists gain more support, thus there was development in CAD system. It took more than 30 years for the researches to become solid and in 1998, Food and Drug Administration (FDA) approved ImageCheckerTM which was the first approved CAD system in mammography [19].

Although, the performance of CAD systems is not superior to the performance of physicians, many clinics in the Europe and USA benefit the aid of CAD systems in the early detection of breast cancer as a second opinion [20]. Now, CAD has become part of the common clinical procedure for breast cancer detection in some clinics, thus CAD turns out to be the leading research area in machine learning for medical diagnosis and imaging [21].

It is well known that anatomic structure of lungs makes nodule detection very hard mission thus radiologists may miss about 35% of the nodules on chest images [22]. Several studies and findings on CAD systems point out that by using CAD systems radiologists can get better results in nodule detection. CAD does not only improve the performance and efficiency of the radiologists, but also improves the precision and accuracy of nodule detection rate as well [19]. Therefore, guidance and aid in nodule detection is the main objective of using CAD systems, which helps to increase the consistency and accuracy of radiologist's diagnosis. Whereas, CAD systems have still more false-positive thus sensitivity of nodule detection is low compared to experienced radiologists [23]. It is understood that CAD systems advances the performance of radiologists in diagnosing process and nodule detection, however it does not become common in clinical usage. Regular usage in the clinics depends on meeting the following requirements: ensure in high sensitivity, decrease in the number of false positives, increase in the diagnosis speed, advance in automation, decrease in running cost, advance in the capability of detection of various shapes and types of nodules [24].

Some challenges for CAD system researchers and developers in the field of lung cancer diagnosis are as follows [25]:

i. The lack of homogeneity in the lung structure and affinity densities in veins, arteries, bronchi and bronchioles makes segmentation a very hard issue.
ii. The level of automation, the quickness of the system, the success rate of varied shape and size of nodule detection.
iii. For efficient validation of the CAD systems larger databases should be provided.
iv. There is a need for more advanced techniques or improvement of the present techniques in segmentation of lungs in order to detect nodules $\leq 3\ mm$.
v. Moreover, the other challenges faced by CAD system are insufficiency of well labelled medical datasets, generalizability issues and poor interpretability of the detection results.

vi. In addition to above challenges, the existing CAD systems are complex which leads convergence issue and have high computation time.

vii. Still the performance of CAD system is low because the pulmonary nodules have large variations in sizes, shapes and locations. Moreover, the contextual environments around them are often diversified for different categories of lung nodules. Thus, the designers of CAD systems have to compromise on either high sensitivity or high FP rate.

1.2 Motivations of Work

Lung cancer is the most prevalent cancer diagnosed (11.6% of all cancer cases) and the major cause of cancer mortality (18.4% of all cancer deaths) in both sexes combined, according to a worldwide survey. As a result, early detection is critical in ensuring the patient's chance of life. The anatomical structure of the lungs is widely recognized to make nodule identification problematic for radiologists. As a result, around 35% of nodules seen on chest imaging may be ignored. Hence, many CAD systems have already been established to assist radiologists with nodule diagnosis. Both machine learning and deep learning approaches are used in CAD. Complex machine learning approaches include separating the lung region in the CT image, selecting nodules, identifying distinguishing characteristics and reducing FPs during categorization. A basic machine learning approach using Bag-of-Features (BOF) is used for implementing the Content Based Image Retrieval (CBIR) approach was carried out in this work. Bhatt et al. suggested a CBIR technique using Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI) dataset for lung cancer classification that used the Speeded-up Robust Features (SURF) and K-means algorithms for feature extraction and classification. The approach yielded training and testing accuracy of 99% and 98.56%, respectively.

Machine learning approaches need a significant amount of time to create, execute and optimize hyperparameters. Deep learning may remove these subcomponents, enabling the system to extract insights from the input. To develop end-to-end models for lung cancer classification, seven separate Deep Neural Networks (DNNs) are trained: MobileNet, SqueezeNet, AlexNet, GoogleNet, Residual Network18 (ResNet-18), Visual Geometry Group (VGG)16 and VGG-19. Bhatt and Soni suggested a Simplified VGG (SVGG) model for lung cancer classification that was built on the VGG-16 framework and attained an accuracy of

99.60%. Bhatt et al. suggested a hybrid technique that employs deep neural networks for feature extraction and machine learning for feature classification. The first step involves extracting nodules from the LUng Nodule Analysis 2016 (LUNA16) dataset using CT scans. Instead of using complete images, patches were used for feature extraction and categorization. To evaluate the LUNA16 in this way, ResNet-50 was employed as a feature extractor. Support Vector Machines (SVM) was used to classify the deep traits. The approach attained an accuracy of 97.53%.

CBIR, hybrid approaches and DNNs are classification algorithms used to find the category of the biggest component in an image. While the goal of the study was to diagnose lung cancer, it was also required to locate their specific position on the lung image. Faster R-CNN, Single Shot Detector (SSD), You Only Look Once, version 3 (YOLOv3), YOLOv4 and YOLOv5 are five object detectors trained for nodule detection in this research. In this work, YOLOv5 surpassed the five aforementioned nodule detectors, achieving 90% precision, 80% recall, 85% mAP, 0.85 F1 Score and 217 Giga Floating-Point Operations (GFLOPs). However, it encountered a few obstacles, such as complexity, excessive calculation time and poor performance.

Consequently, two automated lung cancer diagnostic models have been developed to address these issues. A few YOLOv5 layers are substituted with Focus and CSP layers in the initial suggested model. This model attained 85% precision, 82% recall, 85% mAP, an F1 score of 0.84 and 35.2 GFLOPs. In order to lower the calculation time of Proposed Model-1, we suggest the second model, which is a variation of the previous one in which the backbone convolution layers are replaced with DepthWise Separable Convolutions (DWSC). With a precision of 95%, a recall of 90%, a mAP of 95%, F1 score of 0.92 and 30.6 GFLOPs, Proposed Model-2 outperforms all the implemented nodule detectors in this work.

Previous research demonstrates the need of using large medical datasets for diagnosis. So, in this study, 15 distinct datasets were utilized to train the various models are Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI), LUng Nodule Analysis 2016 (LUNA16), Data Science Bowl 2017 (DSB2017), National Lung Screening Trial (NLST), Non-Small Cell Lung Cancer (NSCLC)-Radiogenomics, NSCLC-Radiomics, Lung CT Segmentation Challenge (LCTSC), Quantitative Imaging Network (QIN), The Cancer Genome Atlas Lung Squamous Cell Carcinoma (TCGA-LUSC), TCGA-Lung Adenocarcinoma (TCGA-

LUAD), 4D-Lung, LungCT-Diagnosis, Reference Image Database to Evaluate Response (RIDER), European Collaboration on Low-Dose Aspirin in Pblycythemia (ECLAP), Anti-Programmed cell death protein 1 (Anti- PD 1) Lung. Except for NLST, which is funded by a grant from the National Institutes of Health (NIH), all of the datasets are freely accessible.

A unique user interface Lung nodule diagnosis Graphical User Interface (GUI) is designed for the nodule detectors in this study to identify lung cancer, making the procedure more user-friendly.

1.3 Objectives of Thesis

The main objective of this thesis are as follows:

1) Acquisition of the knowledge of various machine learning and deep learning frameworks.
2) The evaluation of different methods for classification of lung cancer using machine learning and deep learning approaches and thereby implementing better approach for classification.
3) Development of an algorithm for feature extraction and matching methods to detect nodules on NLST datasets.
4) Evaluation and analysis of various nodule detection approaches by training on different datasets and thereby proposing models for automatic diagnosis of pulmonary nodule that will have accurate detection and will require less time for diagnosis.

1.4 Original Contributions of Thesis

Original contribution by this thesis are as follows:

1) Machine learning approach for lung cancer classification using CBIR method.
2) Hybrid model consisting of ResNet-50 and SVM for lung nodule classification.
3) Training and evaluating various Deep Neural Networks (DNNs) namely MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and VGG-19 for lung cancer classification. Furthermore, improvement of the performance for lung cancer classification by a proposed SVGG network.
4) Training and evaluating object detection models like Faster R-CNN, SSD, YOLOv3,

YOLOv4 and YOLOv5 for diagnosis of pulmonary nodules on CT scans.

5) Design and implementation of two automated CAD models with low complexity and less training time along with better performance in terms of high precision, recall and accuracy.
6) Design and implementation of user-friendly GUI for lung cancer diagnosis using artificial intelligence.

1.5 Datasets

Having established the need for development of the CAD system for the lung cancer diagnosis, it was necessary to determine the sources of data (materials).

The datasets used in this work are LIDC-IDRI/LUNA16, DSB2017, NLST, NSCLC-Radiogenomics, NSCLC-Radiomics, LCTSC, QIN, TCGA-LUSC, TCGA-LUAD, 4D-Lung, LungCT-Diagnosis, RIDER, ECLAP, Anti- PD 1 Lung. All datasets are accessible except NLST. This NLST dataset is obtained via a grant from the NIH.

1.5.1 Lung Image Database Consortium and Image Database Resource Initiative/ LUng Nodule Analysis 2016

The acronyms LIDC-IDRI stand for Lung Image Database Consortium and Image Database Resource Initiative [26]. It is a thoracic CT scan database that owes its success to three research organisations: the National Cancer Institute (NCI), the Foundation for the National Institutes of Health (FNIH) and the Food and Drug Administration (FDA). The LIDC-IDRI dataset has a total of 1,018 cases. In reality, there are only 1,010 different CT scans since eight cases were mistakenly repeated when obtaining the CT scans.

All image data collected are saved in DICOM format with a consistent size of 512×512. The image thickness spans from 0.5 to 5 mm, with 1 mm, 1.25 mm and 2.5 mm being the most common. Each case in the LIDC-IDRI collection consists of hundreds of images as well as an XML file with information about the diagnosed lung lesions. Each identified lung lesion is divided into three primary categories based on its diameter: nodules (3-30 mm of diameter), non-nodules (diameter $\geq$ 3 mm) and micro-nodules (diameter < 3 mm).

The LUng Nodule Analysis 2016 (LUNA16) dataset is a portion of the LIDC-IDRI dataset that is available to the public. It comprises of a total of 888 thoracic CT scans gathered on the assumption that at least three out of the four medical professionals who participated in the annotation procedures noted the lesions in each instance [27-29].

1.5.2 Data Science Bowl 2017

The Data Science Bowl 2017 (DSB2017) dataset was created from 2101 patients, with each patient's file including around 100 to 400 images [30]. The following annotations have been applied to each image: Those without cancer were given label 0 while patients with cancer were assigned label 1.

1.5.3 National Lung Screening Trial

It is a dataset gathered in 2009 by the National Lung Screening Trial (NLST) [31], a study that attempted to compare the accuracy of lung cancer detection by LDCT screening to that of chest radiography screening. The screening experiment was carried out in 33 American medical facilities, with a total of 53,454 individuals. These participants ranged in age from 55 to 74 years old and had a smoking history.

1.5.4 Non-Small Cell Lung Cancer-Radiogenomics

The Non-Small Cell Lung Cancer (NSCLC)-Radiogenomics is a one-of-a-kind dataset derived from a 211 person’s cohort [32]. The collection includes CT images, Positron Emission Tomography (PET), semantic annotations of tumours noticed on medical images. This dataset was intended to help researchers understand the underlying association between genetic and medical image traits as well as to develop and test predictive medical image biomarkers.

1.5.5 Non-Small Cell Lung Cancer-Radiomics

This collection includes images from 422 scans with Non-Small Cell Lung Cancer (NSCLC) [33]. Manual delineation of the 3D volume of the gross tumour volume by a radiation oncologist and clinical outcome data are available for these patient’s pre-treatment CT images.

1.5.6 Lung CT Segmentation Challenge

The Lung CT Segmentation Challenge (LCTSC) data collection was made available in conjunction with a challenge competition and accompanying conference session at the AAPM 2017 annual meeting [34]. The dataset consists of 60 CT scans.

1.5.7 Quantitative Imaging Network

The H. Lee Moffitt Cancer Center and Research Institute provided CT image records of patients detected with Non-Small Cell Lung Cancer (NSCLC) having mixed stage and histology. Scans were gathered from individuals who had pre-surgery diagnostic CTs and had surgical resection. The dataset consists of 47 CT scans.

The scans were de-identified following Health Insurance Portability and Accountability Act (HIPPA) guidelines to protect patient privacy. The data was shared with the Quantitative Imaging Network **(**QIN) [35] collaborators for research purpose complying with collaborative data sharing policy. The mission of the QIN is to improve the role of quantitative imaging for clinical decision making in oncology by developing and validating data acquisition, analysis methods and tools to tailor treatment for individual patients and predict or monitor the response to drug or radiation therapy.

1.5.8 The Cancer Genome Atlas Lung Squamous Cell Carcinoma

The collection of data from The Cancer Genome Atlas Lung Squamous Cell Carcinoma (TCGA-LUSC) [36] is part of a wider effort to develop a research community focused on linking cancer phenotypes to genetics by providing clinical images matched to participants from The Cancer Genome Atlas (TCGA). The Genomic Data Commons (GDC) data portal has clinical, genomic and pathological data, while The Cancer Imaging Archive (TCIA) includes radiological data. The dataset consists of 37 CT scans.

Matched TCGA patient identifiers allow researchers to explore the TCGA-TCIA databases for correlations between tissue genotype, radiological phenotype and patient outcomes.

1.5.9 The Cancer Genome Atlas Lung Adenocarcinoma

The Cancer Genome Atlas Lung Adenocarcinoma (TCGA-LUAD) data collection [37] is part of a larger effort to build a research community focused on connecting cancer phenotypes to genotypes by providing clinical images matched to subjects from TCGA. Clinical, genetic and pathological data resides in the GDC data portal while the radiological data is stored on TCIA. The dataset consists of 69 CT scans.

1.5.10 4D-Lung

The 4D-Lung dataset [38] comprises of images taken during chemoradiotherapy on 20 patients with locally advanced NSCLC.

1.5.11 LungCT-Diagnosis

All of the images are contrast enhanced CT scans for diagnostic purposes. To ensure adequate patient follow-up, the images were obtained retrospectively [39]. The thickness of the slices varies from 3 to 6 mm. All images were taken at the time of diagnosis and before surgery. The study's goal was to identify predictive image elements that define lung adenocarcinomas and are associated with overall survival. The dataset consists of 61 CT scans.

1.5.12 Reference Image Database to Evaluate Response

The Reference Image Database to Evaluate Response (RIDER) [40] was created as part of a research to assess the variability of tumour unidimensional, bidimensional and volumetric measures on the same-day repeat CT scans in non-small cell lung cancer patients.

This research comprised 32 patients with NSCLC who underwent two CT scans of the chest in 15 minutes using the same imaging methodology. Three radiologists independently estimated the two largest diameters of each lesion on both scans and measured the same tumours on the first scan during another session. In a second investigation, computer software was used to calculate the two largest diameters and volume of each lesion on both scans.

1.5.13 European Collaboration on Low-Dose Aspirin in Pblycythemia

The European Collaboration on Low-Dose Aspirin in Pblycythemia (ECLAP) is an acronym of early lung cancer action program. It is a collection comprising of 50 LDCT instances [41]. The image slice thickness is 1.25 mm and the most of the detected nodules have sizes ranging from 2 to 5 mm. It is worth noting that all of the annotated lesions in this dataset are nodules and non-nodules being recognised by medical specialists.

1.5.14 Anti-Programmed Cell Death Protein 1

The Anti-Programmed cell death protein 1 (Anti-PD 1) [42] comprises 46 lung patients treated with anti-PD1 immunotherapy in 2016, each having pre-treatment and the majority with at least one imaging follow-up timepoint.

1.6 Software and Hardware Used for Implementation

In this thesis, the CBIR approach and the hybrid models were implemented in MATLAB 2020a while the models for classification and nodule detection were implemented using python programming as the primary software language and the PyCharm 2021.1.3 is used as integrated development environment (IDE). The basic pre-processing steps were carried out using Open-Source Computer Vision (OpenCV) Library, Python Imaging Library (PIL) and scikit-learn. At primary stage, Jupyter notebook has used which is a web-based interactive computing platform. The deep learning models for lung cancer classification and nodule detection are developed using Keras and TensorFlow libraries.

Initial part of the work of CBIR approach and the hybrid model was accomplished on a low computing system of 8 GB RAM. While the classifier and nodule detection models were trained using a high processing desktop with following features: 9^{th} generation Intel Core i7-9700K/32GB RAM/2TB HDD + 512GB PCIe SSD/Windows 10/8GB NVIDIA GeForce RTX 2070 Graphics.

1.7 Organization of Thesis

The thesis is organized as follows:

Chapter 2 discussed the work that has been done so far by many researchers in the development of the CAD system for lung cancer detection. It includes a review of the literature on machine learning and deep learning methods for classifying lung cancer. Additionally, the research done employing object detection algorithms for the diagnosis of lung cancer is also covered. The chapter concludes with critical observations made based on literature review.

Chapter 3 includes a simple and effective method for identifying and obtaining similarity in lung CT images from a large medical image collection. Two unique techniques to lung cancer categorization have been proposed. For the first approach, a CBIR-based BOF technique for lung cancer categorization has been presented. As precise lung segmentation is not necessary in this approach which leads to increase in the speed of operation. SURF, K-means and SVM were used to depict the visual patterns of nodules present in LIDC-IDRI dataset. In the second technique, the ResNet-50 is employed as feature extractors to examine patches derived from the LUNA16 dataset. The SVM was used to classify these extensively extracted features.

Chapter 4 contains a superior approach in contrast to Chapter 3, which required separate techniques for feature extraction and classification. In this chapter, DNNs are used to classify lung cancer. Patches taken from the LUNA16 and DSB2017 datasets are used to train seven well-known DNNs, including MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and VGG-19. A new model is developed based on the performance and challenges encountered by these seven models. The VGG approach was utilized to construct this new model, so the proposed model is named as Simplified VGG (SVGG).

Chapter 5 employs several deep neural networks to identify and categorize cancer and non-cancer nodules on lung CT scans. Initially, nodule detection networks such as Faster R-CNN and SSD were trained and their outcomes were compared. According to the findings, the Faster R-CNN and SSD algorithms failed to locate nodules in many samples. As a result, the YOLO algorithms YOLOv3, YOLOv4 and YOLOv5 were trained. These algorithms were chosen for their superior performance in real-time applications. When YOLOv3, YOLOv4 and YOLOv5 are compared in terms of performance for nodule detection, YOLOv5 surpasses YOLOv4 and YOLOv3. However, the YOLOv4 and YOLv5 exhibit unequal precision and recall, as well as a longer computational time, leading to the development of new models.

Chapter 6 describes two automated nodule identification models which are modifications of YOLOv5, resulting in improved performance and efficient prediction on test images. In addition, the suggested model required much less training time than the other implemented nodule detectors. Finally, the highest performing Proposed Model-2 was chosen for validation. On test images, the proposed model performed better and predicted more accurately. Furthermore, the proposed model needed much less training time than the previous implemented nodule detectors. In order to make the framework for nodule detection more user-friendly, a GUI that can be trained and used for lung cancer diagnosis is also developed.

Chapter 7 concludes the work and suggests the future scope of work.

CHAPTER-2
Literature Survey

This chapter explains in detail the numerous lung cancer diagnosis techniques. A concise introduction highlights the fundamental concepts that aided readers comprehend this thesis. The majority of literature reviews for lung cancer diagnosis are based on machine learning and deep learning concepts. The chapter concludes with critical observations based on a review of the relevant literature.

2.1 Machine Learning Approaches for Lung Cancer Classification

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. There are numerous successful machine learning algorithms used for classification of lung cancer. These machine learning methods consist of two steps i.e., feature extraction and their classification.

Traditionally, feature engineering was used to define important features to describe nodules. The quantitative characteristics (features) of the lesion candidates were obtained during the feature extraction process. The classifier used these characteristics (from the training data) as input to find the best boundary in the feature space to distinguish between nodules and non-nodules (or benign and malignant). The quality of the extracted characteristics had a significant impact on how accurate the boundary was. The process of feature extraction generates many different kinds of features, such as morphological, texture, gradient-based and context features [43]. Using the candidate segmentation mask, morphological characteristics measure the candidate's shape-based information. From the voxel intensity values of the CT images, texture and gradient-based features were able to capture the visual patterns and intensity information of the regions within or outside the nodule candidate [44]. The relative location information of the candidates was assessed using context characteristics. Sousa et al [45] presented machine learning approach for lung cancer classification using 33 CT scans data which provided accuracy of 95.21%. Casico et al. [46] proposed a stable 3D mass-spring model trained on 84 LIDC-IDRI datasets and achieved accuracy of 97%. Messay et al. [47] proposed the CAD system based on a fully automated lung segmentation algorithm to define the boundaries of the

lung regions. A sequential forward selection process was used to determine the optimum subset of features for two distinct classifiers, a Fisher Linear Discriminant (FLD) classifier and a quadratic classifier. In this method, 84 LIDC-IDRI datasets were used and the accuracy of 92.8% was obtained.

Classification is the process of categorizing or dividing data instances into various groups. The purpose of the classification is to label similar pixels in an image thus form several classes according to their similarities. Classification which is the significant component of image analysis, forms the classes by analyzing the image features.

2.1.1 Content Based Image Retrieval

Content Based Image Retrieval (CBIR) is used to dynamically index and retrieve images based on the contents of the image known as features. CBIR is characterized as a process for finding a similar image database for a query image [48-50]. The discrepancy between query image and database feature vectors is then calculated and graded. The image database shall be extracted with the maximum resemblance to the sample image. The performance review is then carried out based on accuracy. CBIR can offer CAD support by permitting specialists to identify previously diagnosed cases from a database that are close to the cases they are interpreting [51]. A common technique for applying the CBIR approach is a bag of visual terms, also known as Bag-of-Features (BOF) [52-58]. Recently, the BOF approach emerged as a valuable tool for classifying medical image [59-62]. Bhatt and Soni [63] presented CBIR based BOF technique for the extraction of features and their classification on the dataset of LIDC-IDRI. The approach yielded training and testing accuracy of 99% and 98.56%, respectively.

A wide range of detectors and descriptors has already been suggested [64-66]. The Scale-Invariant Feature Transform (SIFT) algorithm is used for the identification and extraction of features suggested by Lowe [67]. The image feature extracted from the SIFT algorithm has a very high uniqueness, such that it can be used in a large database for accurate matching. The scale, rotation, illumination and viewpoint can be varied but still good results are achieved in SIFT algorithm. Constructing a scale space is the initial preparation which generates internal representations of the original image. The Laplacian of Gaussian (LoG) is great for finding key points in an image using a super-fast approximation. Maxima and minima are obtained by applying the Difference of Gaussian (DoG) on images. The edges and low contrast regions have

been omitted in order to make the algorithm effective and robust. Combinations of Local Binary Pattern (LBP) and SIFT descriptors have been proposed for image matching [68]. Krishna et al. [69] suggested a method based on multi-layer perception back proportional neural network using SIFT algorithm for lung cancer classification which achieved the accuracy of 89%. Another method for the extraction of features, Speeded-up Robust Features (SURF) was used and presented [70]. SURF is an approach to detect and describe local image features and finding points of significance in various images that have varying size, viewpoints, depths, scale changes and invariance under rotation. It basically makes use of invariant feature transformation for scale and rotation. It is the best feature descriptor to represent the local features into numerical vector. Three feature extraction algorithms i.e., SIFT, SURF and Principal Component Analysis (PCA) have been proposed for lung cancer detection [71]. The error rate observed for SURF algorithm was less than other two algorithms.

Another approach for feature classification is using K-means to create a codebook. The K-means algorithm works multiple times to make a cluster group. Typically, K-means is used in cluster centers with features that are derived from all images. These cluster centers are then used as a vocabulary for all images to get representations of word vector. A feature vector is created for every image by comparing all the image features to the codebook. It represents the distribution of every group in that image from the codebook and will not allow for its spatial relationship. Sangamithra and Govindaraju [72] proposed K-means algorithm for lung tumour classification and detection. Back propagation network was employed for classification and it correctly identified the image as either a normal image or a tumour image with an accuracy of roughly 90.87%. Kanan and Jagan [73] have introduced K-means clustering algorithm to detect lung cancer. Otsu's Thresholding and K-means clustering methods have been discussed for lung cancer image segmentation and shown that K-means clustering methods give better segmentation result. A Hybrid combination of deep learning and fuzzy along with hashing algorithm has been proposed for distributed cloud to retrieve the information [74]. Potghan et al [75] detected lung cancer using CT scan images from LIDC-IDRI dataset. In their study, separation of lung volume is conducted by a K-means clustering algorithm. Feature extraction was applied using deep learning architecture and then classification performed using Multi-Layer Perceptron (MLP) and k-Nearest Neighbours (kNN). The approach accomplished 98.30%

accuracy with kNN classifier and 98.31% accuracy with MLP classifier. K-means clustering algorithm was used for clustering lung cancer on features extracted using SURF algorithm [63].

Matching feature points is an important basis for image matching in computer vision. Currently, Brute Force (BF) and Fast Library for Approximate Nearest Neighbours (FLANN) are the most common matching algorithms. The BF always searches out all possible matches. It scans all matches one by one and when it finds a similar one, it sets it as the best one. FLANN is a series of algorithms optimized for quick nearest neighbour search in broad datasets with high dimensional features. It is applied for local feature matching to match the query image and reference image in data set. Two dictionaries are required for FLANN based matcher. The first dictionary sets out the appropriate algorithm. The second dictionary defines the search criteria to be passed. This dictionary points out the number of times the trees in the index have to be recursively traversed. To minimize the number of false matches, the FLANN based matching is done by the distance ratio between the two nearest matches of the considered key point and is a successful match when this value is below the threshold. Thus, FLANN constructs an effective data structure that will be used to check for an approximate neighbour. The matching speed of the FLANN algorithm is higher than that of the BF [76]. Bhatt et. al. [77] proposed the SIFT and FLANN based algorithm to detect the abnormalities in the NLST CT scans as the exact clinical nodule locations are not provided in the dataset.

2.1.2 Hybrid Model for Lung Cancer Classification

Support Vector Machines (SVMs) are powerful machine learning techniques for classification and regression [78]. Together with regression estimation and linear operation inversion, the SVMs are capable of providing a novel approach to pattern recognition problems and can establish connections with learning theories from statistics very clearly [79]. In a variety of areas, SVMs have provided a number of successful applications which include pattern recognition, supervised classification techniques, biometrics, image analysis and bioinformatics [80]. SVM algorithm relies on the idea of finding the maximum margin line that minimizes the error of classification among classes. It is a powerful and versatile algorithm, which efficiently performs linear, nonlinear classifications, regressions and outlier detection. The reason these algorithms perform so good is that they take extreme cases (what might be confusing for computers to identify) of classes, which are close to the margin line and it uses that to construct

its analysis. Detection of early-stage lung cancer in CT scan images using an image processing technique has been presented where the images were pre-processed by segmentation of the Region of Interest (ROI) of the lung and Gray Level Co-occurrence Matrix (GLCM) to extract the features [81]. Then extracted features were fed into SVM classifier and then they got an accuracy of 95.16%, sensitivity of 98.21% and specificity of 78.69%. Makaju et al [82] implemented image processing and machine learning to detect lung cancer in CT scan images. They used watershed segmentation and SVM classifier and achieved an accuracy of 92%, sensitivity of 100% and specificity of 50%. In order to distinguish normal from abnormal lung regions, SVM was used by Yao et al. [83]. Shao et al. [84] proposed a SVM based lung cancer classifier and achieved an accuracy of 93.41%. Ciompi et al. [85] proposed method for lung cancer classification using combination of intensity features and SVM on Multicentric Italian Lung Detection (MILD) train dataset and Danish Lung Screening Trial (DLCST) test datasets which provided accuracy of 39.9%. They also proposed the ConvNet for lung cancer classification for three different scales on same datasets and they achieved maximum accuracy of 79.5%.

A hybrid strategy was presented by Bhatt et al. [86] that uses deep neural networks for feature extraction and machine learning for feature categorization. The procedure for extracting the nodules from the LUNA16 dataset using CT images is the first stage. Patches were employed for feature extraction and classification rather than whole images. Residual Network-50 (ResNet-50) was used as a feature extractor to analyze the LUNA16 in this manner. The deep characteristics were categorized using SVM. On the LUNA16 dataset, the approach attained an accuracy of 97.53%.

2.2 Deep Learning Approaches for Lung Cancer Classification

Deep learning techniques have considerably advanced the state-of-the-art in many fields, including computer vision [87, 88], natural language processing [89], signal and speech recognition [90-92] and object recognition [93, 94]. Thus, deep learning techniques have been implemented to various detection and classification tasks. Deep learning's adaptive representation learning capability is superior to classical feature engineering at capturing and extracting the complex structures in high-dimensional data. Because deep learning

doesn't need as much manual engineering as traditional methods do, it could be more successful in the future [95-97].

One of the deep learning approaches i.e., Convolutional Neural Network (CNN), a supervised learning technique especially suited to solve issues of classification of natural images and have recently been used for several applications in chest CT analysis [98-101]. The CNN is a kind of multi-layer neural network that is meant to identify visual patterns directly from pixel images with little pre-processing. CNNs contain several parameters and hyper-parameters, such as weights, biases, the number of layers and processing units (neurons), filter size, stride, activation function, learning rate, etc. [102, 103]. As convolutional operation considers the neighbourhood (locality) of input pixels, therefore different levels of correlation can be explored by using different filter sizes. Different sizes of filters encapsulate different levels of granularity; usually, small size filters extract fine-grained and large size extract coarse-grained information. The CNN architecture is specifically suited for image data processing. This architecture may retain the original data structure while also producing hierarchical representations. CNN has the capacity to reduce the quantity of data processing, increase network resilience and reduce the likelihood of over-fitting. CNN has the benefit of the pooling operation, which further minimizes the amount of computation and considerably enhances data processing efficiency and achieve true automated operation by operating the whole process using GPU.

CNNs have been employed for lung nodule diagnosis [104, 105] with encouraging results. Ciompi et al. [104] classified candidates as peri-fissural nodules (PFNs) or non-PFNs using pre-trained CNN models. Deep features from the pre-trained model were retrieved for three 2D image patches in axial, coronal and sagittal perspectives. Then, for the PFN classification problem, an ensemble of deep features and a bag of frequency features were utilized to train supervised binary classifiers. Kumar et al. [105] described a CADx system for nodule classification that uses deep features extracted from a Stacked Auto Encoder (SAE). This method obtained 75.01% overall accuracy and 83.35% sensitivity. Rehman et al [106], built a lung cancer detection model using CNN and the dataset used in this study was taken from Japanese Society of Radiological Technology (JSRT) and they got an accuracy of 88%. Many studies have observed the diversity of nodule sizes and shapes using CNN and suggested approaches using multi-scale characteristics [107]. Shen et al.

[108] introduced a Multi-scale CNN (MCNN) approach that learns a collection of class-specific features using 3D nodule patches. On the dataset of the LIDC-IDRI, the suggested technique attained an accuracy of 86.84%. Juan and Sai [109] suggested a multi-level CNN for ternary nodule classification, achieving 84.81% accuracy without the need of a hand-crafted pre-processing approach. Setio et al. [110] proposed multi-view CNN for lung cancer classification using LIDC-IDRI dataset and achieved sensitivity of 90.1%. Hua et al. [111] trained models to identify lung nodules as benign or malignant using deep belief networks and CNNs. Later in 2017, the approach was expanded from MCNN to multi-crop CNN (MCCNN) [112]. It was tested on 880 benign and 495 malignant nodules from LIDC-IDRI and had an accuracy of 87.14%.

The machine learning approaches involve two steps consisting of feature extraction and classification. So, in order to have end-to-end models for lung cancer classification, seven distinct Deep Neural Networks (DNNs) namely MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, Visual Geometry Group-16 (VGG-16) and VGG-19 are trained in this work and based on their performance and challenges a new model is proposed which is named as Simplified VGG (SVGG).

2.2.1 MobileNet

MobileNet is a light-weight CNN with 28 layers deep (27 convolution and one fully connected layers) [113, 114]. The conventional convolution process filtering and merging data based on the convolutional kernels to build a new representation. For a significant decrease in computational cost, the filtering and combining phases may be divided into two parts using factorized convolutions termed as DepthWise Separable Convolutions (DWSC). When compared to networks with normal convolutions of the same depth, MobileNet utilizes DWSC, which minimizes the number of parameters. As a consequence, lightweight deep neural network is created.

A deep neural network-based technique for lung cancer detection and prediction was presented by Rahman et al. [115]. The suggested method was utilized to anticipate the development of lung cancer as well as to identify it in its early stages. The suggested approach consisted of two stages: First, pre-processing techniques like blurring and thresholding were used on the CT images to enhance their quality and make them easier to read. Then the left and

right lungs divided to reduce the pattern and decreased the complexity of the computation over time as well as improve the accuracy of the deep neural network. They have used MobileNet, VGG-8 and Inception-V3 deep neural network models as an image classifier to validate the method. The MobileNet model has been awarded the greatest accuracy of 97% among the several train models, with 96.26% sensitivity and 97.85% specificity [115]. Souid et al [116] presented the categorization and prediction of lung diseases in frontal thoracic X-rays using a modified version of MobileNet V2. Utilizing the NIH Chest-Xray-14 database, they evaluated the effectiveness of their algorithm against other cutting-edge approaches for pathology classification. The major comparison examined the variations across classifiers using Area under the Curve (AUC) statistics. Overall, with an accuracy of 95% and an average AUC of 0.811, they noticed a significant dispersion in the result obtained. The nodule segmentation and feature extraction in CAD system from raw 3D nodule CT patches using CNNs was suggested by Zhang et al. [117]. The MobileNet model has been awarded the greatest accuracy among the other train models, with a 71.13% sensitivity and 83.95% specificity.

2.2.2 SqueezeNet

SqueezeNet [118] is an example of lightweight network which consist of a small CNN that can categorize images into 1000 distinct categories at a depth of up to 18 learnable layers. Eight fire modules are placed before the first stand-alone convolution layer in the network, which is followed by a final convolution layer. The core of SqueezeNet is the proposed fire module, which comprises of the squeeze and expand components. The squeeze part is a 1 × 1 convolutional kernel and a 1 × 1 convolutional layer. The expand part is 1 × 1 and 3 × 3 convolutional kernels and convolutional layers, respectively. For the identification of lung CT images, SqueezeNet was proposed [119]. On the test dataset, the model had 83% accuracy, 85% sensitivity, 81% specificity and F1 value of 0.833.

2.2.3 AlexNet

LeNet [120] though, begin the history of deep CNNs, but at that time, CNN was limited to hand digit recognition tasks and didn't perform well to all classes of images. LeCun et al. [121] used a first-time backpropagation algorithm in multilayered CNN, namely ConvNet, to identify

handwritten zip codes. A pulmonary nodule identification method employing shape-based and statistical characteristics in CT images was described by Khehrah et al. [122]. An artificial neural network-based method for detecting lung nodules utilizing texture and form characteristics was presented in a different research [123]. The accuracy of the model was 89.62%. Similar to this, Nasser and Abu-Naser [124] suggested an ANN-based lung cancer detection model that had 96.67% accuracy. A neural network using CT scans was used by Miah et al. [125] to demonstrate a lung cancer detection system that achieved 96.67% accuracy. The LeNet-5 model was used by the Zhang et al. [126] to categorize benign and malignant pulmonary nodules in thoracic CT images. The experimental findings were collected using LIDC-IDRI databases. The model classification was tested using the 10-folder cross-validation method. The LeNet-5 classified benign and malignant nodules with an accuracy of 97.04% and severe and moderate malignancies with an accuracy of 96.68%.

AlexNet [93] is considered as the first deep CNN architecture, which showed ground breaking results for image classification and recognition tasks. In early 2000, hardware limitations curtailed the learning capacity of deep CNN architectures by restricting them to a small size. In order to get the benefit of the representational capacity of deep CNNs, AlexNet was trained in parallel on two NVIDIA GTX 580 GPUs to overcome shortcomings of the hardware.

Before AlexNet, the most commonly used activation functions were *sigmoid* and *tanh.* Due to the saturated nature of these functions, they suffer from the Vanishing Gradient (VG) problem and makes it difficult for the network to train [127, 128]. AlexNet uses Rectified Linear Activation Unit (*ReLU)* activation function which doesn't suffer from the VG problem [129-132].

In AlexNet, depth was extended from 5 to 8 layers to make CNN applicable for diverse categories of images. Generally, depth improves generalization for different resolutions of images but, the main drawback associated with an increase in depth is overfitting. To address this challenge, Krizhevesky et al. [93] exploited the idea of [133, 134], whereby their algorithm randomly skips some transformational units during training to enforce the model to learn more robust features. Overlapping subsampling and local response normalization were also applied to improve the generalization by reducing overfitting. Moreover, data augmentation is carried out to reduce over-fitting.

In [135], the network automatically retrieved the characteristics from CT scan images. To increase training phase precision and validation accuracy, binary cross-entropy was used. 99% training precision and 97% validation accuracy were attained by the lung nodule classification model utilizing AlexNet. A hybrid convolutional neural network based on LeNet and AlexNet was published by Zhao et al. [136] for the categorization of lung nodules. The LIDC-IDRI dataset had 1018 CT images, which were used to train and test the agile convolutional neural network. The framework obtained 82.2% accuracy and 0.877 AUC, with the kernel size set to 7×7, the batch size at 32 and the learning rate at 0.005.

In order to identify and categorize lung cancer, Yasriy et al. [137] developed a computer-assisted method-based CNN using AlexNet architecture. The lung cancer CT scan dataset, which was used to train and test the system, was gathered from Iraqi hospitals. The system scored a 0.96 F1 score, 95% specificity, 95.71% sensitivity, 97.10% precision and 93.54% accuracy. To extract lung areas, Agarwal et al. [138] employed multilevel thresholding. Regions were segmented using the morphological and thresholding approaches. After segmentation of the tumor regions, AlexNet-CNN categorized the tumor areas into benign and malignant with 96% accuracy.

Bhandary et al. [139] introduced Modified AlexNet (MAN), which uses X-ray images to distinguish between two classes: normal and pneumonia class. The SVM model was used to categorize lung cancer using the MAN approach. Utilizing the LIDC-IDRI dataset, the MAN method outperformed other deep learning techniques including ResNet-50, VGG-19, VGG-16 and AlexNet achieved 97.27% accuracy.

2.2.4 GoogLeNet

The main objective of the GoogLeNet architecture was to achieve high accuracy with a reduced computational cost [140]. It introduced the new concept of Inception block in CNN [141, 142] whereby it incorporates multi-scale convolutional transformations using split, transform and merge idea. The exploitation of the idea of split, transform and merge by GoogLeNet helped in addressing a problem related to the learning of diverse types of variations present in the same category of images having different resolutions. In GoogLeNet, conventional convolutional layers are replaced in small blocks similar to the idea of substituting each layer with micro–Neural Network (NN) as proposed in Network in Network (NIN) architecture [143]. This block

encapsulates filters of different sizes (1x1, 3x3 and 5x5) to capture spatial information at different scales, including both fine and coarse grain level. GoogLeNet regulates the computations by adding a bottleneck layer of 1x1 convolutional filter before employing large size kernels. In addition to it, it used sparse connections (not all the output feature-maps are connected to all the input feature-maps), to overcome the problem of redundant information and reduced cost by omitting feature-maps that were not relevant. Furthermore, connection's density was reduced by using global average pooling at the last layer instead of using a fully connected layer. The parameter tunings caused a significant decrease in the number of parameters from 138 million to 4 million parameters. Other regulatory factors applied were batch normalization [144] and the use of Root Mean Square Propagation (RMSprop) as an optimizer [145]. GoogLeNet also introduced the concept of auxiliary learners to speed up the convergence rate. However, the main drawback of the GoogLeNet was its heterogeneous topology that needs to be customized from module to module. Another limitation of GoogLeNet was a representation of the bottleneck that drastically reduces the feature space in the next layer and thus sometimes may lead to loss of useful information.

A new computer-aided lung cancer screening technique based on Median Intensity Projection (MIP) and transfer learning from GoogLeNet was proposed by Fang [146]. In his study, a CNN structure resembling GoogLeNet was created by using a transfer learning strategy. Unlike other research, MIP was used to include the multi-view capabilities of 3D CT images. The system was tested using lung nodule images from the LIDC-IDRI public dataset and 100-fold data augmentation was done to assure effective training. The trained system provided 81% accuracy, 84% sensitivity and 78% specificity. Huseiny and Sajit [147] suggested the usage of DNN to identify images of lung CT scans with cancerous nodules. The Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases (IQ-OTH/NCCD) lung cancer dataset was used to create a learning model using the GoogLeNet. According to experimental findings, the trained model has improved its overall accuracy on the validation data by 94.38%. Artificial Immune Approach-based Inception Networks Fusion (AIA-INF) algorithm [148] obtained overall sensitivity, specificity and accuracy of 82.22%, 93.17% and 88.67% respectively.

2.2.5 Residual Network

Residual Network (ResNet) was proposed by He et al. [149], which is considered as a continuation of deep networks. ResNet revolutionized the CNN architectural race by introducing the concept of residual learning in CNNs and devised an efficient methodology for the training of deep networks [150].

ResNet introduced shortcut connections within layers to enable cross-layer connectivity. However, in ResNet, residual information is always passed and identity shortcuts are never closed. Residual links (shortcut connections) speed up the convergence of deep networks, thus giving ResNet the ability to avoid gradient diminishing problems. ResNet-18 has around 11 million trainable parameters. It consists of Conv layers with filters of size 3x3. Only two pooling layers are used throughout the network one at the beginning and other at the end of the network. Identity connections are used between every two Conv layers.

ResNet gained a 28% improvement on the famous image recognition benchmark dataset named Common Objects in Context (COCO) [151]. ResNet, which was 20 times deeper than AlexNet, showed less computational complexity than previously proposed networks. Bhatia et al. [152], detected lung cancer from CT scan images using deep residual learning where Unet and ResNet models were used for feature extraction whereas, Extreme Gradient Boosting (XGBoost) and Random Forest used for classification. They achieved an accuracy of 84% on LIDC-IDRI dataset.

A transfer learning approach was used by the Sajja et al. [153] to provide a lung cancer detection system. The recommended approach used a maximum dropout ratio to shorten processing time and prevented overfitting during the learning phase. Lung cancer was identified using the transfer learning CNNs designs GoogLeNet, AlexNet and ResNet-50. When compared to pre-trained approaches, the proposed model and the pre-trained model both produced excellent accuracy using LIDC-IDRI. A technique based on ResNet-18 architecture was proposed by Nibali et al. [154] for classifying lung nodules. Using LIDC-IDRI, the suggested model attained an accuracy of 89.90%. Similarly, in [155], pulmonary nodule identification using the Darknet-53 CNN-based architecture was implemented on the LUNA16 dataset which obtained accuracy ranged from 70.5% to 73.9%. For the purpose of detecting pulmonary nodules in LDCT, Cui et al. [156] built a deep learning model based on ResNet. The deep learning model's AUC of 0.86 was higher than the radiologists' AUC of 0.73 and it took much

less time to complete the identical detection job. Using whole-slide histopathology images, Saric et al. [157] proposed CNN architectures based on Visual Geometry Group (VGG) and ResNet for lung cancer detection and the outcomes were compared using the receiver operating characteristic (ROC) plot. Patch level accuracy for ResNet-50 and VGG-16 was 75.41% and 72.05%, respectively, which is a very poor result.

2.2.6 Visual Geometry Group

Simonyan et al. [158] proposed a simple and effective design principle for CNN architectures named as Visual Geometry Group (VGG). VGG regulates the complexity of a network by placing 1x1 convolutions in between the convolutional layers, which, besides, learn a linear combination of the resultant feature-maps. For the tuning of the network, max-pooling is placed after the convolutional layer, while padding was performed to maintain the spatial resolution [159].

VGG showed good results both for image classification and localization problems. The use of VGG increased due to its simplicity, homogenous topology and increased depth. The main limitation associated with VGG was the use of 138 million parameters, which make it computationally expensive and difficult to deploy it on low resource systems.

VGGNet was born out of the need to reduce the of parameters in the convolution layers and improve on training time. There are multiple variants of VGGNet (VGG-16, VGG-19, etc.) which differ only in the total number of layers in the network.

VGGNet consists of 16 convolutional layers and is very appealing because of its very uniform architecture. The important point to note here is that all the conv kernels are of size 3×3 and maxpool kernels are of size 2×2 with a stride of two.

A precise lung segmentation method based on VGG-16 and a dilated convolution network was presented by Geng et al. [160]. Dilated convolution indicated an increase in the size of the relevant field and employed a dilation rate parameter. To improve the robustness of the lung segmentation approach, multi-scale convolution features were fused using the hypercolumn features technique. The multilayer perceptron (MLP) and ReLU activation function were employed after the improved VGG-16. A dice similarity coefficient of 0.9867 was obtained using the procedure.

VGG-16 was introduced with a boosting strategy by Pang et al. [161] intended to identify the pathological kinds of lung cancer. The 125 individuals in the dataset had early-stage lung cancer and replicating, shifting and rotating procedures were used to improve it. VGG16-T has a kernel size of 3×3 and five convolution layers. In comparison to ResNet-34 and AlexNet, VGG16-T with boosting produced experimental results with an accuracy of 85%.

VGG was made 19 layers deep compared to AlexNet to simulate the relation of depth with the representational capacity of the network [93]. It was suggested that small size filters can improve the performance of the CNNs. Based on these findings, VGG replaced the 11×11 and 5×5 filters with a stack of 3×3 filters layer and experimentally demonstrated that concurrent placement of small size (3×3) filters could induce the effect of the large size filter $(5 \times 5 \ and \ 7 \times 7)$. The use of the small size filters provides an additional benefit of low computational complexity by reducing the number of parameters. These findings set a new trend in research to work with smaller size filters in CNN.

Simonyan and Zisserman [162] examined the impact of VGG-16 and VGG-19 on ImageNet Challenge 2014. The experimental findings demonstrated that the depth representation is useful for categorizing issues and improved state-of-the-art accuracy. Both networks were also tested on different datasets and achieved high accuracy as compared to other techniques. Dehmeshki et al. [163] proposed a shape-based Genetic Algorithm Template-Matching (GATM) which was trained on 70 Thoracic CT scan data and the accuracy of 90% was obtained. Ye et al. [164] presented a shape-based CAD algorithm which was trained on 108 Thoracic CT scan data and the accuracy of 90.2% was obtained. Chaoi et al. [165] proposed a 3D shape-based feature descriptor which was trained on 84 LIDC-IDRI datasets and achieved an accuracy of 97.4%.

Elnakib et al. [166] developed a deep learning genetic system to identify early lung cancer. Three strategies were employed in the pre-processing method: the histogram stretching technique was used to improve the contrast of the raw image; the Wiener filter was used to eliminate noise; and the image was cropped into 224×224 for VGG-16 and 227×227 for VGG-19 for the AlexNet architecture. To extract features, LDCT images were employed and three CNN architectures, VGG-16, VGG-19 and AlexNet, were used. The most relevant features were chosen using a genetic algorithm. Finally, to identify pulmonary lung nodules, k-nearest neighbour (KNN), decision tree and SVM classifiers were examined. When compared to other CNN models and classifiers, the experimental findings showed that VGG-19 with a

SVM classifier achieved a stunning 96.3% accuracy, sensitivity of 97.5% and specificity of 95%. Bhatt and Soni [167] proposed a Simplified VGG (SVGG) model based on the VGG-16 framework and achieved an accuracy of 99.60% for lung cancer classification.

2.3 Object Detection Approaches for Lung Nodule Diagnosis

There are several applications for the object identification technology in the areas of computer vision [168, 169]. Object detection combines object classification and localization. Classification is the challenge of selecting one label from a group of predefined options. The localization problem is resolved by surrounding the object with a bounding box [170]. Object detection is vital for face identification [171, 172], autonomous driving [173] and medical image analysis [174-177]. Object detection is the fundamental goal of computer vision. Using the object detection paradigm, objects are recognised in the feature space. The necessary computer infrastructure for the aforementioned applications may be provided by cloud computing services, generic Graphics Processing Units (GPU), Internet of Things (IoT) clusters, or a single embedded device. The objective of generic object detection is to identify and classify the objects inside an image. Prior to the advent of deep CNN, traditional detection was relied on handcrafted features such as SIFT [67], Linear Binary Patterns (LBP) [68], Deformable Part Model (DPM) [178], etc.

Deep CNN methods are used most often for detection because they are better at machine learning tasks. A modern detector usually has two parts: a backbone that is a classifier that has already been trained and a head that predicts class labels and bounding boxes. Backbones for detectors that run on GPU platforms could be GoogLeNet [140], ResNet [149], VGG [158] and so on. The core of detectors that run on CPU platforms could be MobileNet [113] or SqueezeNet [118]. Most of the time, the head can be split into two types: one-stage and two-stage object detectors [179].

The most common two-stage object detector includes Fast R-CNN, Faster R-CNN and R-FCN. Whereas, Single Shot Detector (SSD), YOLO, YOLOv2, YOLOv3 and YOLOv4 are the most well-known models for one-stage object detectors. Object detectors designed in recent years often incorporate specific layers between the backbone and the head, which are generally used to acquire feature maps at different stages. The neck of an object detector is often composed

of several bottom-up and top-down routes. This approach is used in networks such as the Path Aggregation Network (PAN) and the Feature Pyramid Network (FPN).

2.3.1 Faster R-CNN

The Faster R-CNN model is made up of three neural networks: a Feature Network, a Regional Proposal Network and a Detection Network [180]. The Feature Network is in charge of producing excellent features of the input images while retaining the original qualities of the input image, such as structure and shape in the output. In general, an image classification network serves as a Feature Network [181]. The Regional Proposal Network (RPN) is made up of three convolutional layers: one for classification and one for bounding box regression, with the third serving as a common layer that feeds into the first two. The Regional Proposal Layer is in charge of creating a large number of bounding boxes with a high likelihood of including an object. These bounding boxes are also known as Regions of Interest (ROIs) [182]. A mechanism known as anchor boxes generates a series of bounding boxes. Anchors are just the pixels that appear in the feature image. A bounding box is identified by the co-ordinates of the pixels situated at the box's two diagonal corners, followed by a value of 1, 0 or -1. A value of 1 signifies that an object is present in that bounding box. Similarly, a 0 means that no object is present, whereas a value of -1 implies that the specific bounding box may be disregarded [181]. The Non-Maximum Suppression (NMS) procedure is used to lower the number of boxes by deleting boxes that overlap with the other boxes with a high score based on the possibility of having an object. The probability ratings are then normalized using the SoftMax function. Along with the output of the Feature Network, the resultant bounding boxes (ROIs) are sent into the Detection Network. The Detection Network is responsible for creating the final class and its matching bounding box using data from both the Feature Network and the Regional Proposal Network. The Detection Network typically consists of a classification layer and a regression layer with bounding boxes. Furthermore, the two layers share a pair of stacked common layers. These four layers are fully connected layers. The features are trimmed according to the bounding boxes, such that the network only classifies the inside portion of the bounding boxes [182]. There were many studies [183-185] where the Faster R-CNN approach was used for lung cancer detection.

Ding et al. [186] used Faster R-CNN with a deconvolutional structure to discover nodule candidates from axial slices. Xie et al. [187] proposed a Faster R-CNN based automated pulmonary nodule detection approach for lung cancer classification using Luna16 dataset. This approach obtained the recall of 86.2%. A 3D Deep Convolutional Neural Network (DCNN) was subsequently used to minimize the number of false positive data. On the LUNA16 dataset, this approach obtained 94.6% sensitivity with 15 false positives per scan. Zhu et al. [188] suggested a 3D Faster R-CNN for detection of lung nodules and a deep 3D dual route for nodule classification. On the same dataset, they got a competition performance metric (CPM) score of 84.2% for nodule detection. Su et al. [189] used experiments to fine-tune and optimize the hyperparameters of the Faster R-CNN detection network, achieving 91.2% average precision on the LIDC-IDRI dataset. Ardila et al. [190] proposed 3D Inception-V1 for lung cancer detection on NLST dataset and got the accuracy of 94.4%.

Li et al. [191] tested different Faster R-CNN variants using a private 3D thoracic magnetic resonance imaging dataset. At 3.47 false positives per scan, the fine-tuned model obtained 85.2% sensitivity. Wang et al. [192] suggested a nodule-size-adaptive technique for candidate identification from 3D CT images based on Faster-RCNN. Their plan is to identify nodules ranging in size from 3 mm to 70 mm by estimating nodule aspect ratios of 0.6, 1.0 and 1.65. A clustering approach is also used to discover a 3D candidate from 2D bounding boxes. Then two Inception-V4 networks with distinct receptive fields are considered as a classifier for false positive reduction. On the LUNA16 dataset, the authors acquired a sensitivity of 90% with 15 false positives per scan and a CPM score of 90.3%. Liao et al. [193] proposed a 3D Faster R-CNN based on combination of 3-D region proposal network, leaky noisy OR Gate, backbone network, a modified U-net. This model was trained on LUNA16 and DSB2017 dataset and achieved recall of 85.62%.

Tong et al. [194] presented Iterative Self-Organizing Data Analysis Techniques Algorithm (ISODATA) for analyzing anchor box to Faster R-CNN in order to further increase candidate identification performance in a more generic way. This is a K-means clustering modification that automatically merges and splits clusters throughout its iteration phase, making it more adaptable than the original K-means approach. On the LUNA16 dataset, this K-means-based anchor box clustering approach achieves 91.43% sensitivity with 3.19 false positives per scan. Fan et al. [195] offered a Local Density based D-ISODATA algorithm for automatically creating

anchor boxes in a similar study. This study has a sensitivity of 93.6% and a false positive rate of 0.15 per scan. Although these approaches have shown reasonable results, the majority of them depend on manually constructed configurations or basic clustering algorithms to create Faster R-CNN anchors, which are seldom appropriate given the variety of nodule sizes and shapes in CT images.

Several efforts to develop a quicker detector by altering each step of the detection pipeline have been attempted. However, any considerable improvement in speed caused by such adjustments only resulted in a reduction in detection accuracy, leading researchers to conclude that, rather than modifying an existing model, they would have to develop a completely new object identification model.

Each anchor suggests a bounding box with a varying level of confidence. The issue is that an object often collects many region proposals. At the completion of each of these algorithms, a post-processing procedure known as NMS [196] is performed to eliminate duplication amongst region proposals by maintaining only the one with the greatest confidence score when compared to its neighbors.

2.3.2 Single Shot Detector

Liu et al. [197] developed a novel approach for detecting objects within images using a single deep neural network. This method was called the Single Shot MultiBox Detector (SSD). SSD is much simpler than other approaches that need object proposals since it fully eliminates feature resampling phases or pixel and proposal creation by including all processing in a single network. As a result, SSD is very easy to train and may be quickly included into systems that have detection as part of their roles. Its design is highly reliant on the creation of bounding boxes as well as the extraction of feature maps, that are also referred to as default bounding boxes. Khosravan and Bagci [198] investigated a down-sampling approach for tiny nodule detection using Single-Shot Single-Scale Lung Nodule Detection (S4ND). These frameworks use just a single scale feature map, which limits their ability to recognize nodules of varying sizes.

The network calculates loss by comparing the predicted class sets and the default bounding boxes to the ground truth values of the training samples using various filters for each iteration. All parameters are changed using the back-propagation technique and determined loss value. This allows SSD to learn the most ideal filter structures for reliably identifying object

characteristics and generalizing the provided training samples in order to minimize the loss value, resulting in higher accuracy during the testing. SSD is simple to train and integrate which facilitates detection. SSD's main feature is the use of multiscale convolutional bounding box outputs which are coupled to several feature maps [197].

2.3.3 You Only Look Once

Current detection methods repurposed classifiers to perform detection. To identify an object, this system uses numerous object categorization evaluations, object locations and scales. Some techniques, such as Deformable Part Model (DPM), use the sliding window approach. The categorization via the sliding window method is applied uniformly throughout the whole image. Region proposal approaches first generate probable bounding boxes in an image and then classify the boxes. Post-processing is then used to refine the bounding boxes. The boxes are then rescored based on other objects present in the image and duplicate detections are removed. Because each component is trained separately, these sophisticated pipelines are sluggish and difficult to implement.

Consequently, the object identification problem is reframed as a single regression issue. To identify and find an object in an image, the algorithm examines the image and the object's complete features. This approach, in which an image is seen just once, is termed as You Only Look Once (YOLO) [199].

Various bounding boxes and class probabilities for the boxes are computed using a single neural network at the same time in this method. This technology offers numerous benefits over traditional object detection methods. The YOLO technique is rapid, thus there is no need for a sophisticated pipeline in detection as a regression problem. The second benefit of this method is that it provides universal grounds for predicting images. The YOLO methodology involves seeing the complete image during training and evaluation. As a result, YOLO encodes the contextual information of classes as well as their appearance. The third benefit is that YOLO can train generalizable object representations [200].

A. You Only Look Once

The You Only Look Once (YOLO) is one of the fastest object detections. It employs the Darknet framework, which was trained on the ImageNet-1000 dataset. It has a substantially greater identification rate for unusual image objects than R-CNN series detection approaches.

To forecast the category and position of distinct targets, the one-stage approach employs CNN. The training and prediction are end-to-end and the algorithm is simple and fast. YOLO has a greater field of vision while identifying the target since it convolves the whole image and it is difficult to misinterpret the background. Masni et al. [201] suggested a deep learning YOLO-based CAD system for simultaneous detection and categorization of breast masses. Using the masses positioned over the pectoral muscle, this approach detects instances of difficult dense breast tissue.

Sindhu et al. [202] proposed the DetectNet architecture based on YOLO which uses a single convolutional network to identify different bounding boxes and class probabilities for lung nodules. Because the output layer is a fully connected layer, it had trouble generalizing objects when tested if the image was of different proportions from the training image.

YOLO is quick; however, it cannot detect little objects if they occur in a group. When the YOLO technique was compared to R-CNN, it was discovered that YOLO causes a considerable amount of localization errors. As a result, as compared to proposal-based methodologies, the YOLO methodology has a poor recall. Aside from maintaining classification accuracy, it is critical to focus on retrieving and localizing.

B. You only look once version 2

The second version, YOLOv2 [203], employs a customized deep architecture Darknet-19, a 19-layer network that has been extended with 11 additional layers for object detection.

In YOLOv2, the fully connected layer has been eliminated and the anchor boxes have been introduced to determine bounding boxes. YOLOv2 seeks to discover the optimal anchor box shapes to help the network to learn detection. One of the most important issues that must be solved in the YOLO is the recognition of tiny objects in the image. This has been addressed in YOLOv2 by splitting the image into 13*13 grid cells, which is smaller than in prior versions. This allows the YOLOv2 to recognize and locate tiny objects in the image while still being

successful with bigger ones. Furthermore, the YOLOv2 can distinguish a huge number of objects inside an image at the same time, which is one of the most important advantages over the YOLO [204]. Li et al. [205] integrated the Inception-V3 module into the YOLOv2 network to create a pulmonary nodule diagnosis network that can successfully extract multi-scale characteristics from 2D images.

C. You only look once version 3

The areas of object detection are always developing. The preceding algorithm must then be improved to make it quicker and more accurate. Even though previous versions of YOLOv2 were reliant and fast, it had some drawbacks. As a result, the YOLOv3 [206] was introduced in 2018 to improve the algorithm's accuracy and speed. YOLOv3 network delivers finer-grained information and so performs better for little objects.

The newest version of this method, YOLOv3, employs a new feature extractor known as Darknet-53, which employs 53 convolutional layers as the name implies, while the whole algorithm employs 75 convolutional layers and 31 additional layers for a total of 106 layers. For down sampling, pooling layers have been eliminated from the structure and substituted with another convolutional layer of stride '2'. This critical adjustment was implemented to avoid feature loss during the pooling process. To create predictions at scales 3, 2 and 1, the sizes of the input image were down sampled to 32, 16 and 8 by Feature Pyramid Network (FPN) [207, 208]. As compared to its predecessors, YOLOv2 and YOLO, YOLOv3 detects tiny objects better.

The YOLOv3 is intended for excellent accuracy as well as real-time performance. The bounding boxes in YOLOv3 have been replaced with anchors which resolves the unstable gradient problem that used to arise during algorithm training. As a result, whenever an image or video is fed into the algorithm, YOLOv3 predicts outcomes with confidence values by creating a collection of bounding boxes [209].

To appropriately assign and identify breast masses, Aly et al. [210] developed the YOLOv3 model in conjunction with the anchor boxes approach. The best results were achieved by identifying 89.4% of the masses in the INbreast mammograms with an average accuracy of 94.2% and 84.6% for categorizing the masses as benign and malignant, respectively based on mammograms with varied resolutions using YOLOv3. Based on YOLOv3, Xu et al. [211]

suggested a novel approach for identifying lung nodules. After experimental validation, the average precision and sensitivity achieved of 83.5% and 92.6% respectively. The YOLOv3 model is substantially more intricate than that of the previous model and the accuracy and speed may be balanced by altering the size of the model framework.

D. You only look once version 4

Wang et al. [212] suggested the YOLOv4 algorithm in 2020 based on YOLOv3, which combines the CSPNet (Cross Stage Partial Network) with Darknet53 as the backbone, termed CSPDarknet53. In addition, Spatial Pyramid Pooling (SPP) [213] and Path Aggregation Network (PANet) [214] were used to improve YOLOv4 feature extraction network [215]. With the Cross Stage Partial (CSP) network, YOLOv4 object detection maintains maximum speed and precision [216]. The YOLOv4 network scaling method adjusts the network's depth, width, resolution and structure. Bhatt et al. [77] proposed a SIFT and FLANN-based method for identifying abnormalities in NLST CT images. These recognised and labelled nodules were used to train a YOLOv4 nodule detector, which achieved 95% precision, 81% recall and 89.1% mean Average Precision (mAP). The YOLOv5 object detection method is a lightweight detection model that employs the CSP structure and adds it to the network's backbone and Neck in order to improve feature fusion.

E. You only look once version 5

The YOLOv5 object detection technique is a lightweight detection model that maintains to employ the CSP structure and adds it to the network's backbone and Neck to improve feature fusion. YOLOv5 additionally incorporates the Focus structure into the backbone network to partition the feature map, hence reducing the computational burden and accelerating the procedure.

Yan and Wang [217] suggested an enhanced YOLOv5 pulmonary nodule identification method to address the detection impact with varying CT equipment imaging quality in CT images. In addition, to increase the feature extraction capabilities of the model for pulmonary nodules, the backbone of YOLOv5 is modified by incorporating the Residual Block (ResBlock) unit of ResNet at various feature extraction stages [218]. Thus, the network's detection

capabilities for tiny pulmonary nodules were improved utilizing the concept of BiFPN, a cross level feature fusion mode placed on top of the original feature fusion network. Liu et al. [219] proposed an approach called STBi-YOLO, which is derived on YOLOv5, but makes substantial improvements from three dimensions by using the spatial pyramid pooling network, which is based on the stochastic-pooling method; then a bidirectional feature pyramid network is used to perform multi-scale feature fusion; and finally, the loss function of the YOLOv5 is improved and the EIoU function is used to optimise the trials reveal that STBi-YOLO detects lung nodules with an accuracy of 96.1% and a recall rate of 93.3% while using 4 times less memory than YOLOv5 and achieving equivalent results in terms of mAP and time cost as Faster R-CNN and SSD.

F. Proposed Model-1

In this work, YOLOv5 surpassed the five aforementioned nodule detectors, achieving 90% precision, 80% recall, 85% mAP, 0.85 F1 Score and 217 GFLOPs. However, it encountered a few obstacles, such as complexity, excessive calculation time and poor performance. Consequently, two automated lung cancer diagnostic models have been developed to address these issues. A few YOLOv5 layers are substituted with Focus and CSP layers in the initial suggested model. This model attained 85% precision, 82% recall, 85% mAP, F1 score of 0.84 and 35,2 GFLOPs.

G. Proposed Model-2

In order to lower the calculation time of Proposed Model-1, we suggest the second model, which is a variation of the previous one in which the backbone convolution layers are replaced with DWSC. With a precision of 95%, a recall of 90%, a mAP of 95%, an F1 score of 0.92 and 30.6 GFLOPs, Proposed Model-2 outperforms all the implemented nodule detectors in this work.

2.4 Research Gap

Based on the literature review, the research gap observed can be summarized as below:

- The anatomical structure of the lungs is widely recognized to make nodule identification

problematic for radiologists. . Hence, Lung cancer is rarely diagnosed in the early stage. Its solution is routine screening programmes using LDCT to detect the first signs of lung cancer. The early-stage lung cancer manifests itself in the form of pulmonary nodules visible on CT scans which are only a few millimetres in diameter. They appear as small bright spots surrounded by the darker lung parenchyma, with grey-values very similar to those of blood vessels in the lungs. The examination of a chest CT scan by radiologists for detecting the small sized nodules is a time-consuming and error-prone task because of human-error and fatigue. This motivated the development of CAD solutions for lung nodule diagnosis on CT scans. The CAD system mainly consists of two steps: 1) candidate screening and 2) FP reduction. The aim of the first step is to detect nodule candidates at a very high sensitivity which produces large number of FPs along with True Positives (TPs). The reason for large number of FP candidates is because they carry quite similar morphological appearance to the TP nodules. Thus, the aim of second step is to remove the FP nodules and thus improve the performance of CAD systems. Still the performance of CAD system is low because the pulmonary nodules have large variations in sizes, shapes, and locations. Moreover, the contextual environments around them are often diversified for different categories of lung nodules. Thus, the designers of CAD systems have to compromise on either high sensitivity or high FP rate.

- CAD uses two approaches: machine learning and deep learning. Machine learning algorithms consist of complex processes which involves separation of the lung region in CT, nodules selection, finding distinguishing characteristics and reduction in the FPs during classification. This process is time consuming to design, needs expertise for execution and tuning the hyper-parameters. While, deep learning has great potential to remove these subcomponents, such that the algorithm acquires knowledge indirectly from the data. But the deep learning models are complex which leads convergence issue and have high computation time.
- CAD systems have still more FPs thus sensitivity of nodule detection is low compared to experienced radiologists. It is understood that CAD systems advances the performance of radiologists in diagnosing process and nodule detection, however it does not become common in clinical usage. Regular usage in the clinics depends on meeting the following requirements: ensure in high sensitivity, decrease in the number of FPs,

increase in the diagnosis speed, advance in automation, decrease in running cost, advance in the capability of detection of various shapes and types of nodules.

- Moreover, the other challenges faced by CAD system are insufficiency of well labelled medical datasets, generalizability issues and poor interpretability of the detection results.

2.5 Discussion

This chapter is regarding the work done so far by different researchers in the development of the CAD system for lung cancer diagnosis. It involves a literature survey on the machine learning and deep learning approaches for lung cancer classification. In addition to this, the work carried out for lung cancer diagnosis using object detection techniques are also discussed. The chapter concludes with critical observations made based on literature review.

CHAPTER-3
Lung Cancer Classification Using CBIR Approach and Hybrid Model

Once medical images could be scanned and loaded into a computer, researchers started developing automated analysis methods. From the 1970s through the 1990s, medical image analysis was done sequentially utilizing low-level pixel processing and mathematical modelling to develop compound rule-based systems for specialized tasks. Many commercial medical image analysis software incorporates computer algorithms. Computer algorithms determine the decision boundary in high-dimensional feature space. The extraction of discriminant features from images is a critical step in the construction of such systems.

There are several existing effective machine learning-based and deep learning approaches for feature extraction and classification. Thus, incorporating knowledge from previous work on machine learning and deep learning frameworks, two techniques are implemented in this chapter. Among these two techniques the first one is CBIR which is purely based on machine learning algorithms for feature extraction and classification. While the second technique uses hybrid approach consisting of feature extraction by deep learning method and the features classification by machine learning method.

This chapter is divided into following sections: Lung cancer classification using CBIR approach, Lung cancer classification using the Hybrid model and finally the discussion about the methods implemented in this chapter.

3.1 Lung Cancer Classification Using Content Based Image Retrieval Technique

Content Based Image Retrieval (CBIR) is a technique for dynamically indexing and retrieving images based on their contents, known as features. CBIR is defined as a method for locating a comparable image database for a query image. The discrepancy between query image and database feature vectors is then computed and graded. The image database must be extracted with the greatest possible similarity to the sample image. The performance evaluation is then

carried out based on accuracy. CBIR may be used with Computer Aided Diagnosis (CAD) by allowing specialists to find previously diagnosed instances from a database that are similar to the cases they are interpreting. A bag of visual terms also known as a Bag-of-Features (BOF) is a typical method for implementing the CBIR technique. The BOF technique is a simple approach for viewing image content with collection-dependent patterns. The BOF model is an expansion of the bag-of-words structure necessary for data retrieval and text categorization. The BOF technique uses a similar approach for image analysis. The basic idea is to develop a codebook in which the most common characteristics are coded as codewords or graphic phrases. A histogram created by a basic image codeword frequency analysis is then used to represent images as BOF.

The details about datasets and processes involved in CBIR technique are explained in the next sub-sections.

3.1.1 Dataset

The LIDC-IDRI datasets are used to train the CBIR model. The collection includes 1018 CT images. The resolution of the volumes is 512 x 512. Each LIDC-IDRI dataset contains hundreds of images as well as an XML file with descriptions of the lung lesions discovered. The lung lesions were examined by radiologists and the nodules were categorized into three primary classes: nodules (3-30 mm in diameter), non-nodules (diameter more than or equal to 3 mm) and micro-nodules (diameter less than 3 mm). As this approach was implemented on low computing system of 8GB RAM so, a small set of data consisting of 210 CT scans from LIDC-IDRI datasets were used for training purpose.

3.1.2 System Architecture

The detailed configuration depicting the formulation of BOF for CBIR approach is shown in Fig. 3.1. Overall, the essential steps related with CBIR are feature extraction, vocabulary development, BOF formation and similarity calculation. Initially, the image is divided into grids. Each grid corresponds to a point in the feature map which is responsible for image identifying the content of the grid. The grid steps enable in the extraction of lower-level and higher-level feature map of nodules. This method takes into account a lot of blocks, which

reduces the possibility of missing crucial characteristics. Every block is identified by a set of features which are extracted using SURF. The concept behind the SURF descriptors is that local nodule appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. The representation is compact, easy to obtain and translation invariant. The features were thereby defined as visual words by k-mean clustering. Based on the occurrence of visual words, a histogram is generated. Binary support vector machine learning algorithms, that analyse data used for classification, was trained by matching the histogram and predict categories of general untrained images.

In [50], authors have examined the applicability of BOF for CAD to differentiate normal and abnormal radiographic findings evident in CT images of dogs. While in the proposed study, the same architecture shown in Fig. 3.1 with minor modifications in the parameters was used to diagnose lung cancer in humans. In the proposed work, the grid step has been increased from 4 to 8, the block width has been changed from 32,48,64,80,96 to 32,64,96,128 and the number of visual words has been reduced from 1,500 to 500 in order to reduce the computational time and storage requirements.

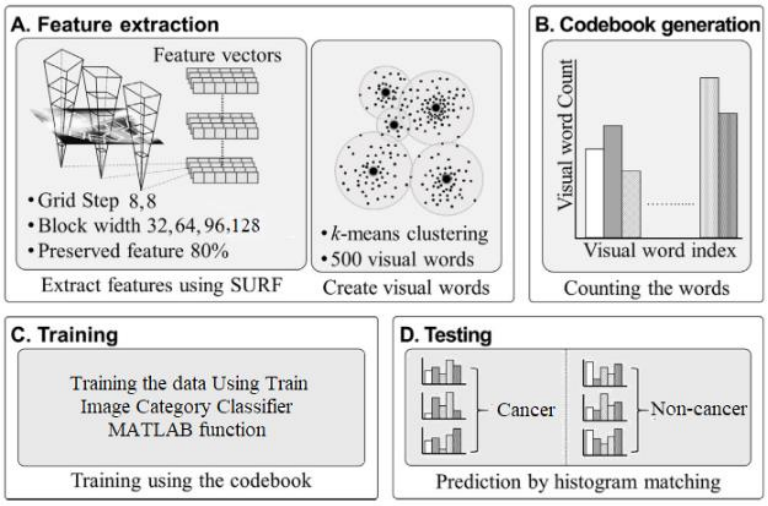


Fig. 3.1 Block diagram of CBIR (A) Feature extraction and creation of visual words (B) Codebook generation (C) Training using codebook and (D) Prediction of cancer or non-cancer using histogram matching [50]

The detailed description of the steps involved in CBIR technique are described in sub-sections below:

A. Block Extraction Using Grid Method

The array image form is used for retrieving the features required for classification. If the goal is to find unique objects within image sets, the local image characteristics retrieved by the object's key-points are a preferable alternative. The grid approach is distinct from standard long multiplication in that it clearly splits multiplication and addition into two phases and relies less on place value. The [8,8] step grid with block width [32,64,96,128] is used in this study.

B. Extracting Features using Speeded Up Robust Features (SURF) Algorithm

There are several machine learning algorithms for feature extractions. Among these Speeded Up Robust Features (SURF) is several times faster and is thought to be more robust to image transformations for these reasons, SURF was selected as the feature extractor in this method. SURF is a method for identifying and defining spatial features in images and discovering points of interest in disparate images with varying image sizes and points of view, varying distances, scale shifts and rotational invariance. Thus, in this step features of two categories are retrieved from the blocks extracted in the previous step.

C. Clustering and Classification

Features derived by SURF algorithm through entire training images are required to build a codebook with K-means clustering algorithm. To build a cluster group, the K-means algorithm is iterated numerous times. K-means are often used across all the images to form cluster centres. Thus, the aim of this step is to create a vector function for each image by matching all of its attributes to the codebook. It depicts the distribution of each cluster in the codebook image but ignores their spatial relationship. For each case, all features were allocated to each codeword according to the lowest possible Euclidean distance from these features. The squared Euclidean distance metric is used to select clustering index in K-means algorithm. This word vector representations may thus be described as a vocabulary for all images using these cluster centres. These vectors are then used by SVM for the classification.

D. Search for Identical Images

The last step is to use the BOF to look for similar images. The CBIR approach returns image category for query image by matching identical features between the query and the BOF formed from the template database.

3.1.3 The Performance Evaluation and Results

The training set comprises 70% of the dataset, whereas the test set contains the remaining 30%. This work describes the implementation of the CBIR technique using MATLAB (R2020a) on a system setup with a 2.50 GHz Intel i7 processor having RAM of 8 GB.

A. Algorithm and Results of Implementation

The CBIR approach is implemented using the following algorithm:

Step 1:

The CBIR approach was implemented on a modest computational machine with just 8 GB of RAM. So, a small set of train and test images was employed, as shown in Table 3.1.

Table 3.1 Train and Test datasets

Categories	**Train Count**	**Test Count**
Cancer	2285	914
Non-cancer	1442	576

Step 2:

The blocks obtained by applying grid step on images are provided to the SURF algorithm for feature extraction. There are total of 1,35,28,625 extracted features.

Step 3:

Maintain 80% of the best features of step 2 in each category, i.e., cancer and non-cancer.

Fig. 3.2 is an example of images used as a train dataset.

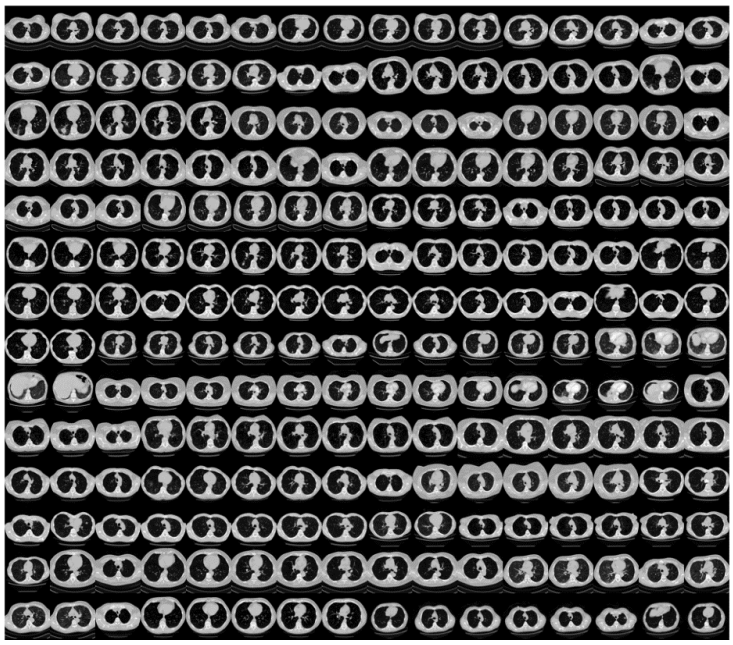


Fig. 3.2 Sample images from train dataset

Step 4:

Applying K-means clustering algorithm on the features extracted by SURF algorithm:

- Total Number of features used by K-means algorithm: 10822900
- Number of clusters (k) used to differentiate the features: 500
- The visual-word frequency histogram is created using the 10822900 features from each image category as shown in Fig. 3.3.

Step 5:

The visual words are thereby used to train SVM for the classification.

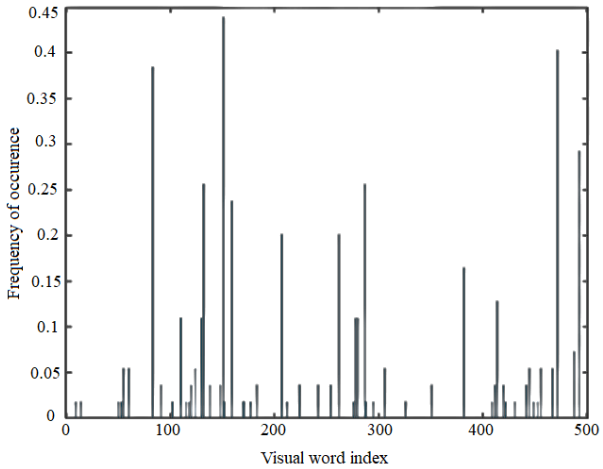


Fig. 3.3 Visual-word frequency histogram

B. Calculations of Results

The different performance metrics used in the performance evaluation of this technique includes sensitivity, specificity, Positive Predictive Value (PPV), Negative Predictive Value (NPV), False Positive Rate (FPR), False Discovery Rate (FDR), False Negative Rate (FNR), accuracy, F1 score and Matthew's Correlation Coefficient (MCC). The calculation of these performance metrics is carried out using the formulas provided in Table 3.2.

Table 3.2 Formulas for Various Performance Metric

Metric	**Formula**
Sensitivity	$TPR = TP/(TP + FN)$
Specificity	$SPC = TN/(TN + FP)$
PPV	$PPV = TP/(TP + FP)$
NPV	$NPV = TN/(TN + FN)$
FPR	$FPR = FP/(FP + TN)$
FDR	$FDR = FP/(FP + TP)$
FNR	$FNR = FN/(FN + TP)$
Accuracy	$ACC = (TP + TN)/(TP + TN + FP + FN)$
F1 Score	$F_1 = 2TP/(2TP + FP + FN)$
MCC	$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}}$

Accuracy is used as an assessment parameter to quantify the efficacy of the proposed model. TP and FP represent correctly classified as cancer and incorrectly classified as cancer respectively. Similarly, TN and FN reflect correctly classified as non-cancer and incorrectly classified as non-cancer respectively.

The results obtained on the 1490 test dataset (cancer-914 and non-cancer-576) is shown in confusion matrix in Table 3.3.

Table 3.3 Results on Test Dataset

		Prediction	
	Known	**Cancer**	**Non-cancer**
Actual	Cancer	0.98	0.02
	Non-cancer	0.01	0.99

The following values are depicted from the confusion matrix obtained on test datasets, as presented in Table 3.3: TP = 0.98, TN = 0.99, FP = 0.01 and FN = 0.02. The performance metrics

provided in Table 3.4 are computed using Table 3.2. According to Table 3.4, the average test accuracy is 98.56%.

Table 3.4 Performance Metrics for Test Dataset

Metric	**Value**
Sensitivity	0.98
Specificity	0.99
PPV	0.9899
NPV	0.9802
FPR	0.01
FDR	0.0101
FNR	0.02
Accuracy	0.9856
F1 Score	0.9849
MCC	0.97

3.1.4 Comparative Evaluation of the Proposed Methods with the Existing Methods

Table 3.5 shows the comparison of performance of the CBIR technique for lung cancer classification with the existing methods based on machine learning methods. In [48], The malignancy of the nodule was divided into five ratings and this reduced the number of nodules used in individual levels. Thereby the accuracy was limited. In [49], the combined approach of hierarchical and k-means clustering both are used in the image retrieval system. The retrieval system proposed clustering based segmentation for diagnoses of the lung cancer. The performance of this method was low because many steps were used for segmentation and that led to loss of nodule features. In [69], filtering technique has been used for pre-processing and SIFT for feature extraction. These extracted features are clustered using K-means clustering. In [74], a hybrid combination of deep learning and fuzzy along with hashing algorithm has been proposed for distributed cloud to retrieve the information. Handling wide range of features is the limitation of this work.

In proposed work, the CBIR based BOF technique has been used for the extraction of features and their classification on the dataset of LIDC-IDRI. Moreover, SURF has been used as a feature descriptor to represent the local features into numerical vector. Compared to other feature extraction algorithms, the error rate observed for SURF algorithm was lesser. In this work the number of features in feature selection process were reduced to minimize the loss and computation complexity.

The highest accuracy attained in the existing approaches is 97.56%, as shown in Table 3.5, while the proposed method yields an accuracy of 98.56%. The high performance can be because of small set of train and test images which are from same CT scan source are used in this work. Another reason can be combination of two best machine learning approaches namely SURF, K-means and SVM are used in this technique.

While this method relies on machine learning algorithm for feature extraction as well as classification, there can be challenge in learning distinctive features. Another drawback of the proposed approach is that the input train data were full CT images from which the features were learnt and thereby classified, but abnormality being very tiny portion in the whole image which can lead to inaccurate classification. So, there is still room for a better method and large dataset with effective receptive area on which model can be trained for lung cancer classification.

Table 3.5 Performance Comparison of Existing Work with Presented work

Author	**Accuracy**	**Dataset**	**No. of CT scans**	**Method**
Wei *et al.* [48]	95.4%	LIDC-IDRI	1018	CAD using CBIR
Malviya *et al.* [49]	67%	LIDC-IDRI	500	CBMIR with hierarchical and K-means clustering
Krishna *et al.* [69]	89%	Rajiv Gandhi Cancer Institute and Research Centre, Delhi	300	Multi-Layered Perceptron Back Propagation Neural Network based on SIFT
Suma [74]	97.56%	KDD Cup 2004 Database	78	Distributed hybrid deep fuzzy hashing algorithm
Bhatt *et al.* [63]	98.56%	LIDC-IDRI	210	CBIR

3.2 Lung Cancer Classification Using the Hybrid Model

The established screening and diagnostic techniques rely on images which varies depending on the CT scanner and other real-world acquisition conditions. There are several algorithms for lung nodule classification have been developed, but there are still issues with consistent performance when working with external datasets. One reason for this failure is because the efficiency of machine learning algorithms for feature extraction is largely dependent on the

visual content's representation. Many classical approaches depend significantly on feature engineering to discover discriminative features, which includes data preparation, modification and hand-crafted feature designs. The lung nodules are extracted using representations like intensity, morphological and texture aspects.

One critical question raised is how to determine the best collection of attributes for encoding lung nodule characteristics. Another issue is that, due to the complexity and variability of the data, such low-level features may not accurately represent the information in the image or capture the underlying statistical properties. Current uses of machine learning techniques show a shortcoming of such approaches, as they are unable to learn representation and discriminative information from raw data in an adaptable manner. This limitation restricts their capacity to generalize to diverse tasks and use of heterogeneous datasets. Thus, in order to overcome the challenges, there are so many opportunities to adapt and improve current techniques to learn complex hierarchical abstractions and representations for lung nodules in a more data-driven manner. Based on this idea a hybrid model is implemented for lung cancer classification in this work. The feature extraction accomplished by Residual Network 50 (ResNet-50) and SVM is used as classifier.

The following sub-section explains the details about the three steps involved for training the Hybrid Model. These steps are pulmonary nodule extraction using pre-processing, image feature extraction using ResNet-50 and feature classification using SVM.

3.2.1 Dataset

The proposed Hybrid Model uses LUNA16 dataset which is a subset of LIDC-IDRI and it is provided with annotations of each nodule and non-nodule in each scan. These annotations have information of the location and diameter of the candidate which helps to extract nodules. Nodules annotated by less than three doctors were deemed unimportant, but they give valuable cancer information, so they are included in this work. The information about such nodules is accessible in the LUNA16 candidates (v2) label set.

3.2.2 System Architecture

The architecture of the hybrid model is shown in Fig. 3.4, it consists of three steps: pulmonary nodule extraction using pre-processing methods, feature extraction using ResNet-50 and classification using SVM.

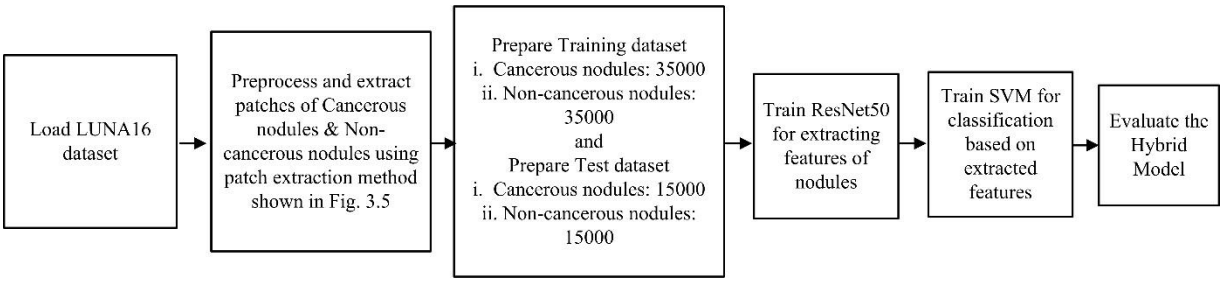


Fig. 3.4 Architecture of the Hybrid Model

The procedure described in sub-section *A* is used to extract nodules and non-nodules. Because the implementation is done on a low computing system with 8 GB RAM, the feature extraction and classification procedure are carried out on patches rather than full CT images.

A total of 100000 nodules and non-nodules each are extracted from LUNA16 datasets and fed as input to train the ResNet-50. The features are extracted from the ResNet-50's final layer, which is the fully connected layer. The SVM performs classification using the extracted features. Further this hybrid model is used for evaluation on test data.

A. Extraction of Pulmonary Nodules

In this pre-processing step the extraction of the nodules from the LUNA16 CT scan is executed. The process for extraction of pulmonary nodules is shown by block diagram in Fig. 3.5. As the CT scans are formed by different settings of CT scanners, they are rendered homogeneous by pre-processing the scans. To do this, all CT images were rescaled with a target voxel of 1mm and resized using an interpolation operation. The raw data is in form of Hounsfield units (HU), which are used to characterize radio density. The HU values are different for tissues among people which will aid for lung segmentation. Because of the biological structure of the lung, the CT image includes other tissues and organs along with the lung. These will create a challenge during detection and the solution is extracting the mask of lung using HU values while ignoring the rest which create challenge in the detection stage. In order to convert the image into lung

mask the processes of conversion of HU values to UINT8 using intensity normalization is performed.

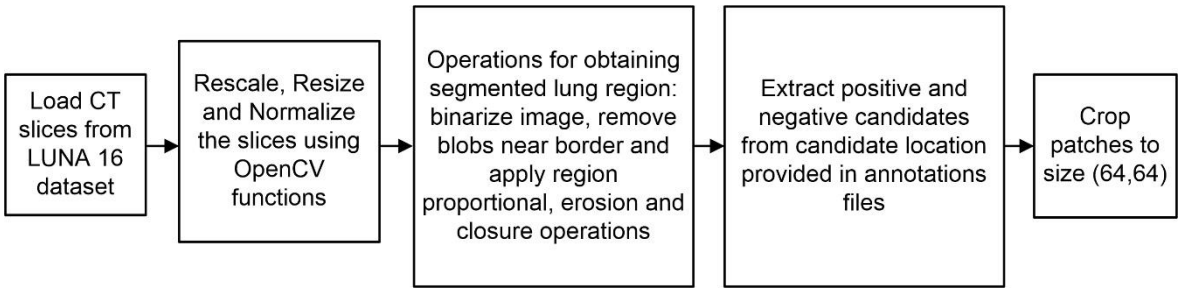


Fig. 3.5 Processing steps for extraction of pulmonary nodules from CT scans

The segmentation process starts with binarizing the image at threshold value of −400 on which the subsequent steps are executed. Blobs connected to the lung's boundary are eliminated by clearing them and the largest area is sorted using a region proportional operation. Because of the morphological architecture of the lung, nodules are often attached to the lungs' edges, vessels and airways. These voxels are eliminated using a morphological erosion operation using a spherical structure with a radius of 2 mm. In order to retrieve the pleural nodules that are attached to the lung walls, a morphological dilation operation using same structure as employed in the erosion operation is performed on a radius of 10 mm. Finally, positive and negative candidates are extracted using the following annotation files: annotations.csv, annotations_excluded.csv and candidates_V2.csv. A total of 100000 cancerous nodules and non-cancerous nodules patches of size 64x64 are obtained and provided as inputs to train the ResNet-50 for feature extraction.

Fig. 3.6 depicts some of the patches that exhibit heterogeneity in positive and negative categories and may be the source of false predictions, posing a challenge for improvement in performance.

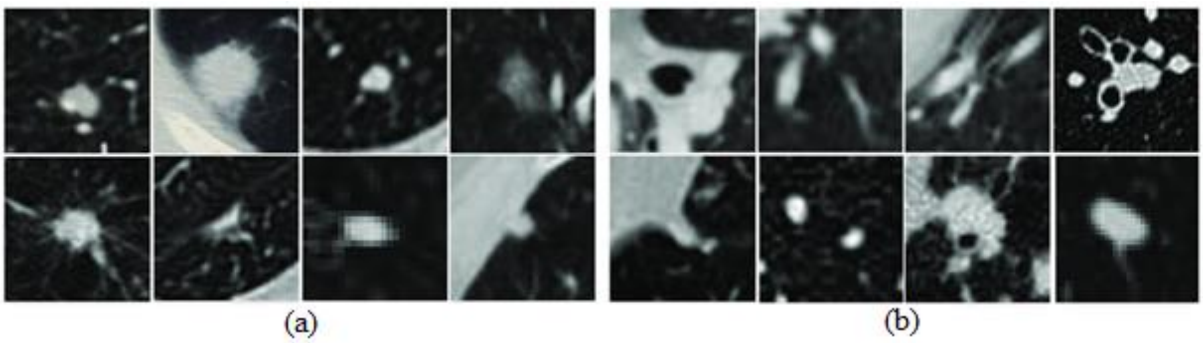


Fig. 3.6 (a) Positive Pulmonary nodules (b) Negative Pulmonary nodules

B. Pulmonary Nodules Feature Extraction Using ResNet-50

Deep convolutional neural networks have resulted in several image classification advancements. Deep networks naturally incorporate features at low/mid/high levels with classifier in an end-to-end multilayer way and the numbers of stacked layers may enrich the level of features. Recent research [140, 162] indicates that network depth is critical and the top ones on the difficult ImageNet dataset all use extremely deep models, with depths ranging from sixteen to thirty.

However, as the network becomes deeper, there is a well-known issue of vanishing gradient problem, which impedes convergence from the start. However, normalized initialization and intermediate normalization layers have substantially solved this issue, allowing networks with tens of layers to begin converge with Stochastic Gradient Descent (SGD) during backpropagation.

When extremely deep networks begin to converge, a degradation problem emerges as network depth increases the accuracy becomes saturated and quickly deteriorate. Surprisingly, such deterioration isn't caused by overfitting, but because of additional layers there is increase in training error. There is a remedy for this issue by construction for the deeper model the new levels are identity mapping while the remaining layers are copies of the learnt shallower model. The presence of this built solution implies that a deeper model should create no more training error than the shallower equivalent. This new network was named as ResNet.

In this work, the ResNet-50 architectures enables to train incredibly deep neural networks with 50 layers. The fundamental principle behind ResNet-50 is that each layer of the architecture learns from residual functions by referring to its own input layer. The design is so easily optimized and obtains considerable accuracy.

The most important component of ResNet-50 is the residual building block (RBB). RBB is built on the concept of exploiting shortcut connections to skip blocks of convolutional layers. These techniques are important for optimizing trainable parameters in error backpropagation to prevent the vanishing gradients issue, which may aid in the construction of deeper CNN structures to enhance final performance for lung cancer diagnosis. RBB is composed of several convolutional layers (Conv), batch normalizations (BN), the Rectified Linear Unit (ReLU) activation function and one shortcut.

In this work, pre-trained ResNet-50 is used to extract features from patches. The basic architectures of ResNet-50 given in Fig. 3.7. The architecture of the ResNet-50 is divided into

four stages. For initial convolution and max-pooling, kernel sizes of 7×7 and 3×3 were employed, respectively. The network then starts with stage 1 having three residual blocks, each with three layers. For convolution, all three layers of stage 1's block employ kernel sizes of 64, 64 and 256. Stage 2 includes four residual blocks each with three layers with kernel sizes of 128, 128 and 512. Following then, Stage 3 of the network has 6 residual blocks comprising three layers each. The kernel sizes in all three layers of the stage 3 block are 256, 256 and 1024. Stage 4 includes three residual blocks, each with three layers having kernel sizes of 512, 512 and 2048. Features were retrieved from the fully connected layer with 1000 neurons of ResNet-50 and provided to SVM for classification.

Name of Layer	Output Size	50-layer
Conv1	112×112	7×7, 64, stride 2
Conv2_X	56×56	3×3 max pool, stride 2
		$\begin{bmatrix} 1\times1,64 \\ 3\times3,64 \\ 1\times1,256 \end{bmatrix} \times 3$
Conv3_X	28×28	$\begin{bmatrix} 1\times1,128 \\ 3\times3,128 \\ 1\times1,512 \end{bmatrix} \times 4$
Conv4_X	14×14	$\begin{bmatrix} 1\times1,256 \\ 3\times3,256 \\ 1\times1,1024 \end{bmatrix} \times 6$
Conv5_X	7×7	$\begin{bmatrix} 1\times1,512 \\ 3\times3,512 \\ 1\times1,2048 \end{bmatrix} \times 3$
	1×1	Average pool, 1000-d fc

Fig. 3.7 Architecture of ResNet-50

C. Support Vector Machines

Data classification is a typical function in machine learning. Assume two classes of data points are supplied and the aim is to determine which category a new data point will be in. A data point is represented as a multi-dimensional vector which can be separated using an n-dimensional hyperplane. This is known as a linear classifier. There are several hyperplanes that might be used to categories the data. One possible candidate for best hyperplane is the one with the greatest isolation or margin between the two categories. The hyperplane is chosen so that the distance between it and the closest data point on either side is as short as possible. If this type of hyperplane exists, it is referred to as a maximum-margin hyperplane and the linear classifier

it creates is referred to as a maximum-margin classifier or alternatively the perceptron of optimum stability.

A support vector machine, in more technical terms, creates a hyperplane or group of hyperplanes in a high or infinite-dimensional space that can be employed for classification, regression or other tasks such as outlier identification. Intuitively, a decent separation is obtained by the hyperplane with the greatest distance to the closest training-data point of any since the bigger the margin, the smaller the classifier's generalization error.

3.2.3 The Performance Evaluation and Results

A. Details of Implementation

The algorithm has been trained on the LUNA16 dataset which is a subset of LIDC-IDRI dataset. The training set comprises 70% of the dataset, whereas the test set contains the remaining 30%. The hybrid model is implemented using MATLAB (R2020a) on a system with a 2.50 GHz Intel i7 processor having RAM of 8 GB.

B. Datasets

The details of train and test dataset for two categories shown in Table 3.6 and Table 3.7 respectively.

Table 3.6 Train dataset

Categories	**Count**
Cancerous Nodule	35000
Non-Cancerous Nodule	35000

Table 3.7 Test dataset

Categories	**Count**
Cancerous Nodule	15000
Non-Cancerous Nodule	15000

C. Calculations of Results

The confusion metrics shown in Table 3.8 is computed based on the outcomes of the test dataset.

Table 3.8 Results on Test Dataset

Known	Cancerous Nodule	Non-Cancerous Nodule
Cancerous Nodule	0.9844	0.0156
Non-Cancerous Nodule	0.0337	0.9663

The following parameters are observed from the confusion matrix based on the test datasets provided in Table 3.8: TP = 0.9844, TN = 0.9663, FP = 0.0337 and FN = 0.0156.

Using Table 3.2, the following performance metrics are calculated as shown in Table 3.9.

Table 3.9 Performance Metrics on Test Dataset

Metric	Value
Sensitivity	0.9844
Specificity	0.9663
PPV	0.9668
NPV	0.9871
FPR	0.0337
FDR	0.0331
FNR	0.0156
Accuracy	0.9753
F1 Score	0.9755
MCC	0.9492

It can be seen from the Table 3.9 that the sensitivity, specificity and accuracy are 98.44%, 96.63% and 97.53% respectively.

3.2.4 Comparative Evaluation of the Proposed Methods with the Existing Methods

The outcomes of the proposed method are compared with those of the existing methods based on SVM and CNN approaches. This comparison is shown in Table 3.10. It can be seen from Table 3.10 that the highest accuracy obtained in the existing method was 95.16% for method which works on SVM for classification while the proposed hybrid model achieves accuracy 97.53%. The reason for achieving high performance can be because the nodules are used for feature extraction instead of relying of full images. This leads to effective receptive field being provided to model to learn features in a better way. Another reason can be the ResNet-50 which is able to extract both semantic and fine-grained features effectively as compared to machine learning methods and simple CNN networks.

Table 3.10 Performance Comparison Between the Existing Work and the Hybrid Model

Author	Accuracy	Dataset	No. of CT scans	Method
Kaucha *et al.* [81]	95.16%	LIDC-IDRI	1018	SVM Classifier
Makaju *et al.* [82]	86.6%.	LIDC-IDRI	1018	Watershed segmentation for detection and SVM for classification
Bhatt *et al.* [86]	97.53%	LUNA16 (Subset of LIDC-IDRI)	888	ResNet-50 + SVM

3.3 Discussion

The main objective of this chapter was to propose a simple and efficient approach for searching and retrieving similarity in the lung CT images from the large medical image dataset. For lung cancer classification, two distinct approaches have been suggested. In the first approach, a CBIR-based BOF strategy has been proposed for the classification of lung cancer. The method speeds up the procedure since precise segmentation of the lung is not required. The graphical patterns were represented using SURF-derived features and K-means. In comparison to other clustering techniques, the combined method offers the accurate and proficient image retrieval system. With 210 lung CT scan images, MATLAB (2020a) is used to implement the CBIR-based BOF approach, which offers accuracy of 98.56%. The major challenge in the development of CBIR methods is extracting effective feature representation from the pixel-level information and associating them with meaningful features. The CBIR community has shown interest in deep CNN model's capacity to learn rich features at various levels of abstraction.

In second method, the ResNet-50 is used as feature extractors to analyze patches extracted from the LUNA16 dataset. SVM was employed to categorize these deeply extracted features. MATLAB (2020a) was used to implement this model utilizing a total of 888 lung CT scan images. The accuracy achieved in this method is 97.53% which is low as compared to first proposed method. The reason for the low performance is because the first method employs small set of data and so it achieved better performance. While on using large dataset in second proposed method there is small decrement in the performance. But still the performance of the hybrid model is not reduced to a large extent and so it can be considered as a better approach among the two proposed methods in this chapter.

The approaches proposed in this chapter involves separate steps for feature extraction and their classification. So, in order to implement an end-to-end model involving a single step, a better approach is implemented in next chapter which consists of various deep neural networks

for lung cancer classification. These DNNs along with the proposed network and their implementation are described in the chapter 4.

CHAPTER-4
Lung Cancer Classification Using Deep Neural Networks

Traditional machine learning algorithms are restricted in their capacity to interpret natural data in its raw form, hence feature engineering and domain expertise are required to identify and extract meaningful values from raw information into learnable representations. In recent years, deep learning has developed as a potential alternative to generate ad-hoc descriptors that can acquire a representation of data from the raw data itself.

This chapter is divided into following sections: overview of Deep Neural Networks (DNNs), datasets used to train the networks, implementation of DNNs, comparison of implemented networks with the existing work and finally the discussion about the work carried out in this chapter.

4.1 Overview of Deep Neural Networks

Deep learning techniques have considerably advanced the state-of-the-art in many fields, including computer vision [87, 88], natural language processing [89], signal and speech recognition [90-92] and object recognition [93, 94]. Thus, deep learning techniques have been implemented for various detection and classification tasks.

Deep learning is an emerging subfield of machine learning in which numerous layers are used to continually extract high-level features from raw data. The concept of deep learning originates from Artificial Neural Networks (ANN). The notion of the biological neural network of the human brain inspires ANNs. Multiple artificial neurons will create a layer inside an ANN (or level). Each neural connection is allocated a certain weight. The neurons on one level are linked in a variety of ways to those on the next level, forming a multilevel representation or network that is completely coupled. Input, hidden and output layers make up the majority of a neural network architecture (each layer contains several neurons). An ANN may include several hidden layers. There are various challenges associated with the use of ANNs. Each successive layer is linked; hence, the number of weights will rise fast as the number of layers increases,

which will impact the learning rate. This issue may be resolved for image analysis by applying many convolution filters on the input image and subsampling the space of filter activation until sufficient high-level features are obtained. CNN are ANNs that use convolution filters; they are presently a very successful method for image classification and recognition applications. These filters are known as convolutional filters and such network is known as CNN. CNNs are a kind of neural network intended to learn convolution parameters from a data set during training. Typically, they consist of several layers, such as convolutional layers, deconvolutional layers, pooling layers, etc. In recent years, many designs have been suggested to enhance CNN's performance and overcome some of its limitations.

This study employs seven deep learning approaches for the classification of lung cancer. There are seven implemented architectures for lung cancer classification in this chapter include MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and VGG-19. These networks are derived from CNN. Based on a comparison of the performance of these networks, a new network is presented in this chapter.

4.2 Datasets Used to Train the Deep Neural Networks for Lung Cancer Classification

In this chapter, patches retrieved from the LUNA16 and DSB2017 datasets are used to train eight DNNs to learn the discriminative abilities of nodules. The candidate's immediate environment is present surrounding the nodule patches is also known as the receptive region plays a crucial role to visualize data and thereby increases the model's capacity.

The steps outlined in Fig. 4.1 are used to obtain the nodules from the DSB2017 datasets. The nodules from the DSB2017 CT scan were extracted using procedures that were similar to those utilized for nodule extractions from LUNA16 datasets described in chapter 3 section 3.2.1. Annotations of each cancerous nodules and non-cancerous nodules for each scan are given with the DSB2017 dataset. These markings provide information on the candidate's location and diameter which aids in the extraction of nodules.

Pre-processing of the CT scans is required to make them homogenous as they are generated by varied CT scanner settings. All CT images were resized using an interpolation process with a target voxel size of 1mm. The procedure of converting HU data to UINT16 utilizing intensity

normalization and thereby binary thresholding is used to transform the image into a lung mask. There are .csv files provided which comprises of locations and diameter, which are then used to extract positive and negative candidates. A total of 1,30,000 patches of 64×64 size, including cancerous nodules and non-cancerous nodules are extracted. Training set contains 70% of the patches whereas a validation set has remaining 30%. The number of datasets utilized to train and validate the neural networks is shown in Tables 4.1 and 4.2 respectively.

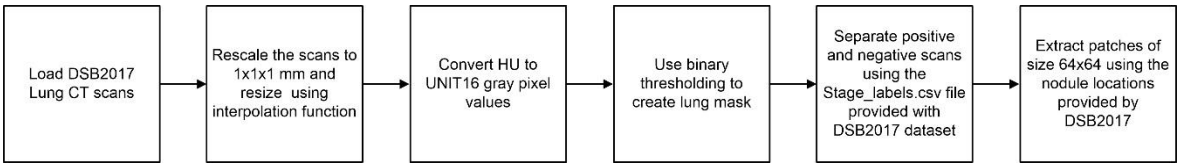


Fig. 4.1 Processing steps for extracting the pulmonary nodules from DSB2017 CT scans

Table 4.1 Train dataset

Categories	**Count**
Cancer	45,000
Non-Cancer	45,000

Table 4.2 Validation dataset

Categories	**Count**
Cancer	20,000
Non-Cancer	20,000

4.3 Implementation of Deep Neural Networks for Lung Cancer Classification

The deep networks employed in this study are designed for supervised learning, which is the process of inferring information from labelled training data. To begin developing an image classification system labelled images are gathered. The algorithm will learn the attributes of each image and then provide two classification scores, one for each category. Tune the system during training such that it provides the greatest classification score when it correctly categorizes the image. This training is carried out by iteratively minimizing a loss function in order to update the weights of the deep learning system using Adam optimization and backpropagation.

The overall process for training the classifiers is shown in Fig. 4.2. The Luna16 and DSB2017 datasets are divided into two categories: cancer and non-cancer. Each category is organized into

two sets: training and validation with a 70% and 30% respectively. Because the images are being fed into the pre-trained network, they must be resized to 224×224 normalized for each channel by removing its average value then divide from the standard deviation. The 224×224 represents the size of ImageNet images and hence what the model anticipates. Images greater than this will be clipped, while images smaller than this will be interpolated.

Various strategies are used in this chapter to enhance the data which are used to train the DNNs. These procedures are carried out with the help of image transform operations, which prepares the data for the neural network. The augmentation of datasets is done to improve the performance of the models. Train data augmentation techniques include resize, crop, flip and rotate. A different random transformation is applied to each epoch, so the network efficiently sees many distinct variants of the same image. Before normalizing, all of the data is converted to Torch tensors. The validation data is only resized and normalized.

In this work data loaders are employed during training to prevent putting the entire data in memory at once. Initially, dataset objects are constructed from the image directories and then sent to a data loader. The data loaders will import the images from disc during training, perform the transformations and generate a batch. Iteration across all batches is performed in the relevant data loaders to train and validate. The significant aspect of data shuffle is performed before transferring it to the network. This implies that the image category ordering varies with each epoch. The Random transformations will be distinct on each epoch, so the network will basically view numerous versions of each training image. There are eight models developed in Keras libraries are trained for lung cancer classification.

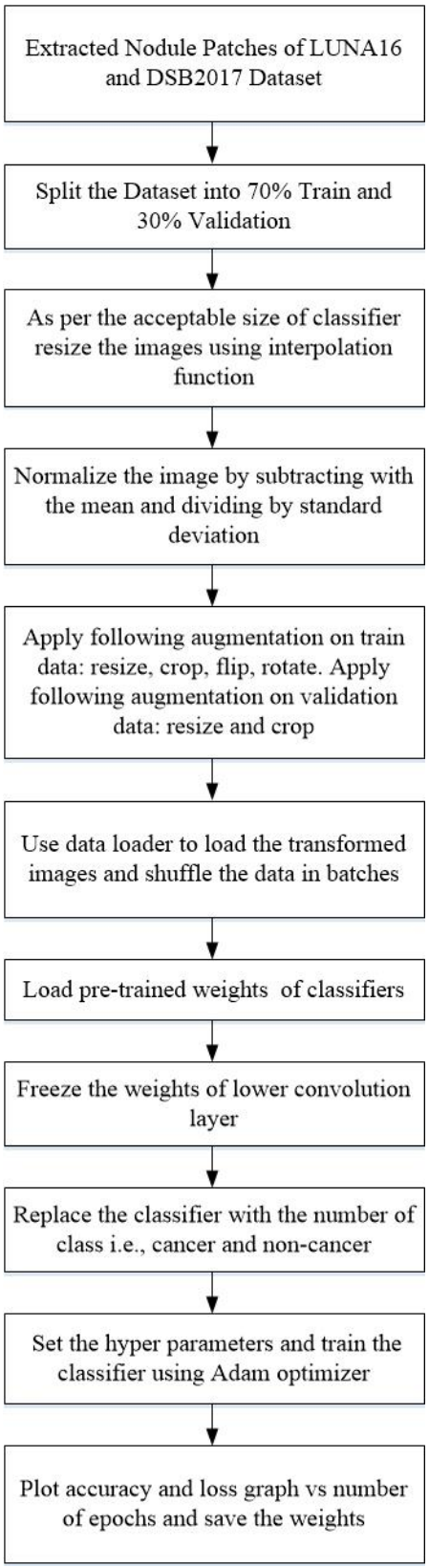


Fig. 4.2 Steps for training the DNNs

All of the pre-trained classifiers discussed in this chapter have previously been trained on millions of images from 1000 different categories. The purpose behind pre-training is the initial convolutional layers of a CNN retrieve features that are useful for various image recognition tasks. By learning higher-level features, fully connected layers specialize to the specific dataset. So, prior to training these pre-trained models, their early layers must be frozen and the classification layer must be replaced based on the number of classes used in this study. Pre-trained networks have shown to be relatively effective for a range of tasks, resulting in a large decrease in training time and in most cases, an improvement in performance. The fine tuning of networks is carried out by setting the epoch value. An epoch is one full run through the training data and classifiers are trained for a predetermined number of epochs or till early stopping is initiated. After each batch, the loss is computed, followed by the loss gradient with respect to the model parameters. Differentiation and backpropagation are used to compute gradients.

In the last layer of the model, the negative log likelihood predicts log probabilities using the SoftMax function. The model parameters are to be optimized by the Adam optimizer. Loss maintains track of both the loss and its gradients with regard to the model parameters.

Monitoring accuracy is computed for each batch and when the training sequence is complete, the validation is initiated and in this process the validation loss is calculated which will be utilized for early stopping.

Training will be terminated if the validation loss has not reduced for a certain number of epochs. When the validation loss decreases, the model weights are preserved so that the optimal model may be loaded in the future. Early stopping is an useful approach for preventing data overfitting. If ongoing training is performed, the training loss will continue to fall, however the validation loss will rise as the model begins to remember the training data. Early stopping avoids this from occurring and if the model is stored during every epoch when the validation loss lowers, it is feasible to retrieve the model with the greatest performance on the validation data. Test data are utilized to get final evaluation of the model. Thus, the model provides its probability and class for a certain image after making its predictions.

To develop an end-to-end model for lung cancer classification, seven popular DNNs among research community were trained using the LUNA16 and DSB2017 datasets. These DNNs includes: MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and

VGG-19. To quantify the efficacy of the model, accuracy is used as an assessment measure. The accuracy is determined according to Eq. (4.1).

$$\text{Accuracy} = (TP + TN)/ (TP + TN + FP + FN) \quad (4.1)$$

where TP and FP denote correctly classified cancer nodules and incorrectly classified cancer nodules respectively. Likewise, TN and FN reflect correctly represented non-cancer nodules and incorrectly classified non-cancer nodules respectively.

Finally, a new model is developed based on the performance and obstacles encountered by these seven models. This suggested model is named as Simplified VGG (SVGG) which achieved the average train accuracy of 99.40% and average validation accuracy of 96.26%.

The networks were built in Python using Keras deep learning library. To train the networks, the weights were randomly assigned using the Gaussian distribution and modified using standard backpropagation. Adam optimizer is used to optimize the model on batch size of 8. The loss was reduced using the categorical cross-entropy loss. Non-linearity is provided via the ReLU activation function. To improve the model's convergence rate and generalization capabilities, the learning rate is set to 0.01 and the dropout rate was set to 0.4 and 0.5 in some models. All of the networks were trained on an NVIDIA GeForce RTX 2070 GPU for 40 epochs with early stopping set to 20.

4.3.1 MobileNet

The conventional convolution process filtering and merging data based on the convolutional kernels to build a new representation. For a significant decrease in computational cost, the filtering and combining phases may be divided into two parts using factorised convolutions termed as DepthWise Separable Convolutions (DWSC).

When compared to networks with normal convolutions of the same depth, MobileNet utilizes DWSC, which minimizes the number of parameters. As a consequence, lightweight deep neural network is created.

A. Architecture of MobileNet

MobileNet's structure is composed of DWSC, with the exception in the first layer, which is a standard convolution. Fig. 4.3 depicts the MobileNet architecture.

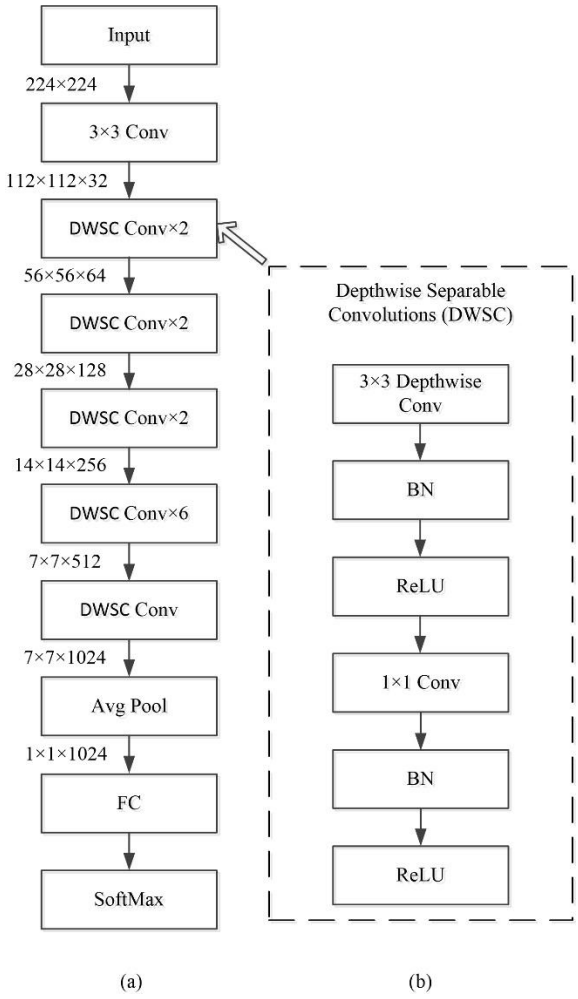


Fig. 4.3 Architecture of MobileNet [114]

With the exception of the last fully connected layer, which has no nonlinearity and feeds into a softmax layer for classification, all layers are followed by a batchnorm and ReLU nonlinearity. As seen in Fig. 4.3, the DWSC is shown in expanded form in which it is factorised to layers with depthwise convolution and 1x1 pointwise convolution with batchnorm and ReLU nonlinearity after each convolutional layer. Down sampling is addressed by strided convolution in both the depthwise and first layer convolutions. Before the fully connected layer, the spatial resolution is decreased by a last average pooling. MobileNet has total 28 layers on considering depthwise and pointwise as independent layers.

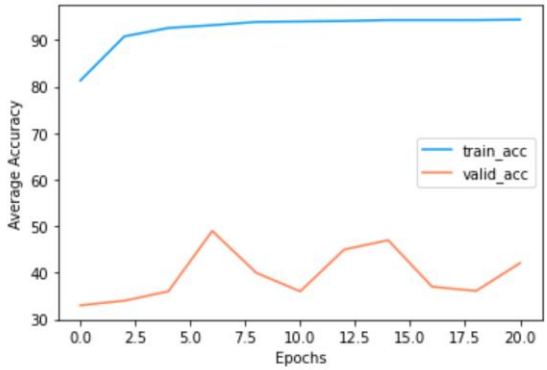
B. Simulation Result of MobileNet

The simulation results of MobileNet are shown in Fig. 4.4. The findings show the graphs of accuracy and loss with regard to the number of epochs. The graph depicts both the training and validation values. The model performs well on the train dataset, with an average accuracy of approximately 92.47%, but performs poorly on the validation dataset, with an average accuracy of around 39.56%. The explanation for the big discrepancy in training and validation performance might be because the model has 28 layers, which is quite deep, hence the model is overfitting and thus delivers very poor validation performance.

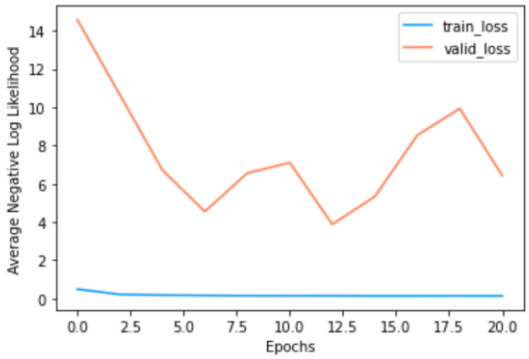
The negative log likelihood loss is calculated using the Eq. (4.2)

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\big(p(y_i)\big) + (1 - y_i) \cdot \log\big(1 - p(y_i)\big) \tag{4.2}$$

where, y is the label (0 and 1 for binary) and p(y) is the predicted probability of the data point being 1 for all N points.



(a)



(b)

Fig. 4.4 (a) Accuracy and (b) Loss for nodule classification using MobileNet

4.3.2 SqueezeNet

The primary goal of developing this model was to find an architecture with few parameters while maintaining competitive accuracy. Because the validation performance in the MobileNet model was poor, so it was necessary to train a network with low computing complexity and a compact network that is not overfitting and gives superior validation performance.

A. Architecture of SqueezeNet

When constructing SqueezeNet architectures, three key approaches have been used to accomplish the model with lower number of parameters as well as better accuracy. The approaches considered are:

Approach 1: Change 3×3 filters to 1×1 filters. Since a 1×1 filter has 9 times fewer parameters than a 3×3 filter.

Approach 2: The total number of parameters in a layer with a kernel size of 3×3 is calculated as follows: (number of input channels) * (number of filters) * (3×3). To keep the total number of parameters in a CNN as low as possible, it is necessary to reduce not only the number of 3×3 filters as per Strategy 1, but also the number of input channels to the 3×3 filters. Using squeeze layers, the number of input channels is reduced to 1×1 filters.

Approach 3: Lately down-sample the network to give the convolution layers huge activation maps. The assumption is that, when all other factors are equal, greater activation maps may result in improved classification accuracy.

The goals of approaches 1 and 2 are to carefully reduce the number of parameters in a CNN while aiming to maintain accuracy. The third approach focuses on increasing accuracy.

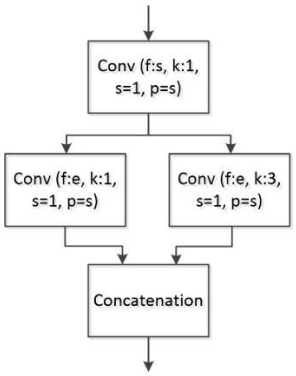


Fig. 4.5 Architecture of the Fire module

The Fire module as shown in the is the Fig. 4.5 is the building component for SqueezeNet architecture that allows to effectively utilize approaches 1, 2 and 3. In Fig. 4.5, f is the number of filters, s means stride, k means kernel size and p means padding. A Fire module consists of a squeeze convolution layer with just 1×1 filters feeding into an expand layer with a combination of 1×1 and 3×3 convolution filters. The extensive deployment of 1×1 filters in Fire modules is an application of approach 1. In a Fire module, three configurable dimensions have been exposed: $s1 \times 1$, $e1 \times 1$ and $e3 \times 3$. The number of filters in the squeeze layer in a Fire module is $s1 \times 1$, the number of 1×1 filters in the expand layer is $e1 \times 1$ and the number of 3×3 filters in the expand layer is $e3 \times 3$. When Fire modules is employed, it is specified that $s1 \times 1$ to be smaller than $(e1 \times 1 + e3 \times 3)$, which helps the squeeze layer restrict the amount of input channels to the 3×3 filters as per approach 2.

Fig. 4.6 depicts the SqueezeNet architecture, which starts with a single convolution layer, then 8 Fire modules and finally a final convolution layer. From the first to the last Fire module in the network, the number of filters per fire module are steadily increased. After the layers conv1, fire4, fire8 and conv10, the SqueezeNet executes pooling with a stride of 2. These late placements of pooling are in accordance with approach 3 for maintaining the accuracy of the model.

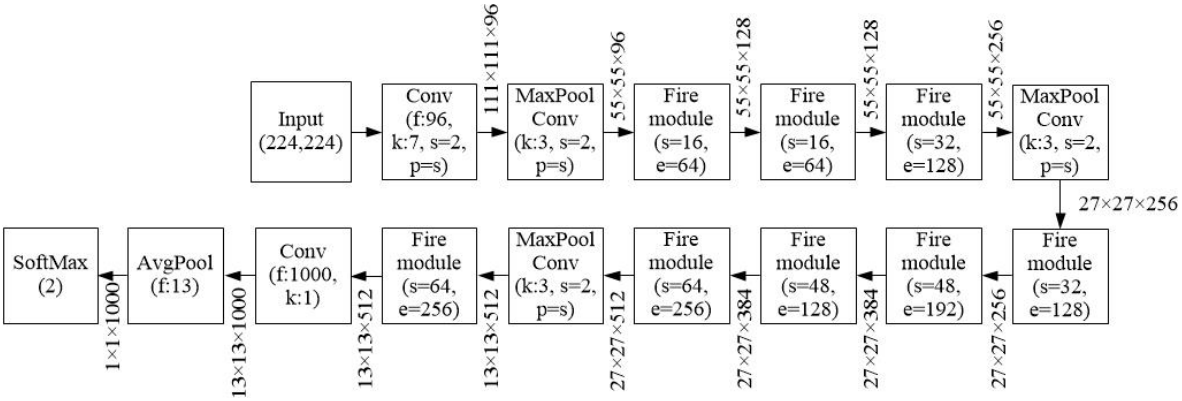
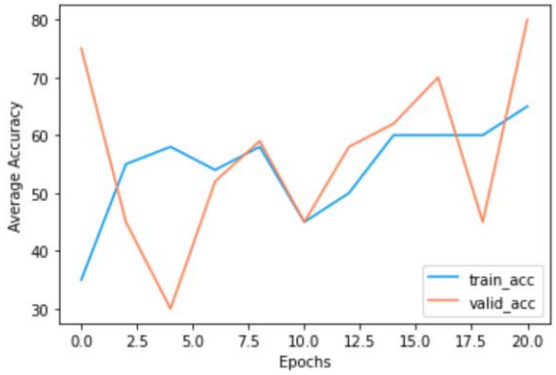


Fig. 4.6 Architecture of SqueezeNet [118]

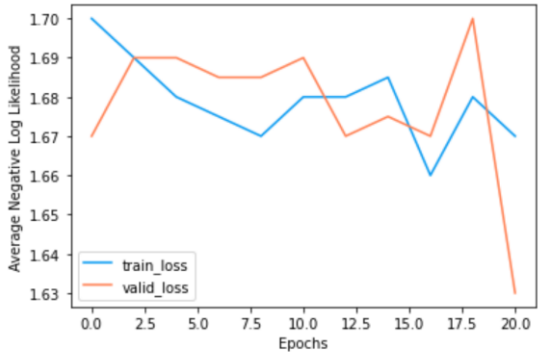
B. Simulation Result of SqueezeNet

Fig. 4.7 depicts performance of SqueezeNet for nodule classification. The SqueezeNet has a 54.54% average train accuracy. The average validation accuracy is about 56.45%, which is better than the prior model, but as it is low it cannot be utilized for clinical approaches. The

reason for the low performance might be because the filters of two sizes 1×1 and 3×3 could not learn mid-level features, resulting in errors in categorization of medium-sized nodules.



(a)



(b)

Fig. 4.7 (a) Accuracy and (b) Loss for nodule classification using SqueezeNet

4.3.3 AlexNet

Overfitting, poor performance and excessive computing cost impacted the previously trained MobileNet model. As a result, SqueezeNet was trained, whose performance was improved but not comparable to be the best model. Thereby AlexNet was trained as it is a simple model with minimal computation time and better performance.

The AlexNet is the first deep CNN architecture which is regarded as having produced ground-breaking results for image classification and recognition applications.

A. Architecture of AlexNet

The AlexNet architecture is shown in Fig. 4.8. The design of the network was fairly similar to LeNet, except it was deeper, had more filters per layer and had stacked convolutional layers. It has filter sizes of 11×11, 5×5 and 3×3 in convolution layers with ReLU activation function.

Prior to AlexNet, the most popular activation functions were sigmoid and tanh. Because these functions are saturated, they suffer from the vanishing gradient issue, making it harder for the network to be trained. AlexNet utilizes the ReLU activation function, which is not affected by the vanishing gradient issue. Due to ReLU's unbounded nature, even if it aids in resolving the vanishing gradient issue, the learnt variables might grow too large. In order to prevent this, Local Response Normalization (LRN) was introduced in AlexNet. The concept of LRN is to normalize a neighbourhood of pixels by simultaneously enhancing the stimulated neuron and damping the surrounding neurons. AlexNet tackles the issue of overfitting by using dropout layers in which neurons are discarded with a probability of p = 0.5 during training.

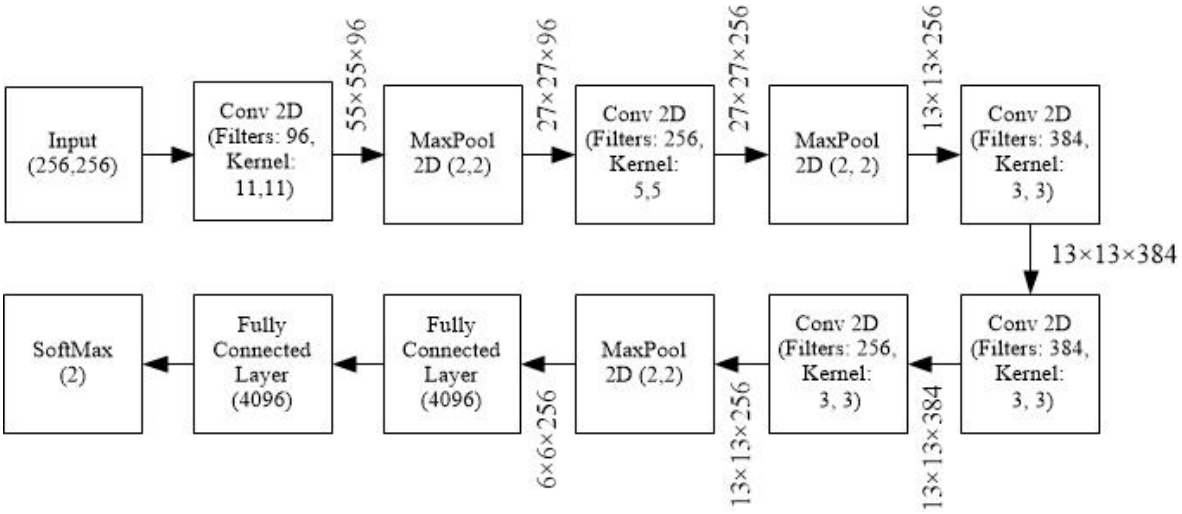
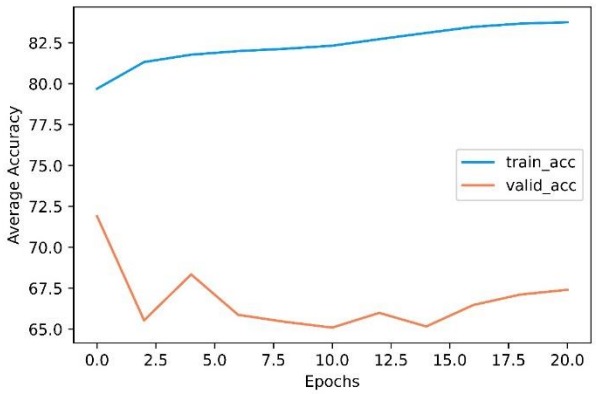


Fig. 4.8 Architecture of AlexNet [93]

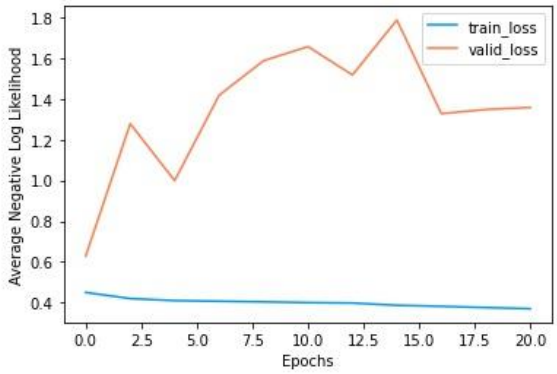
B. Simulation Result of AlexNet

Fig. 4.9 depicts the simulation results of an AlexNet trained for lung cancer classification. The average train accuracy achieved is of 82.35% while average validation accuracy is 66.75% which shows slight improvement as compared to previous models. A relatively tiny

and simplistic neural network might be the cause of the low performance. Thus, the prior trained model's results show that the strategy for achieving low complexity and improving performance is not simply that the model be simple, but that the low, mid and high-level characteristics of tiny, medium and big size nodules be retrieved, which lack in this model as well as in all previously described models.



(a)



(b)

Fig. 4.9 (a) Accuracy and (b) Loss for nodule classification using AlexNet

4.3.4 GoogLeNet

The GoogLeNet architecture's primary goal was to achieve excellent accuracy at a low computing cost. It introduced CNN's new Inception block concept, which uses the split, transform and merge principle to incorporate Multi-Scale Convolutional Transformations (MSCT). With the aid of Inception's principle, a challenge relating to the learning of various

sorts of variations contained in the same category of images with varying resolutions was successfully resolved.

A. Architecture of GoogLeNet

Traditional convolutional layers are changed in tiny blocks in GoogLeNet, comparable to the notion of replacing each layer with micro neural network suggested in Network in Network (NIN) architecture. The Inception network design is made up of many Inception modules with the topology depicted in Fig. 4.10. This block has filters of various sizes $(1 \times 1, 3 \times 3 \; and \; 5 \times 5)$ that collect spatial information at various scales, including fine and coarse grain level. Inception expands the network space from which the best network may be selected via training. Each Inception module may collect key features at various levels. The 5×5 conv layer captures global features, while the 3×3 conv layer captures distributed features. The max-pooling process is responsible for capturing low-level features that stand out in a neighbourhood. Instead of using large size kernels GoogLeNet controls the computations by adding a bottleneck layer of 1×1 convolutional filter. Furthermore, it employed sparse connections to address the issue of duplicated information and lowered cost by removing irrelevant feature-maps. All of these features are extracted and concatenated at a specific level before being sent to the next layer. Moreover, the density of connections was lowered by employing global average pooling at the final layer rather than a fully connected layer.

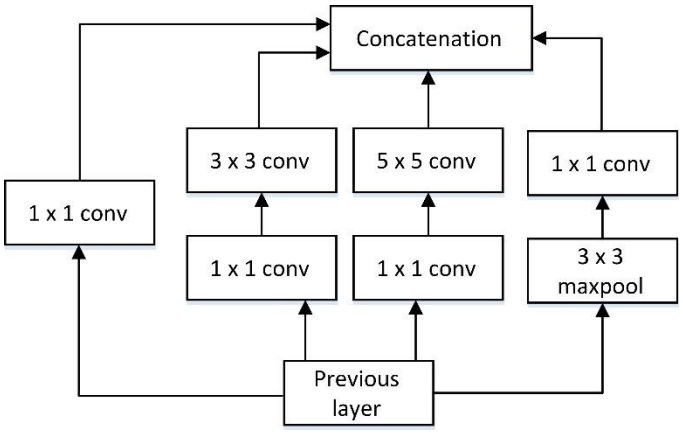


Fig. 4.10 Inception Module [140]

When simply counting layers with parameters, the network has 22 layers in total. Rectified linear activation is used in all convolutions, including those found within the Inception modules.

The network architecture of GoogLeNet is shown in Table 4.3. The numbers "$\#3 \times 3\ reduce$" and "$\#5 \times 5\ reduce$" refer to the number of 1×1 filters used in the reduction layer before to the 3×3 and 5×5 convolutions. The pool proj column shows the number of 1×1 filters in the projection layer following the built-in max-pooling. All of these projection layers make use of rectified linear activation.

Table 4.3 Architecture of GoogLeNet [140]

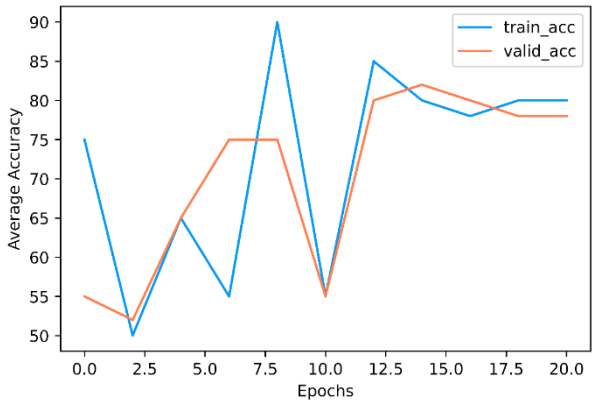
Type	**Patch size/ Stride**	**Output Size**	**Depth**	**#1 × 1**	**#3 × 3 Reduce**	**#3 × 3**	**#5 × 5 Reduce**	**#5 × 5**	**Pool Proj**
convolution	$7 \times 7/2$	$112 \times 112 \times 64$	1						
max pool	$3 \times 3/2$	$56 \times 56 \times 64$	0						
convolution	$3 \times 3/1$	$56 \times 56 \times 192$	2		64	192			
max pool	$3 \times 3/2$	$28 \times 28 \times 192$	0						
Inception (3a)		$28 \times 28 \times 256$	2	64	96	128	16	32	32
Inception (3b)		$28 \times 28 \times 480$	2	128	128	192	32	96	64
max pool	$3 \times 3/2$	$14 \times 14 \times 480$	0						
Inception (4a)		$14 \times 14 \times 512$	2	192	96	208	16	48	64
Inception (4b)		$14 \times 14 \times 512$	2	160	112	224	24	64	64
Inception (4c)		$14 \times 14 \times 512$	2	128	128	256	24	64	64
Inception (4d)		$14 \times 14 \times 528$	2	112	144	288	32	64	64
Inception (4e)		$14 \times 14 \times 832$	2	256	160	320	32	128	128
max pool	$3 \times 3/2$	$7 \times 7 \times 832$	0						
Inception (5a)		$7 \times 7 \times 832$	2	256	160	320	32	128	128
Inception (5b)		$7 \times 7 \times 1024$	2	384	192	384	48	128	128
avg pool	$7 \times 7/1$	$1 \times 1 \times 1024$	0						
dropout (40%)		$1 \times 1 \times 1024$	0						
linear		$1 \times 1 \times 1000$	1						
softmax		$1 \times 1 \times 2$	0						

The ability to effectively transmit gradients back through all the layers was a worry due to the network's very high depth. The solution to this issue is using a relatively shallower networks that perform well on test, it is likely that the intermediate layers of the network would provide highly discriminative features. It is expected that adding auxiliary classifiers linked to these

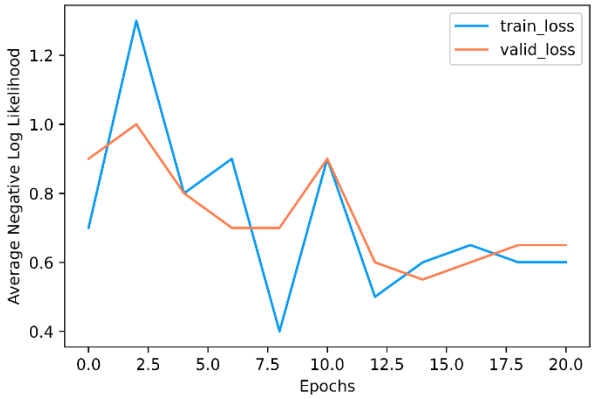
intermediary layers will improve discrimination in the classifier's lower stages, boost the gradient signal that is sent back and provide additional regularization.

B. Simulation Result of GoogLeNet

The simulation results of GoogLeNet are shown in Fig. 4.11. The model attains train and validation average accuracy of 69.81% and 70.45% respectively. GoogLeNet introduced the concept of auxiliary learners to speed up the convergence rate. However, the main drawback of the GoogLeNet was its heterogeneous topology that needs to be customized from module to module. Another limitation of GoogLeNet was a representation bottleneck that drastically reduces the feature space in the next layer and thus sometimes may lead to loss of useful information. These factors may have affected the model performance during training and validation of this network for lung cancer classification.



(a)



(b)

Fig. 4.11 (a) Accuracy and (b) Loss for nodule classification using GoogLeNet

4.3.5 Residual Neural Network-18

Deep CNN have led to numerous breakthroughs in image classification. Deep networks naturally include features at low/medium/high levels with classifier in an end-to-end multilayer manner and the number of stacked layers may increase the depth of features. The studies [140, 162] suggested that network depth is crucial and the leading models on the challenging ImageNet dataset all utilise models with varying depths from sixteen to thirty.

Several more nontrivial visual recognition tasks [180, 169, 213] have profited greatly from very deep models. Adding more layers, however, presents the well-known problem of vanishing gradient problem, which inhibits convergence from the beginning. When deeper networks started to converge, a deterioration problem arises in which, as network depth grows, accuracy gets saturated and degrades rapidly. Interestingly, this is not the result of overfitting; rather, adding additional layers to a sufficiently deep model increases the training error. This problem was solved by introducing ResNet which is regarded as a continuation of deep networks. ResNet transformed the CNN architectural race by proposing the notion of residual learning in CNNs and devising an effective approach for deep network training.

A. Architecture of ResNet-18

The fundamental building component of ResNet is a residual block as illustrated in Fig. 4.12, which is repeated throughout the network.

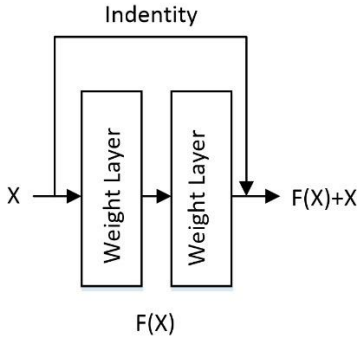


Fig. 4.12 Residual Block

Rather than of learning the mapping from $x \rightarrow F(x)$, the network learns it from $x \rightarrow F(x)+G(x)$. When the input x and output F(x) have the same dimension, the function $G(x) = x$ is an identical function and the shortcut connection is known as the identity connection. Since it is simpler to

push weights to zero than to one during training, the equivalent mapping is acquired by zeroing out the weights in the intermediate layer. The projection connection is used in place of the identity connection when the dimensions of F(x) are different from x.

The ResNet-18 design is made up of many residual blocks with the topology depicted in Fig. 4.13. It comprises of convolution layers with 3x3 filters. Throughout the network, just two pooling layers are employed, one at the starting and the other at the finish.

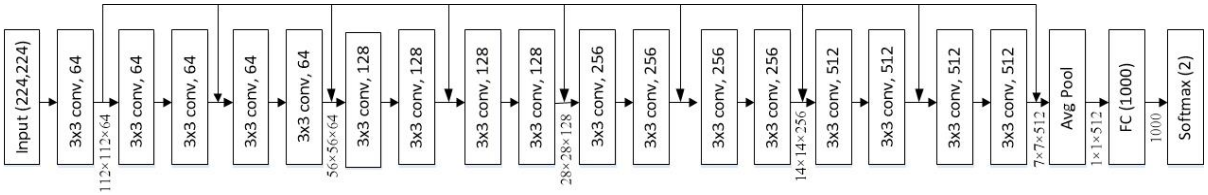
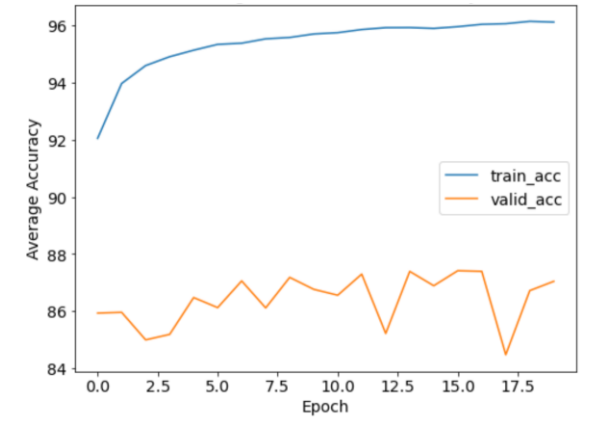


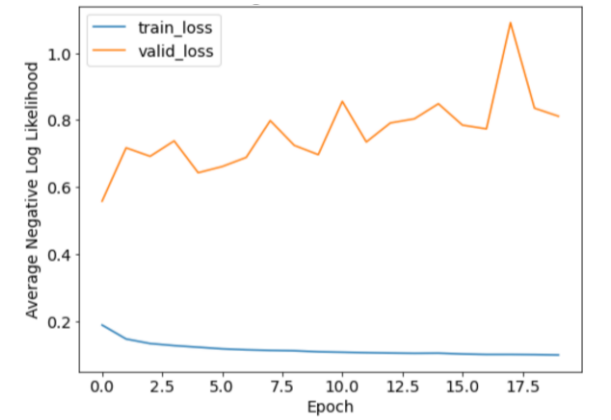
Fig. 4.13 Architecture of ResNet-18 [149]

B. Simulation Result of ResNet-18

The simulation result of ResNet-18 is shown in Fig. 4.14. According to the simulation results, the train and validation average accuracy attained are around 95.32% and 86.50%, respectively. Around 10% disparity between train and validation results is due to the model's huge number of parameters and high complexity which causes the model to be overfitted.



(a)



(b)

Fig. 4.14 (a) Accuracy and (b) Loss for nodule classification using ResNet-18

4.3.6 Visual Geometry Group 16

VGGNet originated from the necessity to minimize the number of parameters in convolution layers while improving the performance in less computational time. The weight configuration of the VGGNet is freely available and has been utilized as a baseline feature extractor in a variety of different applications. VGGNet has many variants i.e., VGG-16, VGG-19 and so on that change solely in the total number of layers in the network.

A. Architecture of VGG-16

VGG-16 has 16 convolutional layers and a highly homogeneous architecture making it quite attractive. It is regarded as one of the best vision model architectures. Fig. 4.15 depicts the VGG-16 architecture used to categorise candidates. The most distinguishing feature of VGG-16 is that instead of a huge number of hyper-parameters, they concentrated on having convolution layers of 3×3 filter with stride 1 and utilized the same padding and maxpool layer of 2×2 filter with stride 2. This configuration of convolution and max pool layers is maintained throughout the design. Finally, it features two Fully Connected layers (fc) and a SoftMax for output.

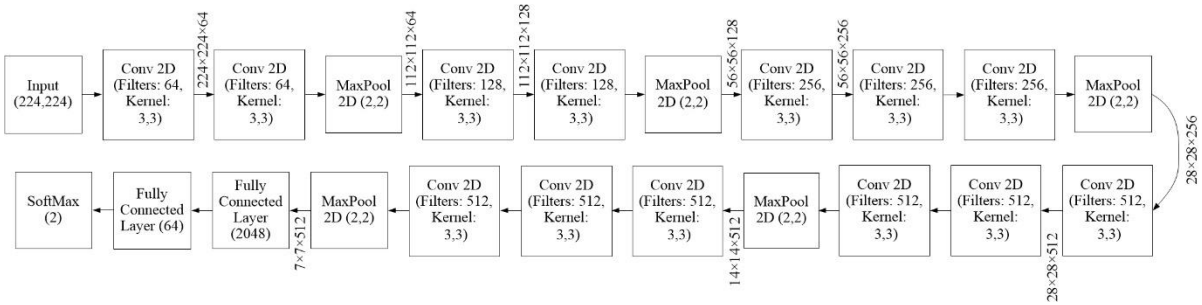
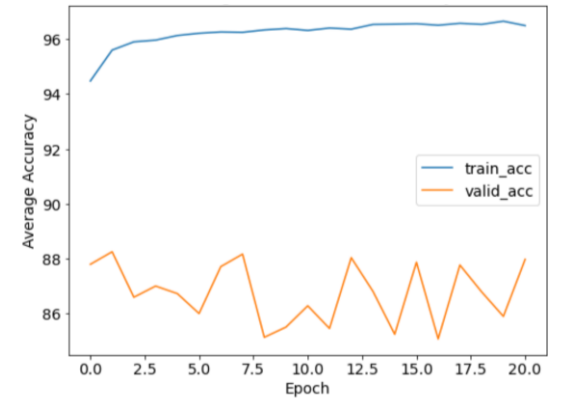


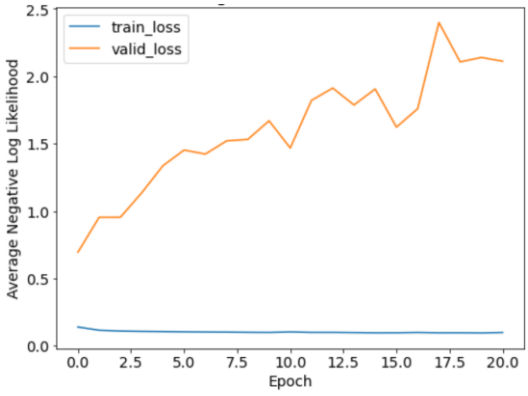
Fig. 4.15 Architecture of VGG-16 [162]

B. Simulation results of VGG-16

Fig. 4.16 depicts the simulation results of VGG-16. The VGG-16 has a train average accuracy of 96.34% and a validation average accuracy of 86.77 %. The findings show a slight improvement in the use of deep neural network. However, VGG-16 includes 138 million parameters, which may be difficult to regulate and need a long learning time.



(a)



(b)

Fig. 4.16 (a) Accuracy and (b) Loss for nodule classification using VGG-16 network

4.3.7 Visual Geometry Group 19

VGG-19 model (also known as VGGNet-19) is similar to the VGG-16 in principle, except that it supports 19 layers. The numbers "16" and "19" represent the number of weight layers in the model. VGG-19 therefore has three additional convolutional layers than VGG-16.

A. Architecture of VGG-19

The VGG-19 model shown in Fig. 4.17 contains 19 weighted layers that are produced by 16 convolutions and 3 fully-connected (fc) layers. The convolutional layers have a tiny kernel size of 3×3 with padding and stride of 1 pixel. The network contains 5 max-pooling layers having a kernel size of 2×2 with a stride of 2. The non-linear function is represented using ReLUs. Following the convolutional layers, there are linear classifiers with three fc layers; the first fc layers contain 4096 features while the second has just 1000. The last fc layer has a SoftMax layer providing the probabilities for classification i.e., cancer and non-cancer.

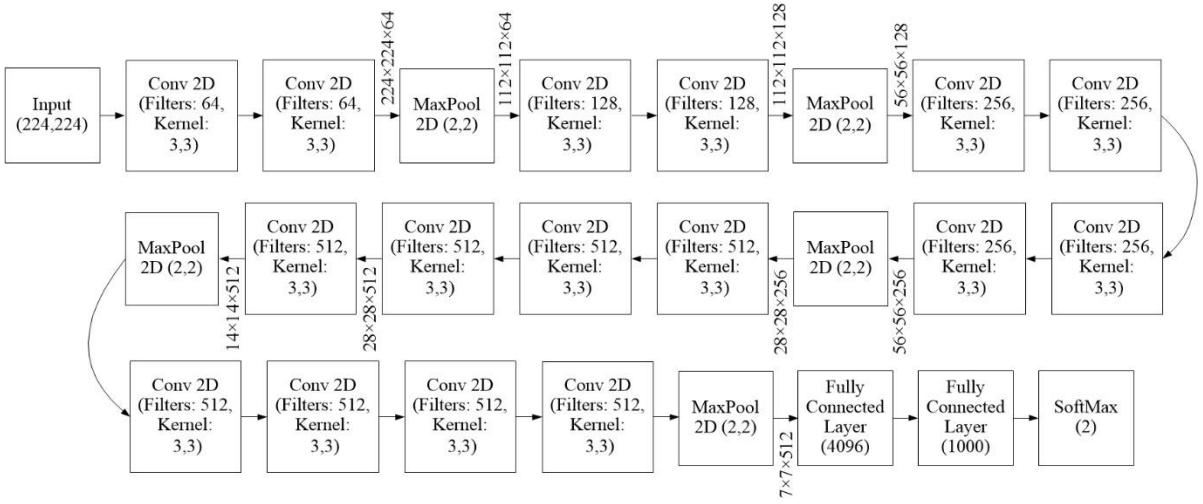
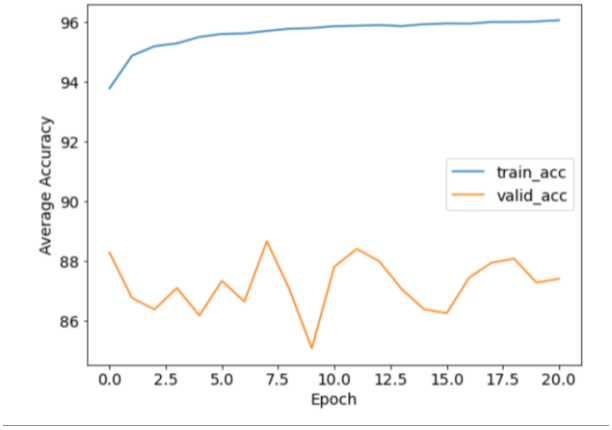


Fig. 4.17 Architecture of VGG-19 [162]

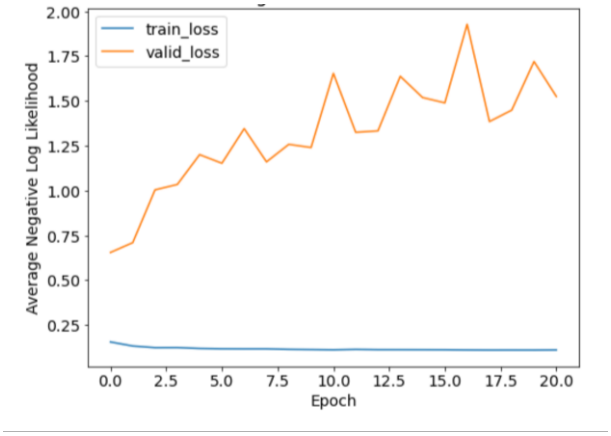
B. Simulation Result of VGG-19

Fig. 4.18 illustrates the simulation result for VGG-19. The train average accuracy of VGG-19 is 95.78%. However, VGG-19 has 143 million parameters, which may be difficult to control and need a long time to train. The validation average accuracy achieved by the VGG-19 is 87.16%. It can be seen that there is still room for development in terms of building neural

networks with fewer parameters to minimize computational complexity and computation time along with performance improvement.



(a)



(b)

Fig. 4.18 (a) Accuracy and (b) Loss for nodule classification using VGG-19 network

4.3.8 Simplified VGG

The VGG-16 includes 138 million parameters and VGG-19 has 143 million parameters, which require a significant amount of processing time and they sometimes overfit. Batch normalization as well as dropout have been employed along with pooling and convolution to overcome over-fitting problem and thereby get high-precision convergence in the proposed network. As the proposed network has few convolution layers leading to less computational time so it is known as Simplified VGG (SVGG).

A. Architecture of SVGG

Deep neural networks include several non-linear hidden layers, making them highly expressive models capable of learning very complex correlations among its inputs and outputs. However, with minimal training data, many of these complex associations will be the consequence of sampling noise, therefore they will be present in the training dataset but not in the actual test data, even if they are derived from the same allocation. This leads to overfitting and there are several ways for preventing this issue. Batch normalization and dropout are utilized in the proposed model to minimise overfitting.

Batch Normalization also improves gradient flow across the network by lowering gradient dependency on the magnitude of the parameters or their starting values. This enables model to use considerably larger learning rates without risk of divergence. Finally, batch normalization enables the usage of saturating nonlinearities by keeping the network from being trapped in saturated modes.

Dropout is another method of dealing with the issue of overfitting. During training, nodes are randomly removed from the neural network. This inhibits units from co-adapting excessively. Dropout samples are drawn from an exponential variety of distinct thinned networks during training. It is simple to approximate the impact of averaging the predictions of all these thinned networks at test time by simply employing a single un-thinned network with reduced weights. This lowers overfitting greatly and provides considerable benefits over other regularization approaches.

Fig. 4.19 shows the architecture of SVGG network. This network was fed a fixed size (64×64) image as input, implying that the matrix was of form (64,64,1). The model used kernels of 3×3 size, which allowed it to capture the whole pictorial representation. The image's spatial resolution was preserved by using spatial padding. Max-pooling of a size 3×3 is applied. This was proceeded by ReLu to incorporate non-linearity into the model to improve classification. The Batch normalization and dropout layers are added between the convolution layers throughout the network in order to reduce overfitting and maintain the features. Two fully connected layers were implemented, the first of which is of size 1024 channels and the second layer is a SoftMax function.

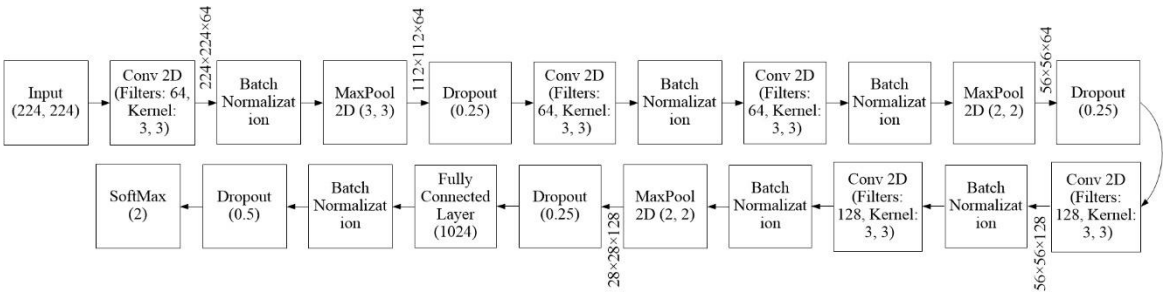
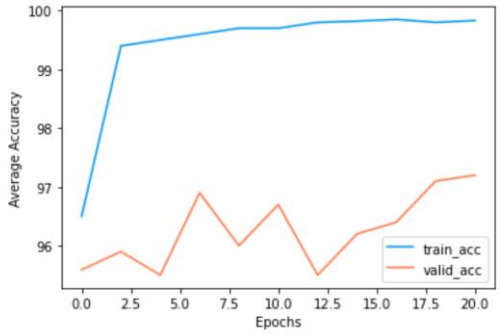


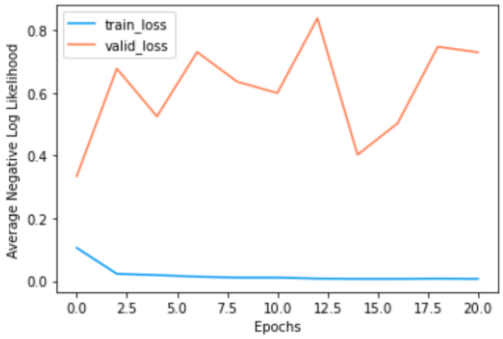
Fig. 4.19 Architecture of SVGG

B. Simulation Result of SVGG

Fig. 4.20 shows the simulation result for lung cancer classification using SVGG and it can be analysed that the train average accuracy achieved using this method is 99.40%. This signifies that as compared to previous models there is significant improvement in validation average accuracy of 96.26%. Thus, it can be seen that the proposed neural network has fewer layers which minimize computational complexity and computation time along with better performance.



(a)



(b)

Fig. 4.20 Accuracy and (b) Loss for nodule classification using SVGG network

4.4 Comparative Evaluation of the Proposed Model with the Existing Models

Table 4.4 provides a comparison of simple network such as AlexNet with a complex network such as MobileNet and GoogLeNet. The comparison shows that, when designing a deep neural network due consideration must be taken with regard to dataset and GPU capability. Among the seven deep neural networks trained the results show that lowest accuracy is 39.56% of MobileNet while SVGG achieved the highest accuracy of 96.26%. It shows on using a more complicated VGG-16 network shows that low accuracy of 86.77 % and which was then improved to 87.16% on using VGG-19. The comparison shows that proposed SVGG network has improvement due to the fact that network is not too simple for learning heterogeneity of nodules nor it's too complex to burden the computing capability and the use of dropout and batch normalization helped in overcoming over-fitting problem.

Table 4.4 Performance of DNNs on LUNA16 and KDSB 2017

Sr. No.	Method of Deep Learning	Average Validation Accuracy
1	MobileNet	39.56%
2	SqueezeNet	56.45%
3	AlexNet	66.75%
4	GoogLeNet	70.45%
5	ResNet-18	86.50%
6	VGG-16	86.77%
7	VGG-19	87.16%
8	SVGG	96.26%

Table 4.5 provides a comparison between the performances of previous work and the proposed SVGG implemented in this work for lung cancer classification. The comparison shows that the SVGG network has 96.26% accuracy which is better than the other DNNs trained for classification. The reason for this can be the model is not too complex as well as the large set of data consisting of LUNA16 and DSB2017 instead of using single dataset for training as in previous work. Another reason can be the use of patches instead of full images to train the networks and this improved the performance by learning the contextual features in a better way.

Table 4.5 Performance Comparison of Existing Work with the Proposed Model

Author	Accuracy	Dataset	No. of CT scans	Method
Ciompi *et al.* [85]	Intensity feature + SVM: 27% Unsupervised features + SVM: 39.9% ConvNet 1 Scale: 78% ConvNet 2 Scale: 79.2% ConvNet 3 Scale: 79.5%	Train: MILD Test: DLCST	Train:943 Test: 468	ML and ConvNet
Shen *et al.* [108]	86.84%	LIDC-IDRI	1010	ML+2D-CNN
Shen *et al.* [112]	87.14%	LIDC-IDRI	1010	3D-CNN
Monkam *et al.* [100]	88.28%	LIDC-IDRI	1010	2D-CNN
Dehmeshki *et al.* [163]	90%	Thoracic CT	70	Shape-based Genetic Algorithm Template-Matching (GATM)
Liao *et al.* [193]	Train: 90%, Test: 87%	LUNA16 DSB2017	LUNA16: 888 DSB: 1186	3D-Faster R-CNN
Ye *et al.* [164]	90.2%	Thoracic CT scans	108	Shape-Based CAD
Messay *et al.* [47]	92.8%	LIDC-IDRI	84	ML
Sousa *et al.* [45]	95.21%	CT scans	33	ML
Cascio *et al.* [46]	97%	LIDC-IDRI	84	Stable 3D mass–spring models
Monkam *et al.* [101]	97.35%	LIDC-IDRI	1010	3D-CNN, ELM
Choi *et al.* [165]	97.4%	LIDC-IDRI	84	3D-shape-based feature descriptor
Bhatt *et al.* [167]	Train: 99.38%, Test: 99.60%	LUNA16	888	SVGG
Proposed SVGG	Train: 99.40%, Test: 96.26%	LUNA16 and DSB2017	LUNA16: 888 DSB: 1186	SVGG

4.5 Discussion

The methodologies presented in the previous chapter needed distinct procedures for feature extraction and classification. Thus, a better strategy is developed in this chapter, which comprises of several deep neural networks for lung cancer classification. The accuracy of the model is employed as an evaluation metric to estimate its effectiveness.

Seven prominent DNNs including MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and VGG-19 are trained using patches extracted from the LUNA16 and DSB2017 datasets.

The MobileNet which is considered to be lightweight neural network due to architecture of DWSC was trained but there was a very large variation in performance of train and validation accuracy. The accuracy during training was 92.47%, whereas it was 39.56% during validation, which was quite poor. This is due to the 28 trainable layers, which caused overfitting and leading to low performance during validation.

As a result, SqueezeNet was trained. This model is made up of fire modules. Because of the varying sizes of kernels, it learns both low and high features. Furthermore, the model was not as much deep as MobileNet. This SqueezeNet validation accuracy is 56.45%. However, still it requires improvement for lung cancer classification.

Consequently, AlexNet, which is considered to be the first developed deep neural network, was trained to achieve a validation performance of approximately 66.75%. The model doesn't show a significant improvement compared to SqueezeNet and the reason may be that AlexNet being a small network is underfitting and therefore was unable to learn the complexity present in abnormalities.

Later, GoogLeNet was developed in order to train a neural network with less complexity and smaller networks inside the larger network in order to learn the characteristics present in abnormalities. GoogLeNet also introduced the notion of supplementary learners to accelerate convergence. However, the primary shortcoming of GoogLeNet was its heterogeneous topology, which required module-specific customization. A further disadvantage of GoogLeNet was bottleneck that substantially lowers the feature space in the next layer, which may occasionally lead to the loss of important information, resulting in validation accuracy of 70.45%.

Considering the concerns with GoogLeNet, ResNet-18 was trained using 18 layers with residual blocks to overcome the vanishing gradient problem occurring in very deep neural network. Still the performance was not progressing toward improvement, resulting in validation accuracy of 86.50%.

Thus, after training and visualising the performance of the previously described classifiers, the VGGNet was trained, which is regarded as the most popular DNN in the research community. Two versions of VGGNet, namely VGG-16 and VGG-19 were trained. After visualizing the performance of the classifiers, it is clear that there has been an improvement, although the performance is still about 87.16%.

Based on the performance and challenges experienced by these seven models, a new model was designed. This recommended model was referred to as Simplified VGG (SVGG) as the approach of VGG was used to design this model and it achieved 96.26% accuracy.

The drawbacks of models described in this chapter is they rely on the nodule candidate generating stage to receive the extracted candidate as input data. Nodule extraction from the LUNA16 and DSB2017 datasets requires a distinct pre-processing component under the present configuration. Therefore, the preceding stage of candidate creation establishes the upper limit of detection sensitivity for the subsequent nodule categorization. Lung nodule classification accomplished in this chapter is a part of objective of lung nodule detection. So, further study will investigate the construction of an integrated and end-to-end deep learning framework for detecting lung nodules.

CHAPTER-5

Lung Nodule Diagnosis Using Object Detectors

Numerous uses for the deep learning-based object detection technology may be found in everyday life. For instance, object detection applications which fall under computer vision are crucial for facial recognition, commercial analytics and medical image analysis. In the feature space, objects are recognized using the object detection model. Cloud computing facilities, generic Graphics Processing Units (GPU), Internet of Things (IoT) clusters, or a single embedded device may be used as the computer infrastructure needed for the aforementioned applications. Finding and categorizing the identified objects in an image is the goal of general object detection. Prior to the implementation of deep CNN, conventional detection relied on handcrafted features such as Histogram of Oriented Gradients (HOG), Linear Binary Patterns (LBP), SIFT, Deformable Part Model (DPM), Aggregated Channel Feature (ACF) and so on.

Deep CNN approaches dominate detection due to their superior performance in machine learning tasks. A typical object detector is often made up of two parts: a backbone that is a pre-trained classifier and a head that predicts object classes and bounding boxes. Backbones for detectors operating on GPU platforms might include GoogLeNet, ResNet, VGG and so forth. MobileNet or SqueezeNet might be the backbone for detectors operating on CPU platforms. The object detector is often divided into two types: one-stage object detectors and two-stage object detectors. The most common two-stage object detector includes Fast R-CNN, Faster R-CNN and R-FCN. Whereas, SSD, YOLO, YOLOv2, YOLOv3 and YOLOv4 are the most well-known models for one-stage object detectors. Object detectors built in recent years often include certain layers between the backbone and the head which are known as neck and these layers are typically utilized to gather feature maps at various stages. The neck of an object detector is often made up of numerous bottom-up paths and top-down paths. The Path Aggregation Network (PAN) and the Feature Pyramid Network (FPN) are examples of networks that are equipped with this technique.

To determine the category of the greatest component in an image, the classification techniques CBIR, hybrid approach and DNNs were discussed and implemented in chapters 3

and 4. While the lung cancer diagnosis requires not only categorising whether an abnormality is cancerous or not but also it is necessary to find the precise location on the lung image.

The major objective of this chapter is to develop a nodule detector that is simple to train, operates fast and provide findings that are accurate, persuasive and real-time. Five object detectors are being trained for the identification of nodules as part of this endeavour Faster R-CNN, SSD, YOLOv3, YOLOv4 and YOLOv5 make up these five detectors. This chapter's framework includes data collection and their labelling, nodule detection on NLST datasets, an overall block diagram for nodule detection, implementation of five nodule detectors and comparison of the nodule detectors.

5.1 Datasets

The nodule detector models were trained on the following datasets: Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI), Lung Nodule Analysis 2016 (LUNA16), Data Science Bowl 2017 (DSB2017), National Lung Screening Trial (NLST), Non-Small Cell Lung Cancer (NSCLC)-Radiogenomics, NSCLC-Radiomics, Lung CT Segmentation Challenge (LCTSC), Quantitative Imaging Network (QIN), The Cancer Genome Atlas Lung Squamous Cell Carcinoma (TCGA-LUSC), TCGA-Lung Adenocarcinoma (TCGA-LUAD), 4D-Lung, LungCT-Diagnosis, Reference Image Database to Evaluate Response (RIDER), European Collaboration on Low-Dose Aspirin in Pblycythemia (ECLAP), Anti-Programmed cell death protein 1 (Anti-PD 1) Lung. All datasets are accessible, with the exception of NLST, which is obtained via a grant from the National Institutes of Health (NIH). The locations of cancerous and non-cancerous nodules are annotated for all datasets except NLST. Pre-processing techniques employing basic OpenCV functions and LabelImg toolbox are used to label these datasets as per the annotations provided along with CT scans. While the nodules present in NLST dataset must first be identified and matched using a method that is described in the next section and then they are labelled using LabelImg toolbox.

5.2 Nodule Detection on NLST Datasets Using SIFT and FLANN Based Algorithm

The LUNA16 dataset contains 888 CT scans with hundreds of images and an annotation file with descriptions of lung lesions. These nodules were extracted using the pre-processing approach shown in Fig. 3.5 described in chapter 3 and serve as the template database and the NLST dataset, which lacks precise nodule labelling, is used as a query database and thereby the nodules present in it are identified using SIFT-FLANN based algorithm.

The image feature derived from the SIFT technique has a very high uniqueness, allowing it to be utilized for accurate matching in a large database. The unique feature of SIFT algorithm is that though the scale, rotation, illumination and perspective may all be changed, yet it still produces outstanding results. The first step of SIFT algorithm produces internal representations of the original image is the construction of a scale space. Thereby the Difference of Gaussian (DoG) on images is used to calculate maxima and minima. To make the algorithm effective and robust, the edges and low contrast areas have been removed. Thus, from this final scale the features are detected and described.

FLANN is a collection of algorithms developed for rapid closest neighbour searches in huge datasets with high dimensional characteristics. A FLANN-based matcher requires two dictionaries. The first dictionary identifies the suitable algorithm for the task. The second dictionary specifies the search criteria that will be used. This dictionary indicates how many times the trees in the index must be explored recursively. The FLANN-based matching is done by the distance ratio between the two closest matches of the considered key point to reduce the number of false matches and is a successful match when this value is less than the threshold.

The SIFT-FLANN based algorithm shown in Fig. 5.1 is used for detection and matching of the nodules present in NLST and LUNA16 dataset.

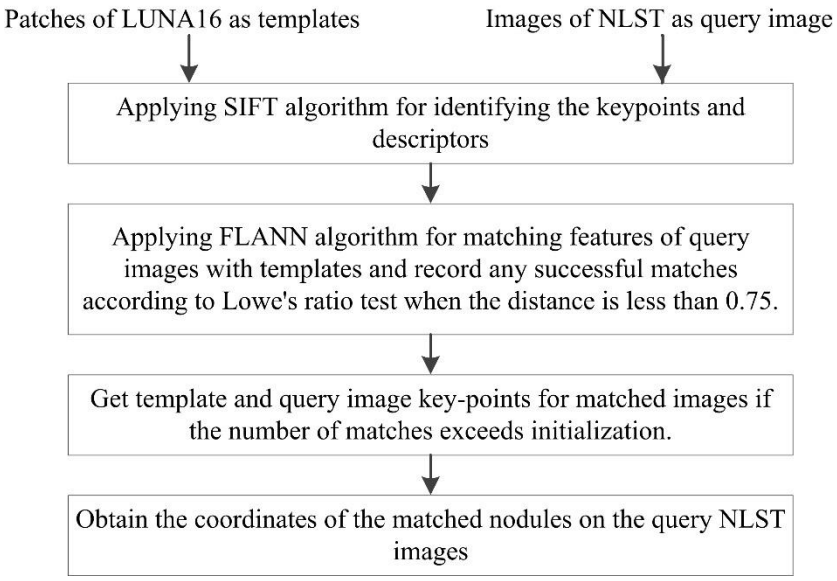


Fig. 5.1 SIFT-FLANN based algorithm to detect and match features of the nodules present in NLST and LUNA16 dataset

The extracted nodules from the LUNA16 dataset are utilized as templates for finding and matching comparable abnormalities in the NLST CT scan dataset. The technique begins with the identification of nodule features in the templates of LUNA16 and NLST lung images using the SIFT algorithm. The FLANN is used to match these identified features between the two datasets.

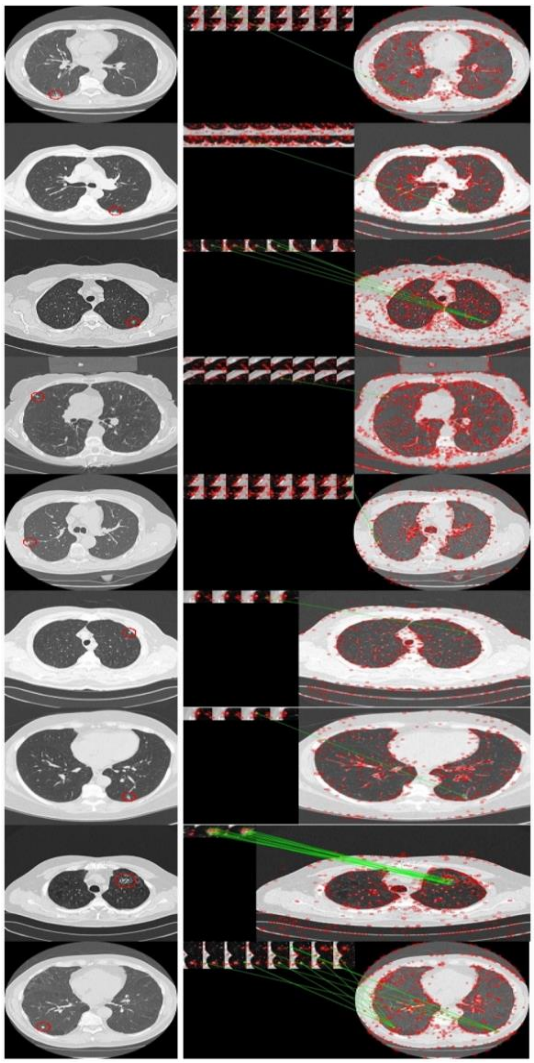


Fig. 5.2 The detected and matched nodule features using SIFT-FLANN based algorithm

The outcomes of nodule feature extraction and matching using algorithm is shown in Fig. 5.2. The left portion of the figure depicts the NLST lung CT scan image, with the nodule indicated by a red circle in accordance with the abnormality variables included in the dataset. The right section displays the nodule templates of the LUNA16 dataset and the NLST lung CT scans, with the detected features indicated in red and the matched features represented by a green line. The identified and matched results demonstrate that the algorithm works precisely for various nodule types and sizes.

The algorithm depicted in Fig. 5.3 is used to confirm the precise location of nodules on NLST CT scans by comparing the matched results obtained on CT scans using the SIFT-FLANN algorithm with the abnormality variables provided in the form of source group and scan year for cancer/non-cancer, the slice number, epicentre, margin and diameter of the abnormality.

There are five source groups to which the CT scans belongs. Among them the source group number 1 and 5 consists of CT scans with cancerous nodules. While, the source group number 2 and 3 consist of CT scans with non-cancerous nodules. Each patient is scanned in a span of three consecutive years leading to three study years T0, T1 and T2. The slice number determines the exact image number on CT scan where the centre of the nodule is present. The epicentre determines the exact position of the nodules in terms of the location on the lung area. There are six regions in which the epicentre is located namely 1: Right Upper Lobe, 2: Right Middle Lobe, 3: Right Lower Lobe, 4: Left Upper Lobe, 5: Lingula and 6: Left Lower Lobe. The margins define the edges of the nodules. There are three types of margins which are spiculated, smooth and poorly defined. The diameter is the longest diameter perpendicular to the longest overall diameter of the nodule, in millimetres.

In this process initially the scan group and study year of the CT scan ID in which the image belongs to is identified. Then the slice number mentioned for this patient ID is matched with the image number where the nodule is detected. After this the epicentre location, margin and diameter of the nodule detected are matched with the abnormality variable. Finally, if the nodule identified is confirmed to that with the annotations in terms of abnormalities variables, the nodules are labelled using LabelImg toolbox.

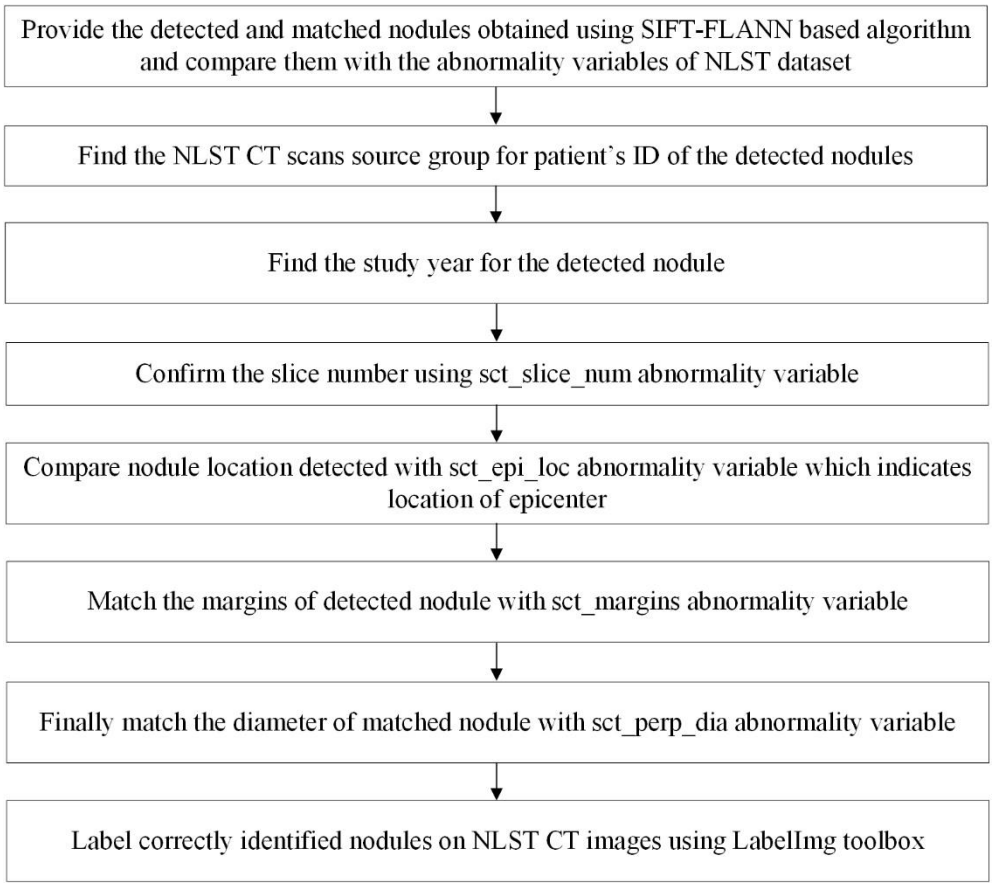


Fig. 5.3 Algorithm for identifying and labelling the nodules on NLST dataset by comparing matched features with abnormality variables

5.3 Algorithm for Configuring and Training the Nodule Detectors

The lung nodule diagnosis using nodule detectors involves a few configuration procedures which are outlined in the algorithm shown in Fig. 5.4. This approach is used to train the nodule detectors in this study. It depicts the whole process, beginning with the labelling and partitioning of data into train and test, label map creation, the formation of the model's configuration file, the setup of hyperparameters for training and lastly training and monitoring of the model's performance.

- Label cancerous and non-cancerous lesions on scans according to the findings specified for the 14 datasets listed in section 5.1. This procedure was carried out using basic OpenCV functions and LabelImg tool box. This resulted in a total of 2,360,00 labelled images with an equal number of cancer and non-cancer labels.
- Separate the total number of labelled datasets into train (70%) and test (30%) data.
- Create a label map according to the labels i.e., cancer and non-cancer.
- Setup configuration file for the five nodule detectors using the TensorFlow library.
- Train the detectors by specifying the path of the datasets and the checkpoint for recursive training. Set the various hyperparameters, such as the number of learning rate, activation functions, number of iterations, epoch, batch size, regularization, dropout, weights and bias initialization, optimization algorithm, learning rate decay, momentum, etc.
- Monitor the outcomes using Tensorboard. Save the checkpoint for model validation.

Fig. 5.4 Algorithm applied for training nodule detectors

5.3.1 Nodule Detection using Faster R-CNN

Deep learning is now the most widely used technology for detecting objects. Faster R-CNN plays an important role in object detection as it has excellent detecting capabilities. It detects objects in images by using two network regions: the Region Proposal Network (RPN) and the detector network. The RPN creates region proposals for the object to be analyzed and the detector network employs these proposals to identify objects in an image.

A. Architecture of Faster R-CNN

The Faster R-CNN is made up of two modules: Inception-V2 with RPN as the first module and Fast R-CNN as the second module. Fig. 5.5 depict the architectures of Faster R-CNN. The first module in the original Faster R-CNN was VGG-16, while Inception-V2 is utilized as the first module in this study because of its feature learning abilities. The feature map of the last layer of Inception-V2 is applied to RPN. The RPN is built by including a few more convolution layers that concurrently regress region boxes and objectness score at each point using anchor boxes of various sizes and aspect ratios. The fixed size feature maps are then generated using Region of Interest (ROI) pooling. The second module, Fast R-CNN detector, establishes the diagnostic

likelihood of being cancer or non-cancer on these feature maps and the bounding box is constructed based on the regressor coordinates. Fast R-CNN permits end-to-end detector training on shared convolutional features and demonstrates remarkable accuracy and speed. As a result, the Faster R-CNN is a unified network for detecting lung nodules.

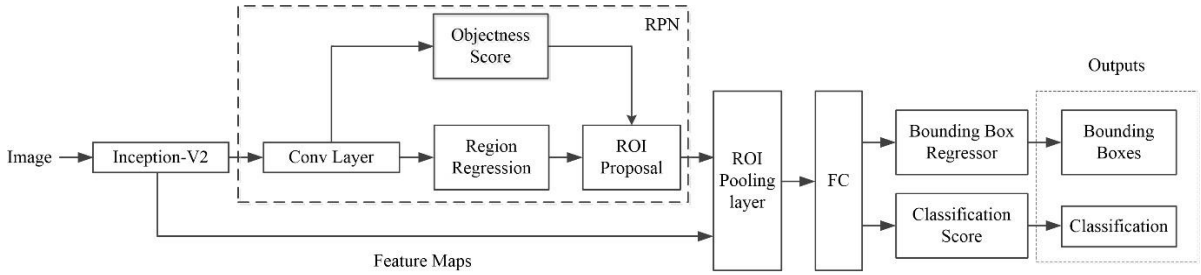


Fig. 5.5 Architecture of Faster R-CNN [180]

Extensive research and improvement on the original Faster R-CNN design has been carried out by extending the network's width and height and this has enhanced the accuracy of detection. In a series of enhancements to the network for object detection, the new network known as Inception has achieved the maximum level of accuracy. This network's name is derived from [143] architecture known as Network in Network NIN. The computing demands and time, however, dramatically rise as network depth grows. This issue may be solved by adding sparsity to the network and substituting fully connected layers by sparse ones. The foundation of a Faster R-CNN's architecture is made up of Inception modules. Inception design accelerates performance three to ten times higher than non-Inception architecture.

In the Faster R-CNN architecture, the features of a images are retrieved using a convolutional neural network loaded with Inception-V2 filter banks. Inception-V2 was designed to increase the stability of the network towards variations by using Module 1, 2 and 3. In Module 1, 5×5 convolutions from the original Inception are factorized into two 3×3 convolution operations to improve the computational speed as shown in Fig. 5.6. This reduces the cost of 5×5 convolution by 2.78 times and leads to a boost in performance. Fig. 5.7 illustrates Module 2 that has a 3×3 convolution made equivalent by performing a 1×3 convolution first and then performing a 3×1 convolution. This method is 33% inexpensive than the existing 3×3 convolution. The representational bottleneck is minimized by the use of wider filter banks instead of deeper filter banks as mentioned in Module 3 and illustrated in Fig. 5.8. Table 5.1 outlines the layers and architecture of Inception-V2.

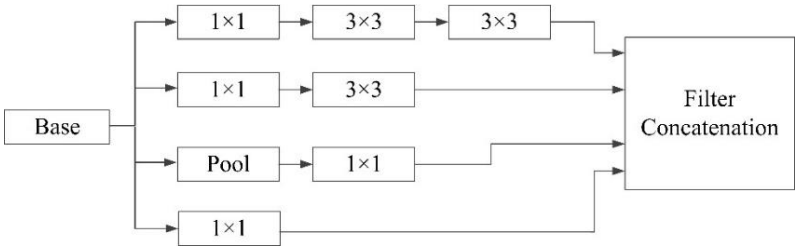


Fig. 5.6 Dimensionality Reduction filter [142]

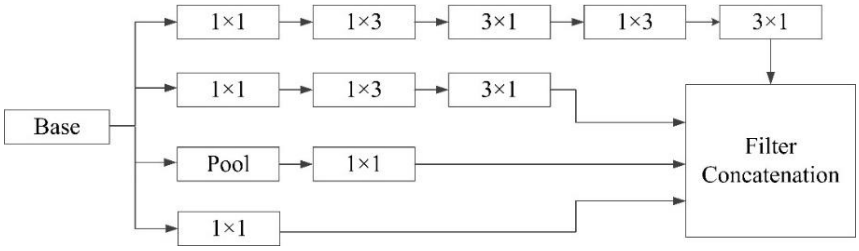


Fig. 5.7 Inception-V2 module with deeper filter banks [142]

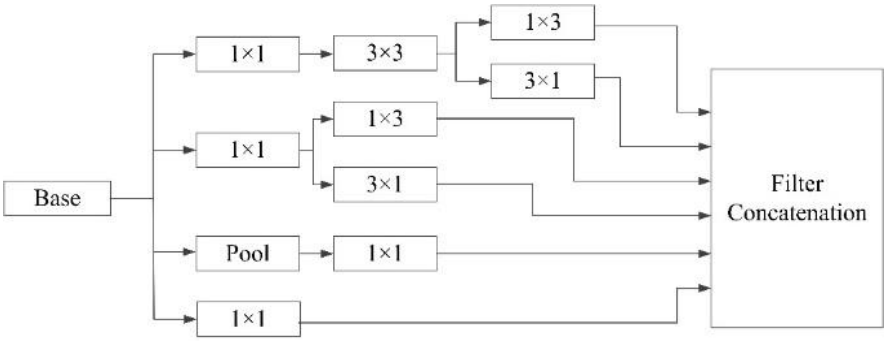


Fig. 5.8 Inception-V2 module with wider filter banks [142]

Table 5.1 The outline of the architecture of the Inception-V2 [142]

Type	**Patch size/stride**	**Input size**
Conv	$3 \times 3/2$	$229 \times 229 \times 3$
Conv	$3 \times 3/1$	$149 \times 149 \times 32$
Conv padded	3×1	$147 \times 147 \times 32$
Pool	$3 \times 3/2$	$147 \times 147 \times 64$
Conv	3×1	$73 \times 73 \times 64$
Conv	$3 \times 3/2$	$71 \times 71 \times 80$
Conv	3×1	$35 \times 35 \times 192$
$3 \times$ Inception	As in Fig. 5.6	$35 \times 35 \times 288$
$5 \times$ Inception	As in Fig. 5.7	$17 \times 17 \times 768$
$2 \times$ Inception	As in Fig. 5.8	$8 \times 8 \times 1280$

The RPN generates region proposals by sliding a small network over the convolutional feature map output from the last shared convolutional layer of the Inception-V2. The RPN architecture is naturally implemented with 3×3 convolutional layer followed by two siblings 1×1 convolutional layers i.e., a box-regression layer (reg) and a box-classification layer (cls). Multiple region proposals are generated simultaneously at each sliding-window location where the number of maximum possible proposals for each location is given by k. Consequently, the regression layer has 4k outputs that contain the coordinates of k boxes and the classification layer has 2k outputs that evaluate the likelihood of an object or non-object for each proposal. The parameters of the k proposals are based on k reference boxes which are called anchors. An anchor with a scale and aspect ratio is centred at the sliding window. Three scales and three aspect ratios are used in this work resulting in k = 9 anchors at each sliding position.

Each anchor has a binary class label that is used to train RPNs. Two types of anchors are given a positive label: (i) those with the greatest Intersection-over Union (IoU) overlap with a ground-truth box, or (ii) those with an IoU overlap greater than 0.7 with any ground-truth box. If the IoU ratio of a non-positive anchor is less than 0.3 across all ground-truth boxes, the anchor is given a negative label.

It's crucial to compute the loss during model training so that weights may be updated according to gradients. These loss functions are always used to aid a computer's learning process. The loss function of an algorithm serves as a measure for how effectively an object can assess a certain set of data inside a given algorithm given the quantity of data that are provided. As indicated in Eq. (5.1), the total loss function is incurred by applying the class loss and regression loss.

$$\boldsymbol{L}(\{\boldsymbol{p_i}\},\{\boldsymbol{t_i}\}) = \frac{\mathbf{1}}{\boldsymbol{N_{cls}}}\sum_{\boldsymbol{i}} \boldsymbol{L_{cls}}(\boldsymbol{p_i}, \boldsymbol{p_i^*}) + \boldsymbol{\lambda}\frac{\mathbf{1}}{\boldsymbol{N_{reg}}}\sum_{\boldsymbol{i}} \boldsymbol{p_i^*}\boldsymbol{L_{reg}}(\boldsymbol{t_i}, \boldsymbol{t_i^*}) \tag{5.1}$$

Where, i represents the index of an anchor in a mini-batch and p_i is the estimated probability of anchor i being an object. If the anchor is positive, the ground-truth label p_i^* is 1 otherwise it is 0. t_i is a vector encoding the predicted bounding box's four parameterized coordinates and t_i^* is the ground-truth box associated with a positive anchor.

The classification error L_{cls} represents log loss across two classes. The regression loss is calculated using the equation $L_{reg}(t_i, t_i^*) = R(t_i - t_i^*)$ where R is the robust loss function. The term $p_i^* L_{reg}$ denotes that the regression loss is enabled only for positive anchors ($p_i^* = 1$) and

is disabled for all other anchors $(p_i^* = 0)$. The *cls* and *reg* layer's outputs are p_i and t_i respectively.

The two terms are weighted by a balancing parameter λ and normalised by N_{cls} and N_{reg}. By setting λ = 10, the weights of the *cls* and *reg* terms are almost identical. The parameterizations are used for the four locations listed below for bounding box regression:

$$t_x = (x - x_a)/w_a\,, t_y = (y - y_a)/h_a,$$

$$t_\omega = \log(w/w_a),\ t_h = \log(h/h_a),$$

$$t_x^* = (x^* - x_a)/w_a\,, t_y^* = (y^* - y_a)/h_a,$$

$$\boldsymbol{t_\omega^* = log(w^*/w_a),\ t_h^* = log(h^*/h_a)}$$

(5.2)

where x, y, w and h denote the box's center coordinates and its width and height. Variables x, x_a and x^* are for the predicted box, anchor box and ground-truth box respectively (likewise for y, w and h).

B. Implementation and Simulation Result

The network is fine-tuned using SGD with an initial learning rate of 0.0001, momentum of 0.8, weight decay of 0.0005 and batch size of 8. For fine-tuning large neural networks in computer vision, SGD is the most used optimizer. It has advantages such as faster convergence and lower memory requirements, especially for large datasets. The layers are seeded at random using weights drawn from a zero-mean Gaussian distribution with a standard deviation of 0.009. The network has been taught to take 45k steps. The IoU threshold is set to 0.5. The training of Faster-RCNN with Inception-V2 was carried out using TensorFlow. Tensorboard is used to visualize model performance.

The model's performance is measured using three kinds of losses: box loss, objectness loss and classification loss. The box loss quantifies how effectively the algorithm can detect an object's centre and how well the anticipated bounding box covers an object. Objectness is simply a measure of the likelihood of finding an object in a particular region of interest. If the objectness value is high, the image window most likely contains an object. Classification loss indicates how successfully the algorithm predicts the proper class of an object.

Fig. 5.9 shows the performance for nodule detection using Faster R-CNN. It shows the graphical representation of the following losses: Box Localization, Box Classifiers, RPN Localization, RPN Objectness and Total Loss.

It can be visualized that the box, objectness and the classification losses reduces till 20k steps and thereby this rate of loss reduction mostly remains constant. So, the model is trained for 45k steps. The total loss is calculated using the above mentioned four losses and it can be analyzed form the total loss graph that the accuracy achieved using this method is 95.80%.

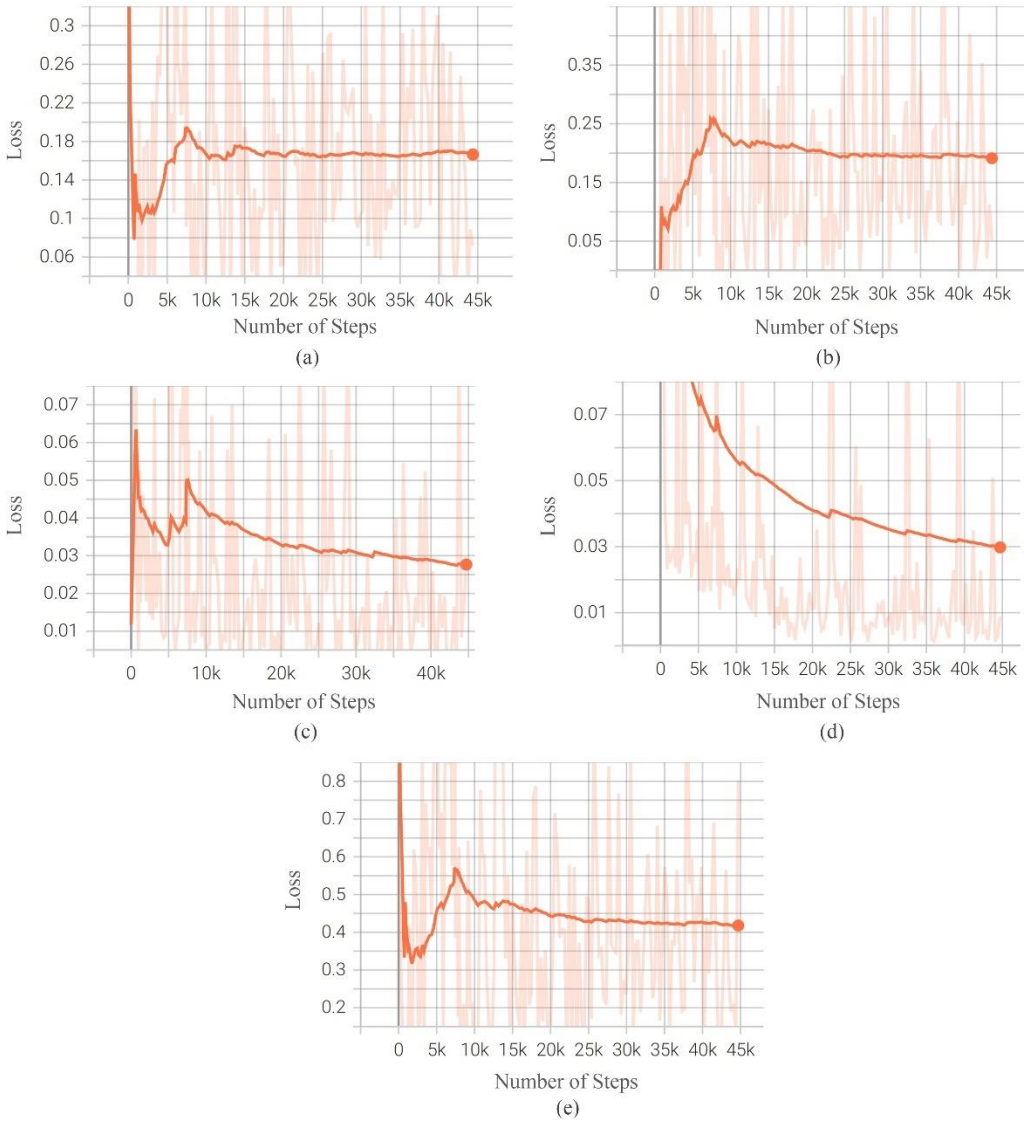
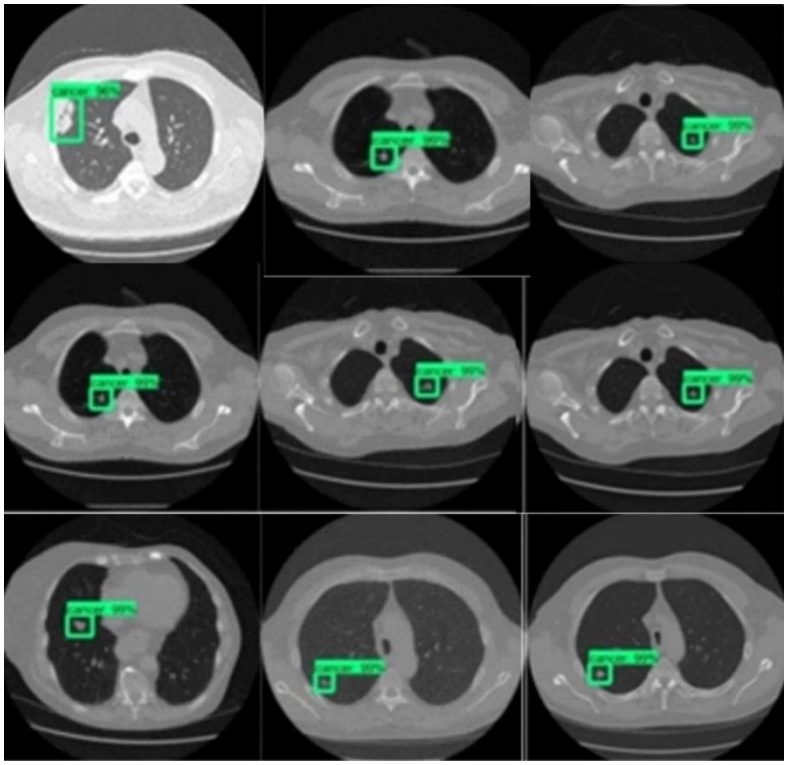


Fig. 5.9 Simulation result of (a) Box Localization (b) Box Classifiers (c) RPN Localization (d) RPN Objectness and (e) Total Loss for nodule detection on Faster R-CNN

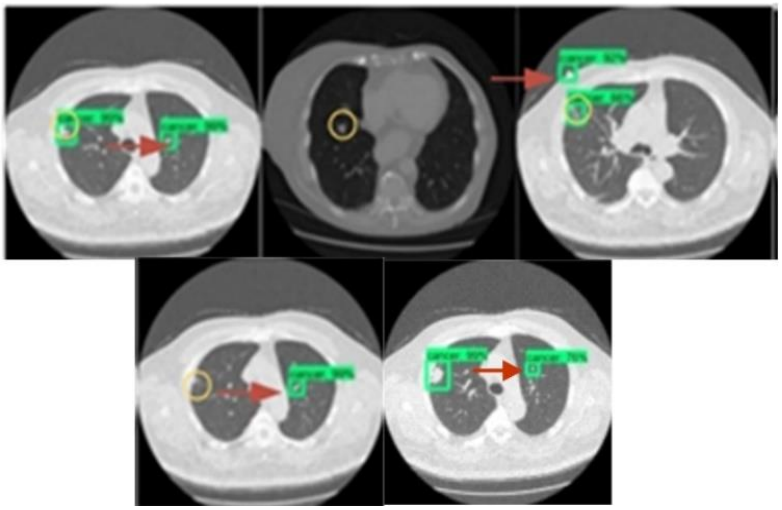
Training the model took 96 hours using NVIDIA GeForce GTX 2070. The maximum and minimum total loss was 0.85 and 0.42 respectively. Minimum loss is desired and a decreasing

value means the model is learning during the training. Training is stopped as the loss is not decreasing anymore. During the training, it periodically saves a checkpoint every five minutes. This checkpoint will be used to export inference graph (.pb file) that contains graph variables frozen as constants.

After the training, inference graph was exported. Inference graph contains the object detection classifier. Fig. 5.10 and Fig. 5.11 show the detection on the sample images with cancerous and non-cancerous nodules respectively. The detection results in Fig. 5.10 (a) shows there are correct detection of cancer nodules while few which are not detected and incorrectly detected are shown in Fig. 5.10 (b). The correct location for non-detected nodules is shown by circle and incorrectly detected and labelled nodules are shown by an arrow. Same is the case for non-cancerous nodules present on sample images which are shown in Fig. 5.11.



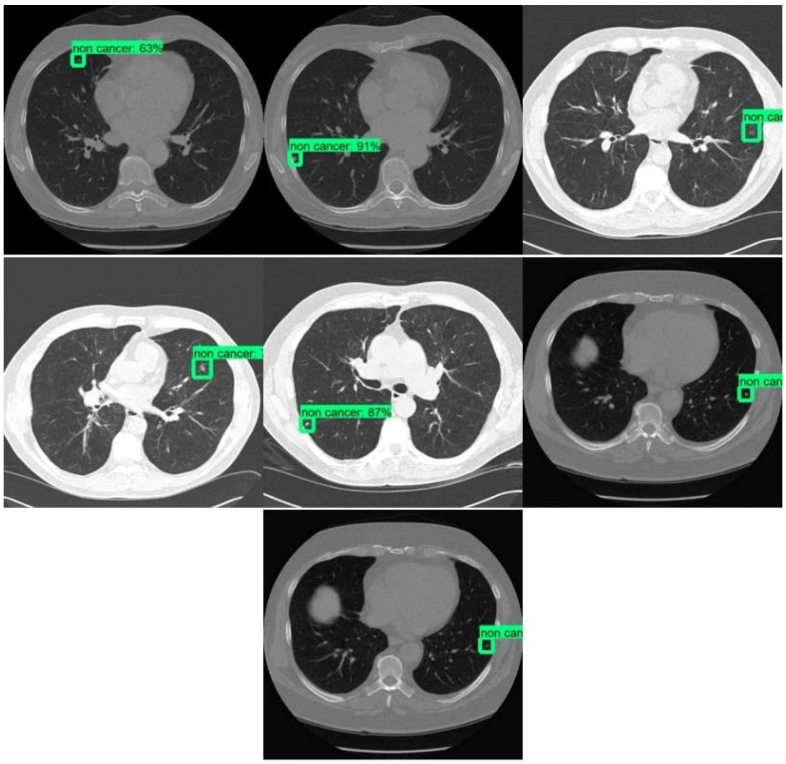
(a)



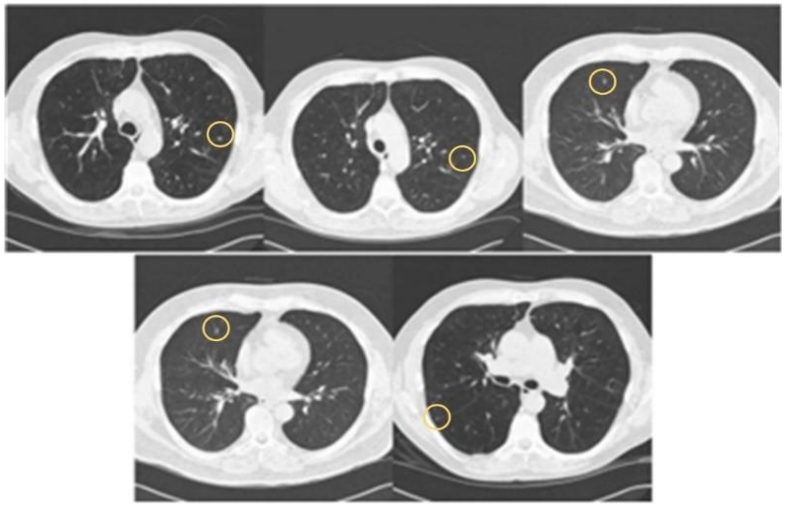
(b)

Fig. 5.10 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using Faster R-CNN

The reason for incorrect cancer detection might be similar features between cancerous and non-cancerous nodules which proves that Faster R-CNN has not learnt to differentiate very similar characteristics between the two. While the reason for non-detection in case of cancer and non-cancer nodules can be the size of the abnormalities are very small and as the Faster R-CNN detects on single scale so it fails to detect the small sized nodules.



(a)



(b)

Fig. 5.11 (a) True Prediction (b) False Prediction/non-detection of non-cancerous nodules on test images using Faster R-CNN

5.3.2 Nodule Detection using Single Shot Detector

Modern object detection systems are variants of the following approach: prediction of bounding boxes, resample pixels or features for each box and then use a high-quality classifier. The object detectors have been computationally demanding for embedded systems and too sluggish for real-time applications even with high-end hardware. By addressing each step of the detection pipeline, there have been several efforts to construct quicker detectors, but so far, this has only resulted in notably inferior detection accuracy. The performance improvement in SSD comes from removing bounding box suggestions and the subsequent feature resampling.

Among the enhancements made in SSD are the use of a small convolution layer filter to identify object categories and offsets in bounding box positions, the use of separate filters for different aspect ratio detections and the application of these filters to various feature maps from final stages of a network to accomplish detection at different scales, which enhances the model's performance.

A. Architecture of SSD

The SSD method uses a feed-forward convolutional network to generate a fixed-size set of bounding boxes and scores for the occurrence of class in those boxes, proceeded by a non-

maximum suppression phase to provide the final detections. The MobileNet-V2 architecture is used in the base network for nodule feature extractor. Then, at the ending of the base network, an auxiliary network is added to provide multiscale feature maps for detection. These layers gradually shrink in size and enable detection predictions at multiple scales. There are six feature maps extracted, two from layers and four from the auxiliary convolution layer. The detector uses these six feature maps to identify and locate nodules. Fig. 5.12 shows the architecture of SSD.

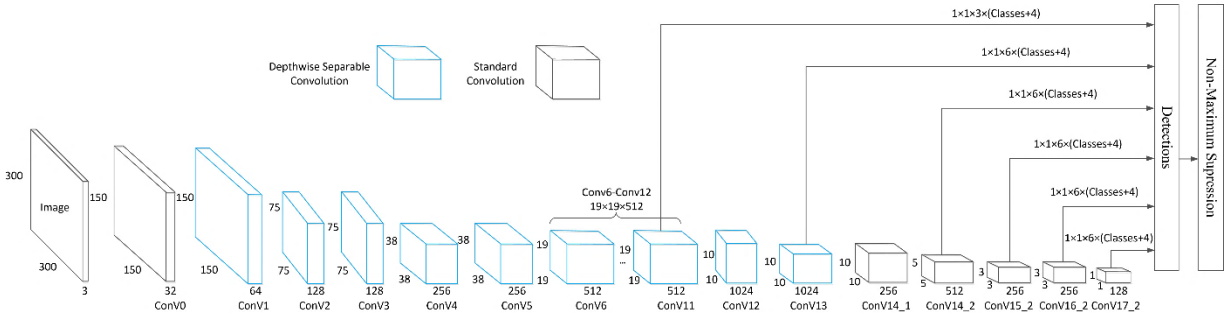


Fig. 5.12 Architecture of SSD [197]

A set of default bounding boxes has been assigned to each feature map cell for the several feature maps at the top of the network. For each feature map, the scale of the default boxes is calculated as follows:

$$\boldsymbol{s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1),\ k \in [1, m]} \tag{5.3}$$

where the bottom layer has a scale s_{min} of 0.2 and the uppermost layer has a scale s_{max} of 0.9 and all layers in between are regularly spaced. Different aspect ratios have been imposed for the default boxes and denote them as $a_r \in \left\{1, 2, 3, \frac{1}{2}, \frac{1}{3}\right\}$. The width ($w_k^a = s_k\sqrt{a_r}$) and height ($h_k^a = s_k/\sqrt{a_r}$) can be computed for each default box. For the aspect ratio of 1, the default box is added whose scale is $s_k' = \sqrt{s_k s_{k+1}}$, resulting in 6 default boxes per feature map location. The centre of each default box is set to $\left(\frac{i+0.5}{|f_k|}, \frac{j+0.5}{|f_k|}\right)$, where $|f_k|$ is the size of the k-th square feature map, $i, j \in [0, |f_k|]$.

The default boxes tile the feature map convolutionally, thus each box's location relative to its associated cell is fixed. The prediction is accomplished for the offsets relative to the default box shapes in the cell, as well as the per-class scores that indicate the existence of a class instance in each of those boxes, at each feature map cell. The *c* class scores and the four offsets relative

to the original default box shape are calculated for each box out of k at a given position. This results in a total of $(c + 4)k$ filters being applied around each feature map position, providing $(c + 4)kmn$ outputs for a $m \times n$ feature map. As there are two classes i.e., cancer and non-cancer, $c = 2$ is used.

By aggregating predictions for all default boxes with varied scales and aspect ratios from all locations of several feature maps, a diversified collection of predictions encompassing a wide range of input nodule sizes and shapes is generated. Let $x_{ij}^{p} = \{1, 0\}$ indicate a match between the i-th default box and the j-th ground truth box of category p. In the above matching approach, $\sum_i x_{ij}^{p} \geq 1$. The total objective loss function is a weighted sum of localization loss (L_{loc}) and confidence loss (L_{conf}):

$$\boldsymbol{L(x, c, l, g) = \frac{1}{N}\left(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)\right)} \tag{5.4}$$

where N is the number of default boxes matched. If N equals zero, the loss is deemed to be zero. The localization loss is a Smooth L1 loss between the parameters of the estimated box (l) and the ground truth box (g). Similar to Faster R-CNN, offsets are regressed for the centre (cx, cy), width (w) and height (h) of the default bounding box (d).

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^{k} smooth_{L1}\left(l_i^m - \hat{g}_j^m\right);$$

$$\hat{g}_j^{cx} = \left(g_j^{cx} - d_i^{cx}\right)/d_i^{w}; \quad \hat{g}_j^{cy} = \left(g_j^{cy} - d_i^{cy}\right)/d_i^{h};$$

$$\boldsymbol{\hat{g}_j^{w} = log\left(\frac{g_j^{w}}{d_i^{w}}\right)} \; ; \; \boldsymbol{\hat{g}_j^{h} = log\left(\frac{g_j^{h}}{d_i^{h}}\right)} \tag{5.5}$$

The confidence loss is the softmax loss over classes confidences and can be represented by

$$\boldsymbol{L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^{p} log\left(\hat{c}_i^{p}\right) - \sum_{i \in N_{eg}} log\left(\hat{c}_i^{0}\right) \; where \; \hat{c}_i^{p} = \frac{exp(c_i^{p})}{\sum_p exp(c_i^{p})}} \tag{5.6}$$

B. Implementation and Simulation Result

The model is fine-tuned using SGD with initial learning rate of 0.0001, 0.9 momentum, 0.0005 weight decay and batch size of 8. The layers are initialized randomly by drawing weights from

a zero-mean Gaussian distribution and with standard deviation of 0.009. The network is trained for 45k steps. The default IoU threshold is set to 0.5. TensorFlow library is used to implement this model. Tensorboard is used to visualize the performance of model.

It can be visualized from Fig. 5.13 (a) and Fig. 5.13 (b) that the box localization and the classification losses reduce continuously. In order to compare with Faster R-CNN the model is trained for 45k steps. Fig. 5.13 (c) shows the total loss versus number of steps of the whole training. The total loss is calculated using the above mentioned two loss and it can be analysed form the total loss graph that the accuracy achieved using this method is 43.80%. The SSD with MobileNet-V2 was trained using TensorFlow library. This model has a higher final loss value than the Faster R-CNN. The maximum and minimum loss was 5.95 and 5.60 respectively. Training using SSD model took 168 hours using NVIDIA GeForce GTX 2070.

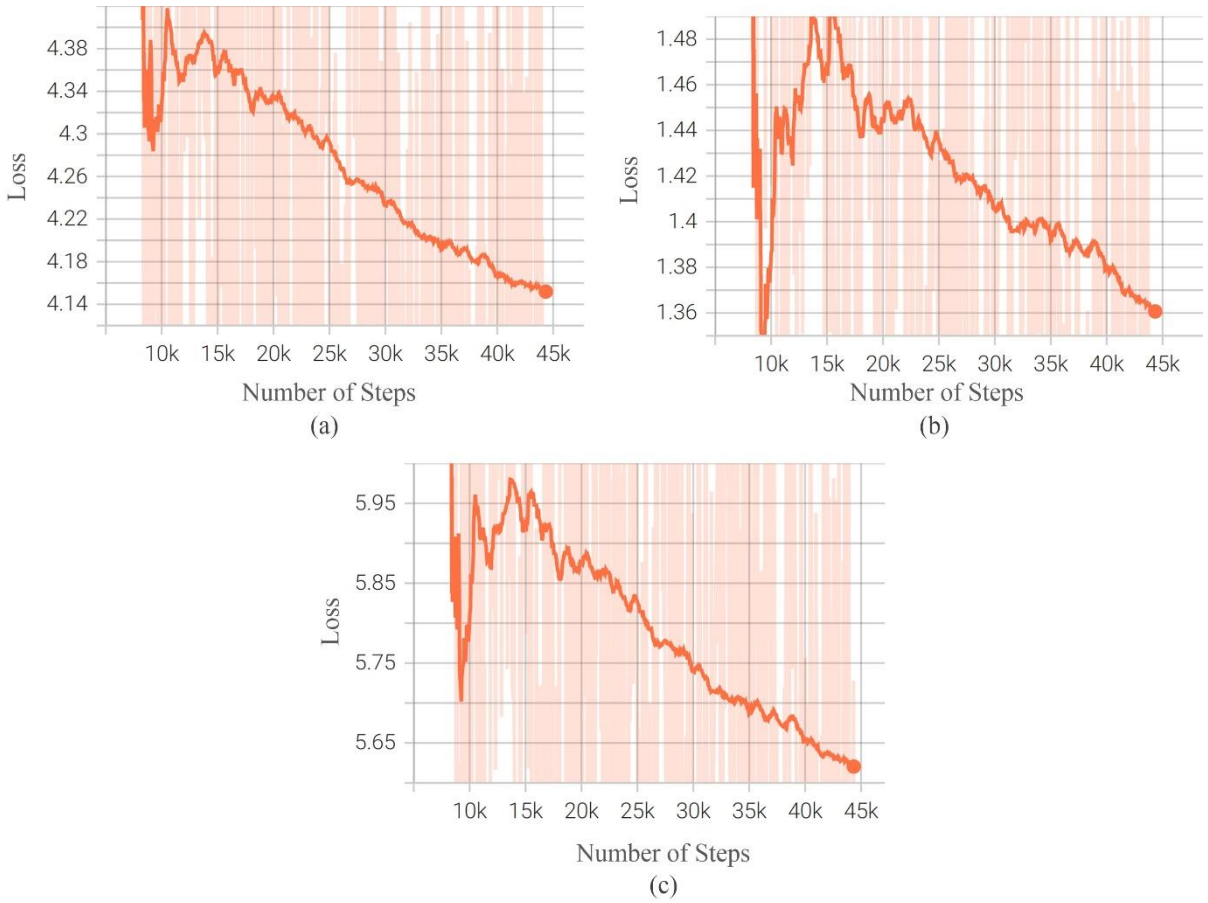
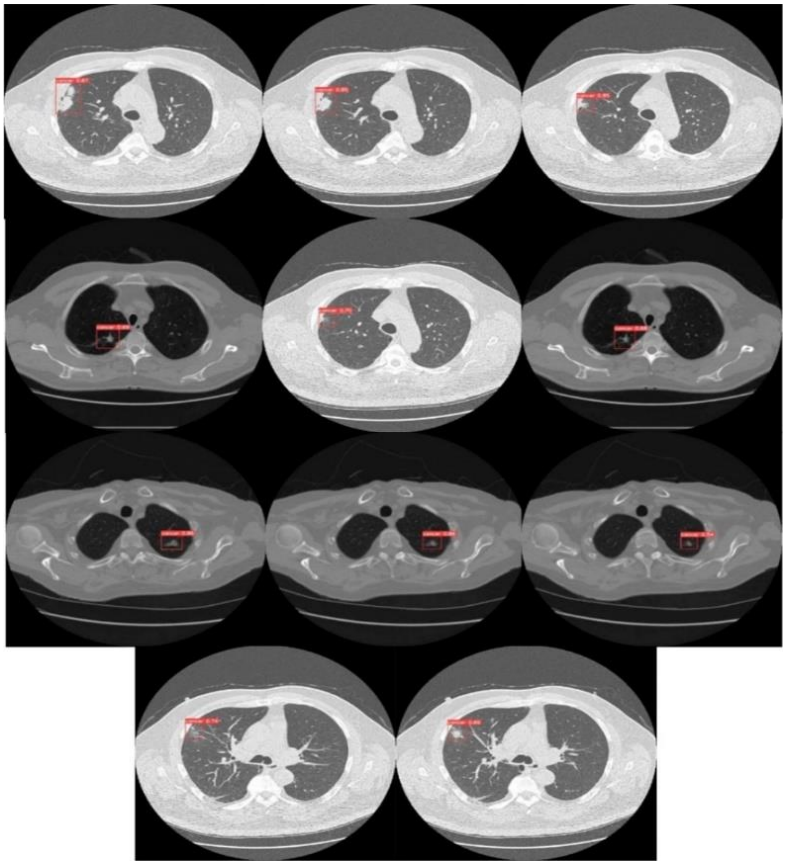


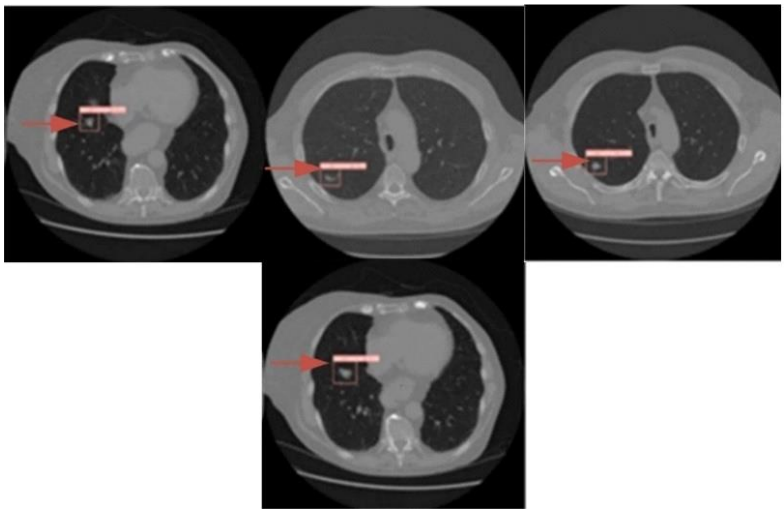
Fig. 5.13 Simulation result of (a) Box Localization (b) Box Classifiers and (c) Total Loss for nodule detection on SSD

Fig. 5.14 and Fig. 5.15 show the performance for nodule detection on cancerous and non-cancerous datasets using SSD. The sample test images are provided to SSD with MobileNet-

V2. The model successfully detected mostly all cancerous nodules as shown in Fig. 5.14 (a) while one which is incorrectly detected is shown in Fig. 5.14 (b). While for the case of non-cancerous nodules, the SSD shows poor performance and many remain undetected because of their small size. The correct location for non-detection cancer/non-cancer nodules are shown by a circle and false prediction of cancerous nodule is shown by an arrow.

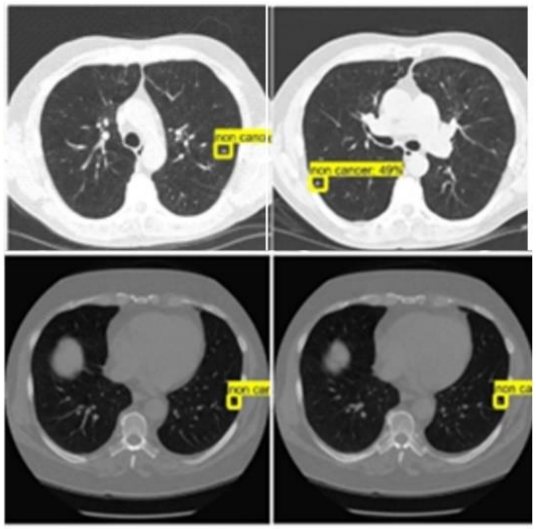


(a)

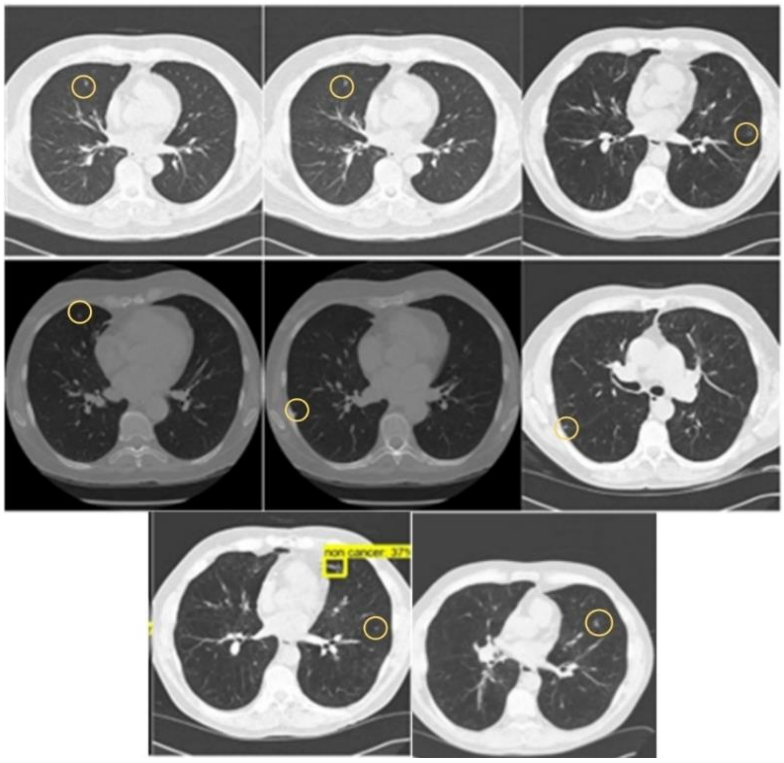


(b)

Fig. 5.14 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using SSD



(a)



(b)

Fig. 5.15 (a) True Prediction (b) False Prediction/non-detection of non-cancerous nodules on test images using SSD

Table 5.2 shows the comparison of detection accuracy for the two trained TensorFlow models. Faster R-CNN with Inception-V2 has lower average loss than SSD with MobileNet-V2.

Table 5.2 Comparison of detection accuracy of Faster R-CNN and SSD

Model	Accuracy	Steps
Faster R-CNN	95.80%	45,000
SSD	43.80%	45,000

Fig. 5.9 (e) and Fig. 5.13 (c) and shows the graph of the total loss against number of steps for the two models: Faster R-CNN and SSD. It can be noticed from the graph that, Faster R-CNN Inception-V2 model takes approximately 45000 steps to reach loss of 0.42. While the SDD with MobileNet V2 model, even though it takes approximately 45000 steps, the loss did not reduce to a level below 5.50. Since the model did not converge properly, the training process has been stopped manually. For comparing the performance of the models, test cancer and non-cancer images were used for nodule detection. These results show poor prediction for nodule detections and so further there was necessity of better method for achieving the goal. From the testing done with two models, it can be concluded that Faster-RCNN Inception-V2 model works better than SSD MobileNet V2 model. Faster R-CNN model performs well in both detecting the nodules and quick convergence during training phase.

Therefore, in this work models based on the You Only Look Once (YOLO) architecture which lets end-to-end models were trained for better results.

The unified architecture of YOLO is a very high-speed design that reframes object detection as a single regression problem, heading directly from image pixels to bounding box coordinates and class probabilities. There have been five YOLO implementations to date. In consideration of the performance of YOLO models, this chapter implements the three most recent versions of YOLO models i.e., YOLOv3, YOLOv4 and YOLOv5. The inputs are annotations in the YOLO format and the architectures are trained over 150 epochs each.

Training an object detection model usually requires two inputs which are the image and the ground-truth bounding boxes for the object in each image. When the model predicts the bounding box, it is expected that the predicted box will not match exactly the ground-truth box. IOU is calculated by dividing the area of intersection between the two boxes by the area of their union which means that higher IOU scores generally translate into better predictions. Additionally, it is worth noting that the IOU threshold is used to calculate precision and recall. IOU is the ratio between area of overlap and area of union of the ground truth label and the prediction label. Specifically, IOU threshold is used to classify whether the prediction is true

positive or false positive. After calculating precision and recall for different IOU thresholds precision and recall plot is created for a single classifier at different IOU thresholds.

The evaluation metrics used are Precision, Recall, mAP and F1 score as the criteria to compare the YOLOv3, YOLOv4 and YOLOv5 algorithms in this work. Note that precision is calculated as the ratio of true prediction to the total number of predictions as shown in Eq. (5.7). Precision does not consider the actual number of true objects present in an image; however, recall calculates the ratio of true predictions to the total number of objects present in an image as mentioned in Eq. (5.8). Having only high precision or only high recall does not necessarily mean the model is accurate. There should be a balance between both precision and recall in order for an object detection algorithm to be considered accurate. Therefore, the F1 score is used to decide whether a model is accurate or not. F1 score is the harmonic mean of precision and recall, shown in Eq. (5.10). It is also the model's test accuracy. The highest possible value of F1 score is 1, which indicates perfect precision and recall and the lowest possible score is 0, which indicates either the precision or recall is zero. In addition, mAP is calculated by taking mean of average precision of all the classes, as shown in Eq. (5.9), where q is the number of queries and AveP(q) is the average precision for that given query.

$$Precision = \frac{True\ Positive}{Ture\ Positive+False\ Positive} \tag{5.7}$$

$$Recall = \frac{True\ Positive}{True\ Positive+False\ Negative} \tag{5.8}$$

$$mAP = \sum_{q=1}^{Q} \frac{AveP(q)}{Q} \tag{5.9}$$

$$F1_{score} = 2 * \frac{(Precision*recall)}{(Precision+recall)} \tag{5.10}$$

Following the training procedure, the model's performance is tested using Precision, Recall and mAP where IOU is between 0.5 (50%) and 0.95 (95%). The graphs of the evaluation measures are plotted using Tensorboard to determine the model's performance during training.

5.3.3 Nodule Detection using YOLOv3

Unlike the previous two models, which employed sliding window and region proposal-based techniques, YOLOv3 views the whole image during training and testing, thus it implicitly encrypts qualitative information about classes as well as their features. YOLOv3 is very quick

and does not need a complex pipeline. End-to-end training is possible with the YOLOv3 architecture while maintaining performance.

A. Architecture of YOLOv3

The YOLOv3 architecture depicted in Fig. 5.16 has three regions: the backbone, the neck and the head. The Darknet53 network, which is the backbone, employs consecutive 3×3 and 1×1 convolution layers with shortcut as shown in Fig. 5.16. The FPN is employed in the neck area to extract features from three distinct scales. The FPN allows prior network layers to be combined with up-sampled features through concatenation. A couple of extra convolution layers are added in the head area to analyze the combined feature maps of the FPN, determine if the nodule is cancerous or not as well as its location.

Darknet-53 has the maximum measured number of Floating-Point Operations Per Second (FLOPs) and greater speed, hence it was chosen as the backbone for YOLOv3. This indicates that the network layout leverages the GPU more efficiently, making evaluations quicker.

YOLOv3 uses FPN to predict boxes at 3 different scales. In the FPN, the feature maps are taken from previous two layers and up-sampling them twice. Concatenation is used to combine previous network feature map with the up-sampled features. With the use of this technique, the extraction of more insightful semantic data from the up-sampled features and finer-grained data from the previous feature map is possible. Then, a couple of additional convolutional layers are added to evaluate this combined feature map and predict a similar tensor that is double the size.

The same design is used one more time to predict boxes for the final scale. Thus, predictions for the third scale benefit from all earlier computation as well as fine-grained features from early in the network is accomplished. Using bounding box priors, predictions are made across these three scales. The scales vary in size (8x8, 16x16, 32x32) to identify both tiny and big nodules.

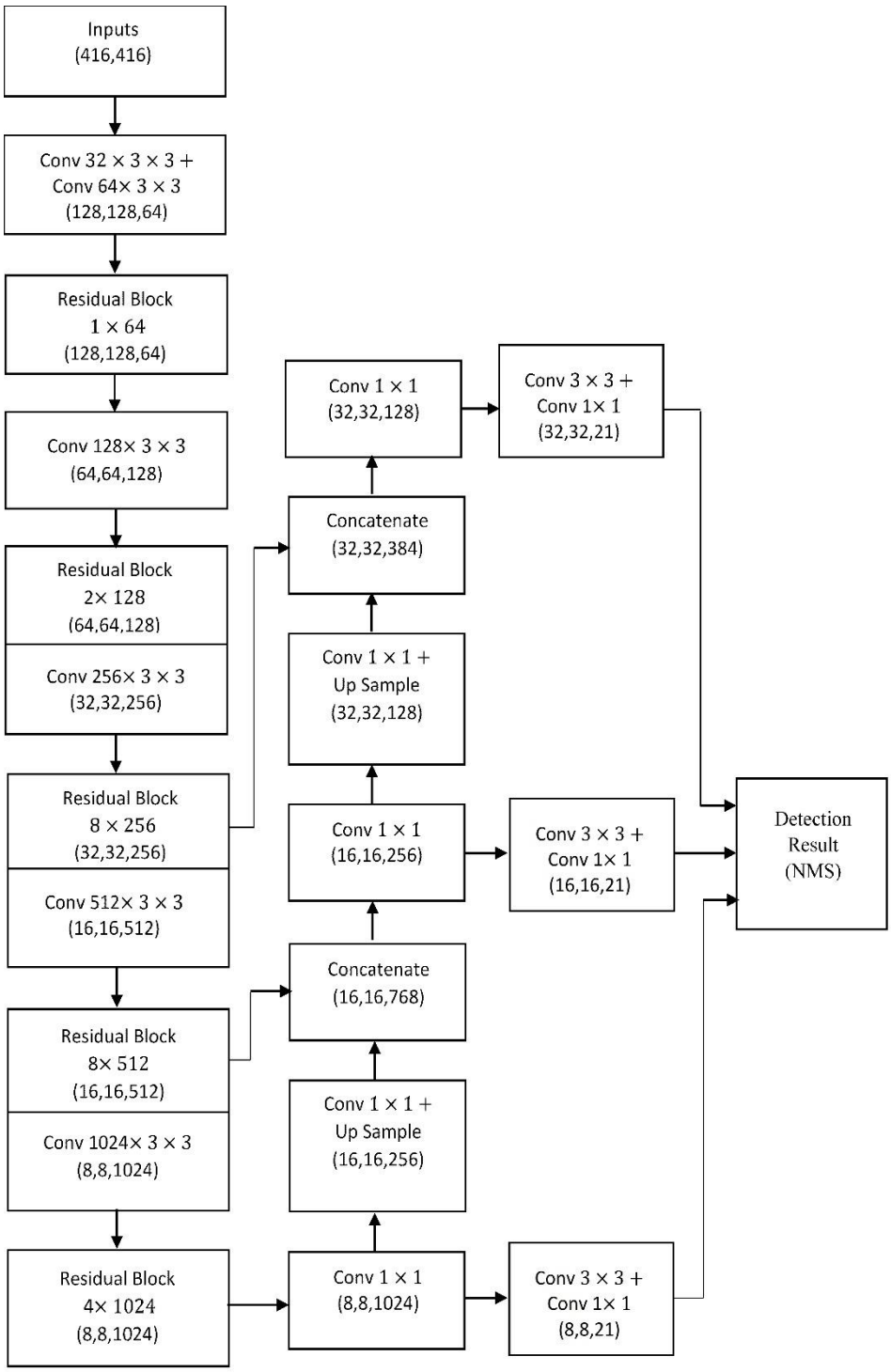


Fig. 5.16 Architecture of YOLOv3 [206]

In the head region K-means clustering is employed to calculate bounding box priors. There are 9 clusters and 3 scales which are distributed equally among the scales. Dimension clusters are used as anchor boxes to predict the bounding boxes. For each bounding box, the network

predicts four coordinates t_x, t_y, t_w, t_h. If the cell is shifted by (c_x, c_y) from the upper left corner of the image and the bounding box prior has width and height (b_w, b_h) then the predictions correspond to:

$$\boldsymbol{b_x = \sigma(t_x) + c_x\ ;\ b_y = \sigma(t_y) + c_y\ ;\ b_w = p_w e^{t_w}\ ;\ b_h = p_h e^{t_h}} \quad (5.11)$$

Several convolutional layers are added to the base feature extractor. The final one predicts a 3-dimensional tensor encoding bounding box, objectness and class predictions. The YOLOv3 model predicts three boxes at each scale, making the corresponding tensor $N \times N \times [3 \times (4 + 1 + 2)]$, for the 4 bounding box offsets, 1 objectness prediction, 2 class predictions and N is the size of the scale.

The total of squared error loss is employed during training for the calculation of the regression loss. If the ground truth for a coordinate is $\hat{t}_*$ and the prediction is t_*, the gradient is given by $\hat{t}_* - t_*$.

Logical regression is used by YOLOv3 to predict an objectness score for each bounding box. If the bounding box prior overlaps a ground truth object more than any other bounding box prior, its value should be 1. The prediction is disregarded if the bounding box prior is not the best but does overlap a ground truth object by greater than a certain threshold. The threshold in this study is set at 0.5. No loss in coordinate or class predictions occurs when a bounding box prior is not allocated to a ground truth object. Utilizing independent logistic classifiers, each box predicts the classes of the bounding box. Binary cross-entropy loss is employed for the class predictions during training.

B. Implementation and Simulation Result

In computation, floating-point operations per second (FLOPS, flops, or flop/s) is a benchmark of computer performance that is important in scientific disciplines requiring floating-point calculations. In certain situations, it is a more precise measurement than calculating instructions per second. The lower value of FLOP means the fewer calculations required per second.

SGD is used to fine-tune the model using an initial learning rate of 0.001, 0.937 momentum, 0.0005 weight decay and batch size of 8. The layers are seeded at random using weights drawn from a zero-mean Gaussian distribution with a standard deviation of 0.009. The IoU threshold is set at 0.5. TensorFlow library is used to implement this model. On an NVIDIA GeForce

RTX 2070 GPU, nodule detection training takes 96 hours over 150 epochs. Fig. 5.17 shows the performance of YOLOv3 where the y axis of first three columns shows the loss, the last two columns show the accuracy and the x axis shows the number of epochs. The result depicts the precision of 72%, recall of 75%, mAP @ 0.5 of 70% and calculated F1 score of 0.73. The F1 score is calculated using the Eq. (5.10). This nodule detector has 30 GFLOPs.

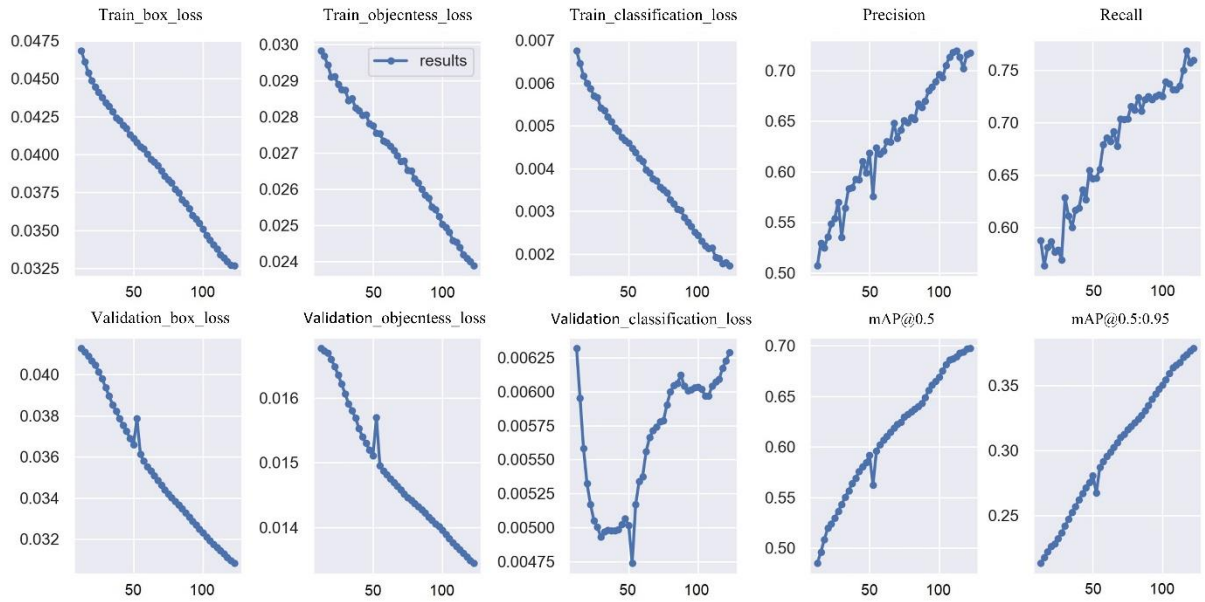
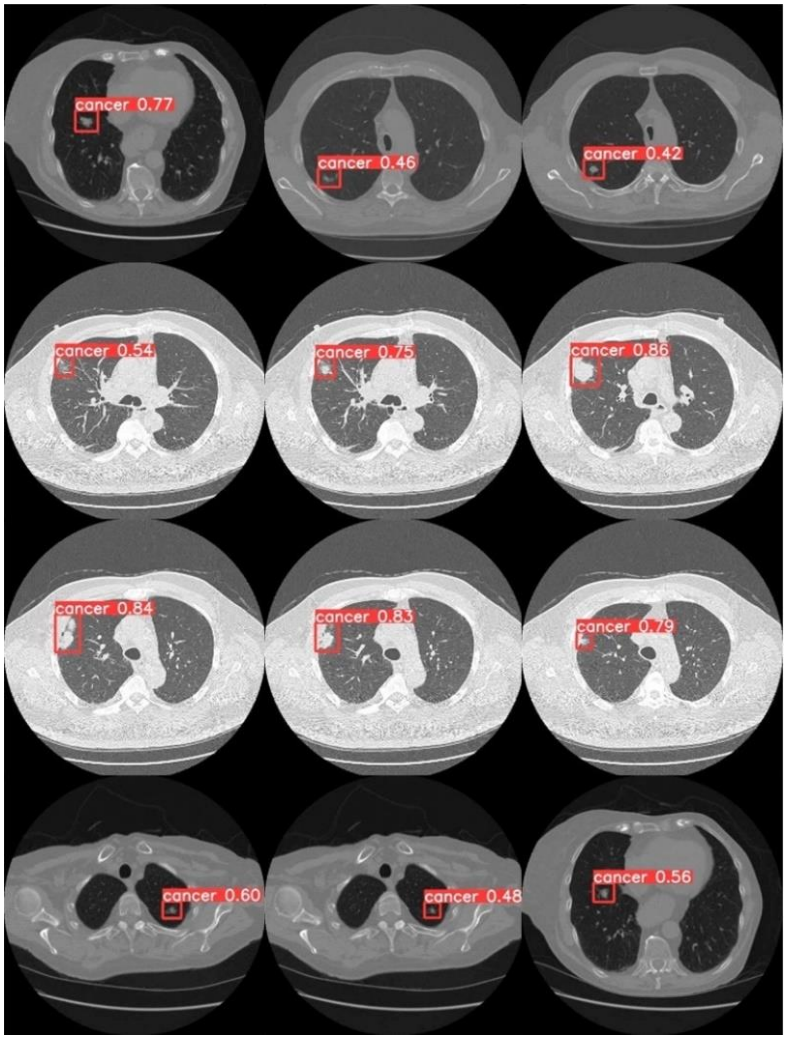


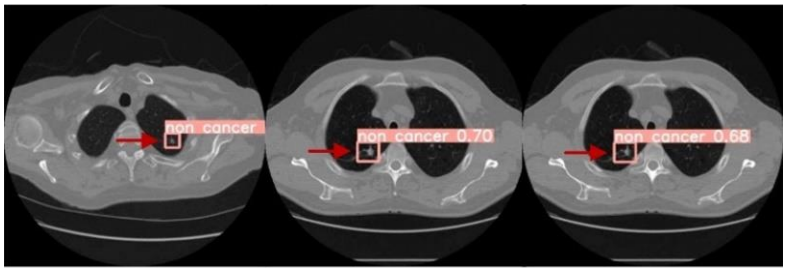
Fig. 5.17 Simulation results obtained on YOLOv3

Fig. 5.18 and Fig. 5.19 show the output of the YOLOv3 algorithms when applied to a sample test images with cancerous and non-cancerous nodules.

The detection results in the Fig. 5.18 show that the YOLOv3 model can detect the cancerous nodules to a less extent while few are undetected and incorrectly labelled. Fig. 5.19 shows, in many images with non-cancer nodules, the model was not able to successfully identify all of them as they are of very tiny size and are even difficult to be detected by the radiologist. The correct location of abnormality in images with non-detection are shown by a circle while the false predictions by an arrow as in Fig 5.18 (b) and Fig. 5.19 (b).

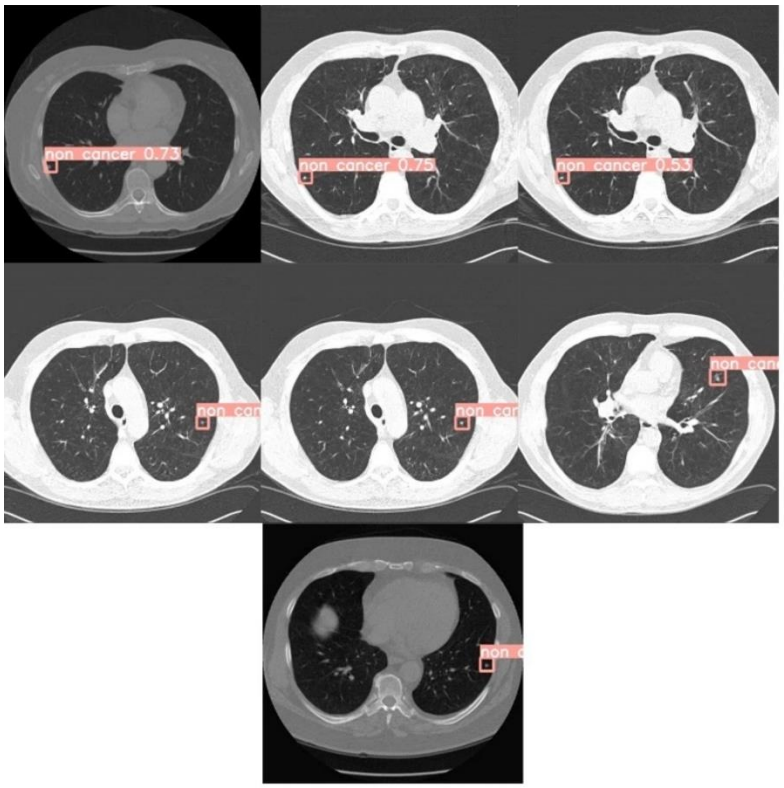


(a)

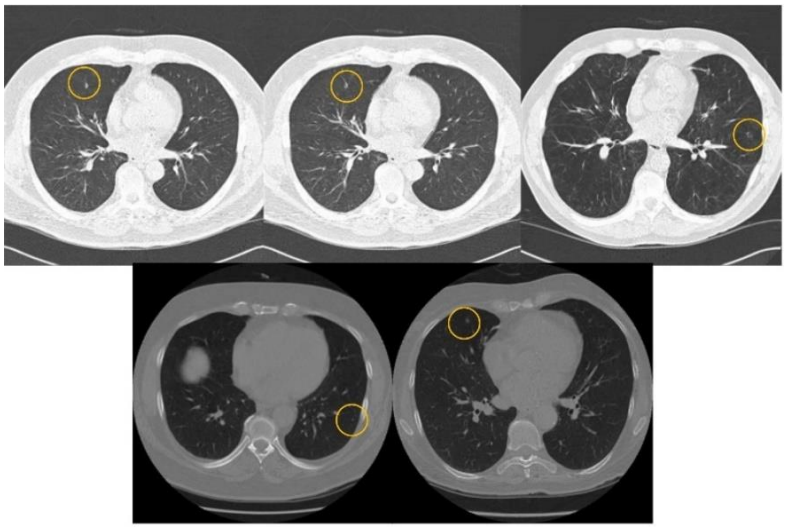


(b)

Fig. 5.18 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using YOLOv3



(a)



(b)

Fig. 5.19 (a) True Prediction (b) False Prediction/non-detection of non-cancerous nodules on test images using YOLOv3

5.3.4 Nodule Detection using YOLOv4

The Darknet53 in the YOLOv3 backbone is computationally costly and suffers from gradient duplication owing to residual layers. As a result, it has been replaced with Cross Stage Partial Darknet53 (CSPDarknet53) in YOLOv4 as an ideal design. The neck area is made up of many bottom-up and top-down routes connected by a PAN mechanism that achieves the goal of enlarging the receptive field. Finally, employing a few convolution layers to detect and locate the nodules along with the likelihood of malignancy.

A. Architecture of YOLOv4

The goal of developing an ideal object detector is to strike the best balance between the input network resolution, the number of convolutional layers, the parameter number (filter size^2 * filters * channel/groups) and the number of layer outputs (filters). As a result, the YOLOv4 model is designed in accordance with this ideal architecture characteristic as illustrated in Fig. 5.20. The scaling parameters in YOLOv4 are adjusted in the backbone and neck areas utilizing CSPDarknet53 and PAN respectively.

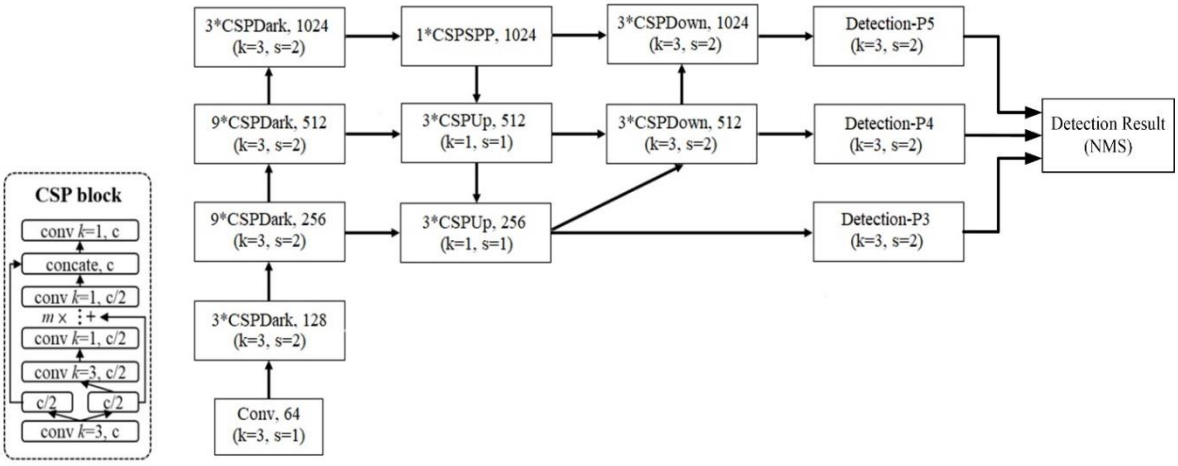


Fig. 5.20 YOLOv4 Architecture [212]

It is generally accepted that the backbone should consist of a model with a bigger receptive field size (with a greater number of convolutional layers 3×3) and a larger number of parameters than the rest of the models in the network. When it comes to object detection the CSPDarknet53 is superior than the Darknet53 utilized in YOLOv3. Each convolution layer is coupled to a batch normalisation (BN) layer and a Mish, a novel self-regularized non-monotonic activation

function which can be mathematically defined as: $f(x) = x\,tanh\big(softplus(x)\big)$. In CSPDarknet53, the total number of convolution layers is 53. The sizes of the convolution layers are 1×1 and 3×3.

The Spatial Pyramid Pooling (SPP) is intended to generate a fixed-size output regardless of the input size, unlike CNN models comprise fully connected layers that only take input images of certain dimensions. In addition, SPP facilitates the extraction of important features by pooling many scales of itself. The SPP block with the CSP structure configuration has been introduced over the CSPDarknet53 because it considerably improves the receptive field, separates out the most important context elements and causes essentially no degradation in network operating performance. The advantage of utilizing a variable receptive field is that it enables to see the full object as well as the viewing the context around it and it increases the number of connections between the image point and the ultimate activation.

The subsequent aim is to pick more blocks for expanding the receptive field and the optimal approach for parameter aggregation from various backbone levels for various detector levels. In place of the FPN used in YOLOv3, PAN network has been implemented as a technique of parameter aggregation.

The final objective function is Bounding Box regression (BBox). The conventional object detector typically employs Mean Square Error (MSE) to directly perform regression on the centre point coordinates and height and width of the BBox, i.e., $\{x_{xentre}, y_{xentre}, w, h\}$ or the upper left point and the lower right point, i.e., $\{x_{top\_left}, y_{top\_left}, x_{bottom\_right},\ y_{bottom\_right}\}$. In anchor-based method used in YOLOv4, the corresponding offset has been estimated, for example

$\{x_{centre\_offset}, y_{centre\_offset}, w_{offset}, h_{offset}\}$ and

$\{x_{top\_left\_offset}, y_{top\_left\_offset}, x_{bottom\_right\_offset},\ y_{bottom\_right\_offset}\}$. For performing bounding box regression, YOLOv4 uses IoU loss which puts the coverage of predicted BBox area and ground truth BBox area into consideration. The Binary cross entropy is used to calculate the objectness and classification scores.

B. Implementation and Simulation Result

YOLOv4 model is trained using the pretrained checkpoint and the adopted tool is SGD optimizer. The default IoU threshold is set to 0.6. The training time required for nodule detector is 144 hours on a NVIDIA GeForce RTX 2070 GPU for 150 epochs. The hyper-parameters considered are as follows: the batch size of 8; the polynomial decay learning rate scheduling strategy is adopted with initial learning rate 0.01; the warm-up epoch is 3; the momentum and weight decay are respectively set as 0.937 and 0.0005.

Fig. 5.21 shows the performance of YOLOv4 where the y axis of first three columns shows the loss and the last two columns shows the accuracy and the x axis shows the number of epochs. It can be known that the performance of YOLOv4 having precision of 85%, recall of 70%, mAP of 80% and F1 score of 0.76. It requires 65 GFLOPs. This proves the effectiveness of YOLOv4 as compared to YOLOv3.

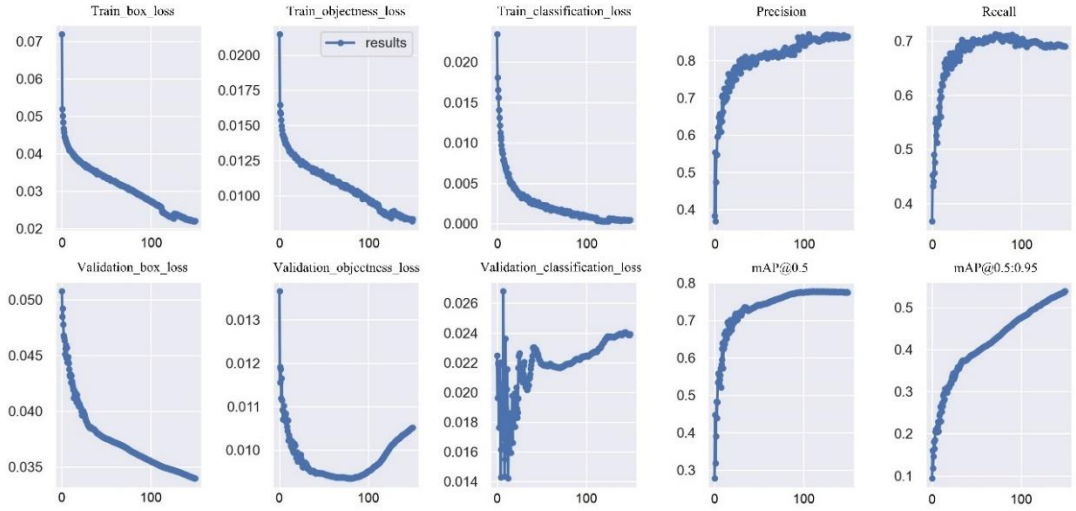


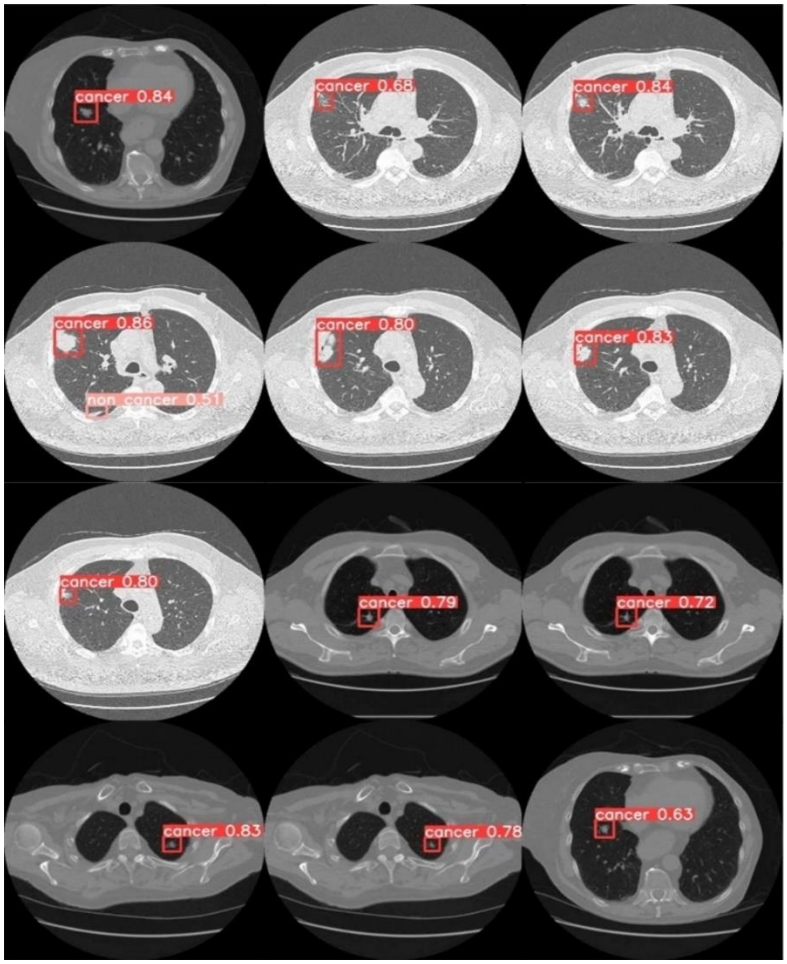
Fig. 5.21 Simulation results obtained on YOLOv4

It is shown that the YOLOv4 object detection neural network based on the CSP approach, scales both up and down and is applicable to detect small and large nodules.

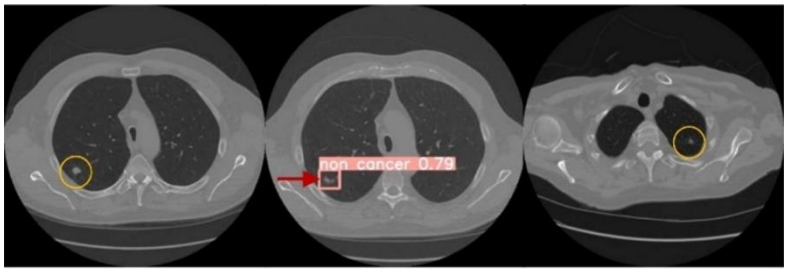
Fig. 5.22 and Fig. 5.23 show the output of the YOLOv4 algorithms when applied to a sample test image with cancerous and non-cancerous nodules are to be detected.

The examples in Fig. 5.22 (a) show that the YOLOv4 model can detect the cancerous nodules to a good extent. While for few non-detections, correct location is shown by a circle and incorrectly prediction is shown by an arrow as shown in Fig. 5.22 (b). Even though it is hard to identify non-cancerous nodule, the model detects them successfully as shown in Fig. 5.23 (a).

In many non-cancer nodules shown in Fig 5.23 (b), the model was not able to successfully identify all of them as they are of very tiny size and the correct locations of non-cancer abnormality is marked by a circle. This detection results on test images proves that YOLOv4 is better as compared to Faster R-CNN, SSD and YOLOv3, but the non-detections and incorrect predictions leads to an implementation of better nodule detector.

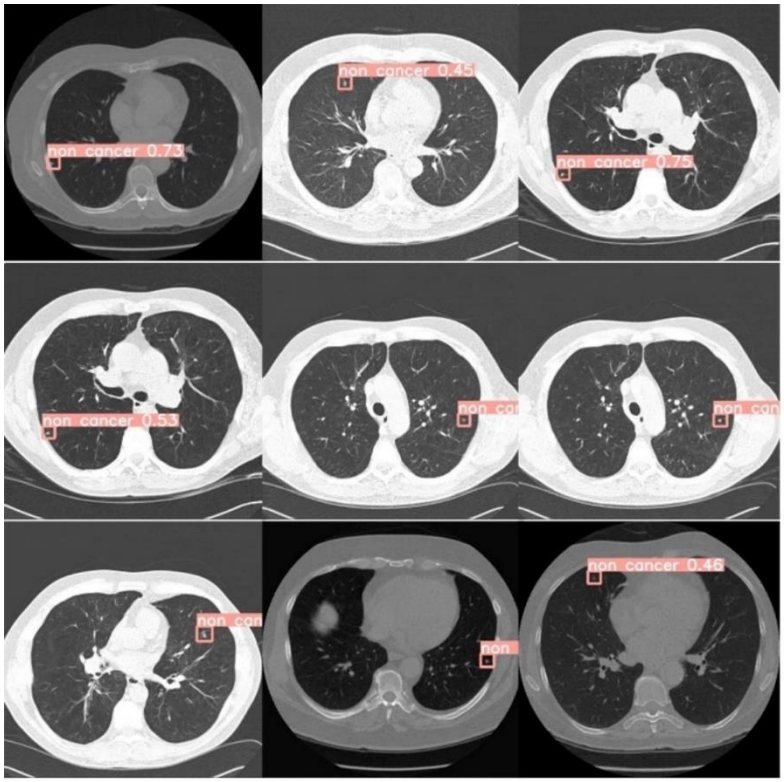


(a)

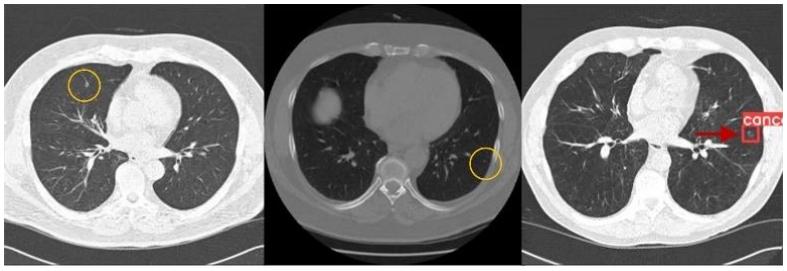


(b)

Fig. 5.22 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using YOLOv4



(a)



(b)

Fig. 5.23 (a) True Prediction (b) False Prediction/non-detection of non-cancerous nodules on test images using YOLOv4

5.3.5 Nodule Detection using YOLOv5

The speed of the YOLOv4 was ideal but it had low performance during training and incorrect predictions on test images. So, there were few changes made in its backbone and neck area, which resulted in the architecture of YOLOv5. The batch normalization is included into the YOLOv5 architecture to stabilize the network and Sigmoid Linear units (SiLu) is used instead of Relu. SiLu is not monotonic and has a soft curve leading to self-stabilizing property as compared to ReLu. The CSPDarknet53 layers and CSP layers in the backbone and neck area of

YOLOv4 are replaced by a combination of Convolution Batch Normalization SiLu (ConvBNSiLU) and C3 layers. The prediction using anchor boxes for nodule identification are generated by the head region.

A. Architecture of YOLOv5

The YOLOv5 network architecture is made up of the backbone, neck and head. In YOLOv5, backbone is created using a combination of ConvBNSiLU and C3. Spatial Pyramid Pooling Fast (SPPF) and a PAN are employed in the neck to aggregate the features. Both regression and classification are conducted in the head region. In Fig. 5.24, the overall structure of the network is shown. ConvBNSiLU is a key block that consists of a Conv layer, a BN layer and a SiLu activation function.

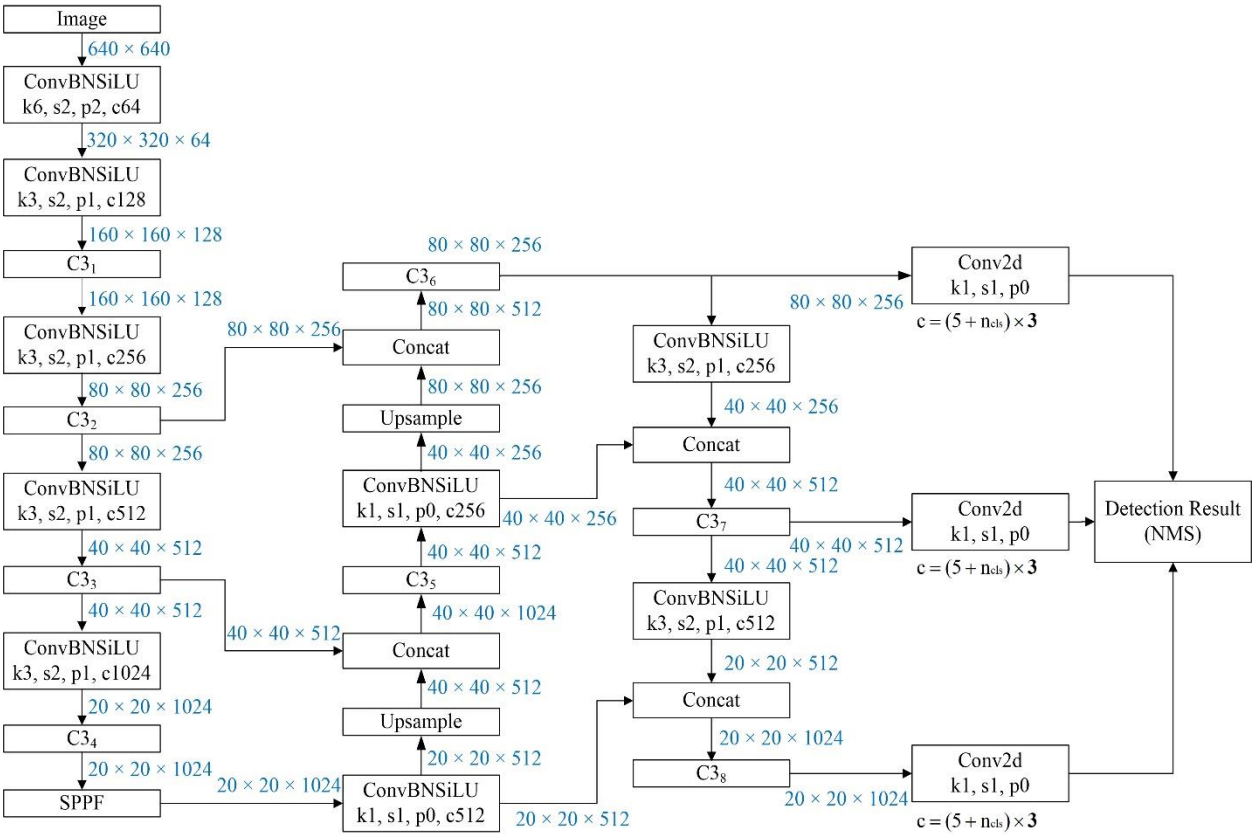


Fig. 5.24 Architecture of YOLOv5 [218]

Fig. 5.25 depicts the structure of C3 layers present in the backbone and the neck region, which consists of three convolution layers in Bottleneck CSP (BNCSP) network structure, where the last convolution in the bottleneck is eliminated to minimize the number of parameters. The C3 blocks is influenced by DenseNet. However, in place of adding the full input and the

output after some CNN layers, the input is divided into two halves. One half is sent via a ConvBNSiLU block, a number of Bottleneck blocks, while the other half is passed directly through a single ConvBNSiLU layer to maintain the features and resolve the problem of duplication of gradients, thereby both the paths are followed by concatenation which combines their features and produces final feature maps using a ConvBNSiLU block.

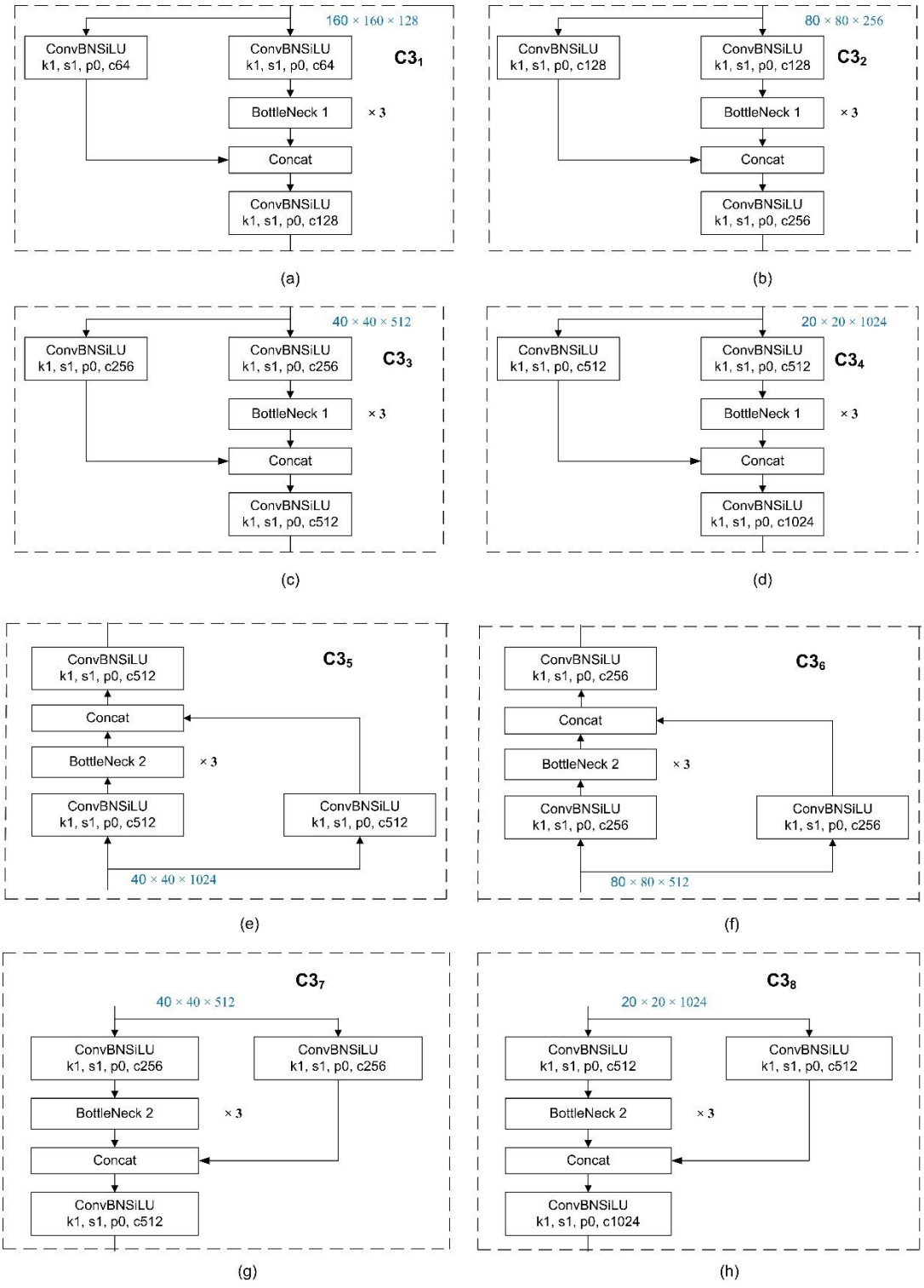


Fig. 5.25 Internal block diagram of $C3_1$ to $C3_8$ of YOLOv5

Fig. 5.26 shows the internal block diagram of bottleneck 1 and bottleneck 2 used inside C3 layers present in the backbone and neck region. In the block diagram the height, width and the

number of filters is denoted by h, w and c, while k, s and p are used for kernel size, stride and padding values.

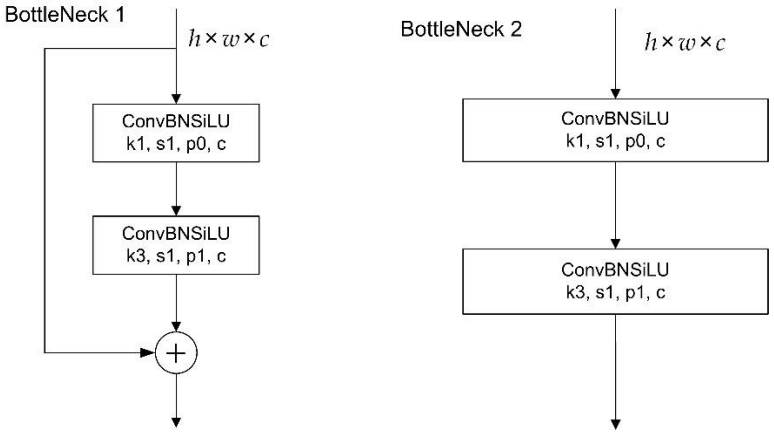


Fig. 5.26 Internal block diagram BottleNeck 1 and BottleNeck 2

The YOLOv5 backbone's output feature maps are sent to an SPPF block, which increases the receptive field and separates out the most essential features using the CSP structure. Fig. 5.27 depicts the underlying structure of SPPF. Before forwarding to the feature aggregation block in the neck, SPPF with smaller kernels is inserted.

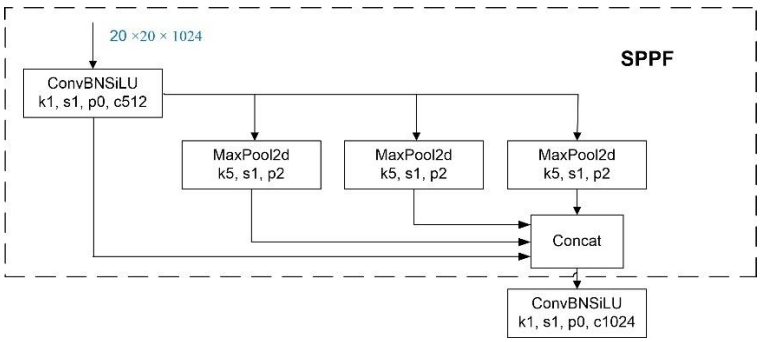


Fig. 5.27 Internal block diagram of SPPF of YOLOv5

The YOLOv5 object detector's backbone is made up of multiple layers. As the layers grow deeper, the spatial resolution of feature maps decreases due to down-sampling, resulting in a loss of spatial information as well as fine-grained features. To maintain these fine-grained features, the PAN proposes a bottom-up augmentation method in addition to the top-down path utilized in FPN. Furthermore, the element-wise addition operation is substituted by concatenation when connecting the feature maps to the lateral architecture. The PAN layer produces three output feature maps, which correspond to feature scales of $80 \times 80 \times 256$,

$40 \times 40 \times 512$, $20 \times 20 \times 1024$. As a result, the convolution layers present in the detector layer of the head detect tiny nodules as well as big nodules.

B. Implementation and Simulation Result

YOLOv5 model is trained using the pretrained checkpoint and the adopted tool is SGD optimizer. The default IoU threshold is set to 0.65. The hyperparameters considered are as follows: the batch size of 8; the polynomial decay learning rate scheduling strategy is adopted with initial learning rate 0.01 and weight decay of 0.005. A momentum of 0.937 is used in the first three warming-up epochs. After that, the momentum is changed to 0.8.

Fig. 5.28 shows the simulation results obtained on YOLOv5 where the y axis of first three columns shows the loss and the last two columns shows the accuracy and the x axis shows the number of epochs. The result shows that, the YOLOv5 model had a precision of 90%, recall of 80%, mAP of 85% and F1 score of 0.84. This proves the effectiveness of YOLOv5 in achieving the best performance but have high computation complexity because it contains 444 layers, 86,267,620 parameters, and 217 GFLOPs. Moreover, the training time required for YOLOv5 is 288 hours on a NVIDIA GeForce RTX 2070 GPU for 150 epochs.

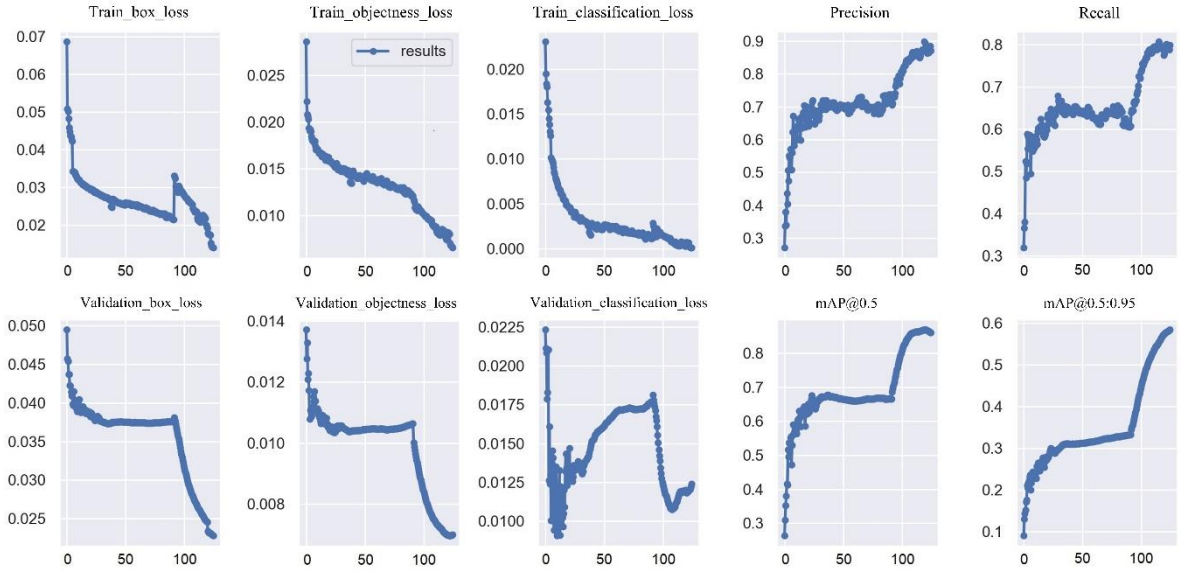
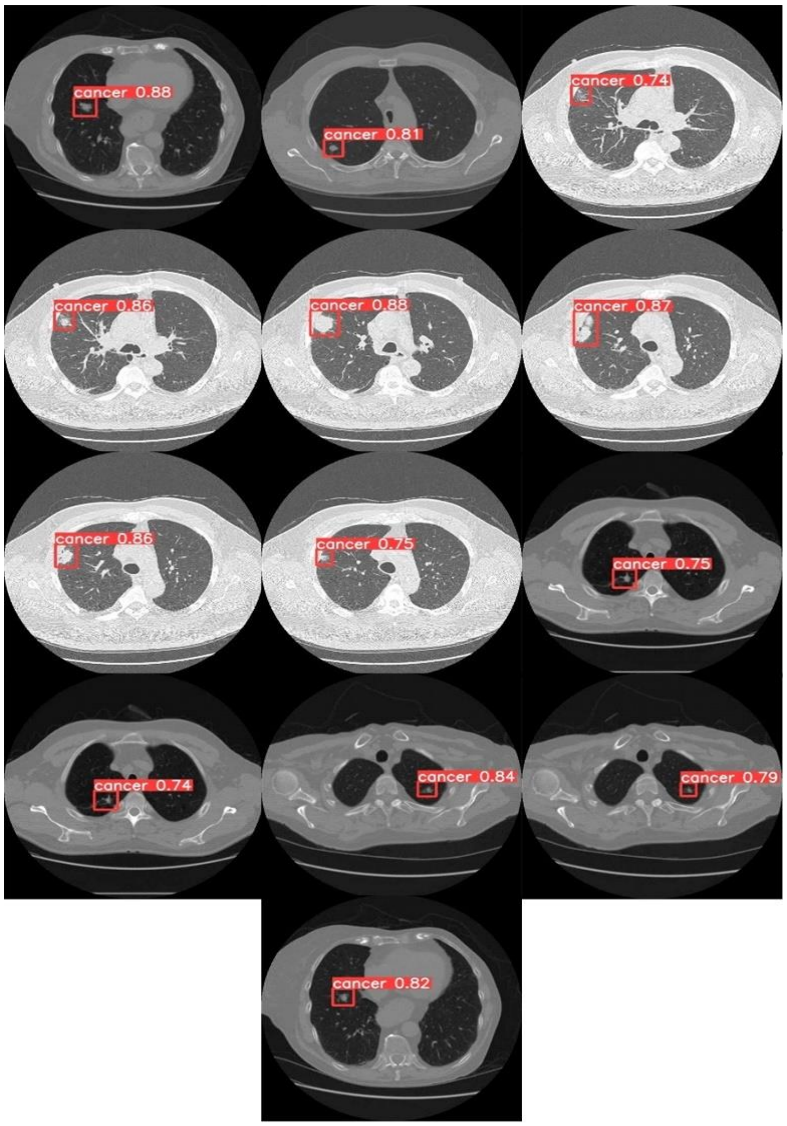


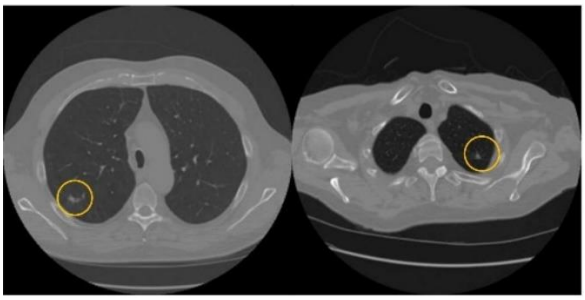
Fig. 5.28 Simulation results obtained on YOLOv5

Fig. 5.29 and Fig. 5.30 show the output of the YOLOv5 algorithms when applied to a sample test images with cancerous and non-cancerous nodules.

The Fig. 5.29 (a) show that the YOLOv5 model can detect the cancerous nodules to a certain extent while few are undetected and incorrectly predicted as shown in Fig. 5.29 (b) where the correct locations are shown by circle and incorrect class predictions are shown by arrow. Even though it is hard to identify non-cancer nodule, the model detects them successfully as shown in Fig. 5.30 (a). In some non-cancerous nodule, the model was not able to successfully identify all of them as they are of very small size as shown by circle in Fig. 5.30 (b). However, this is a situation that is difficult to detect even by the radiologist.

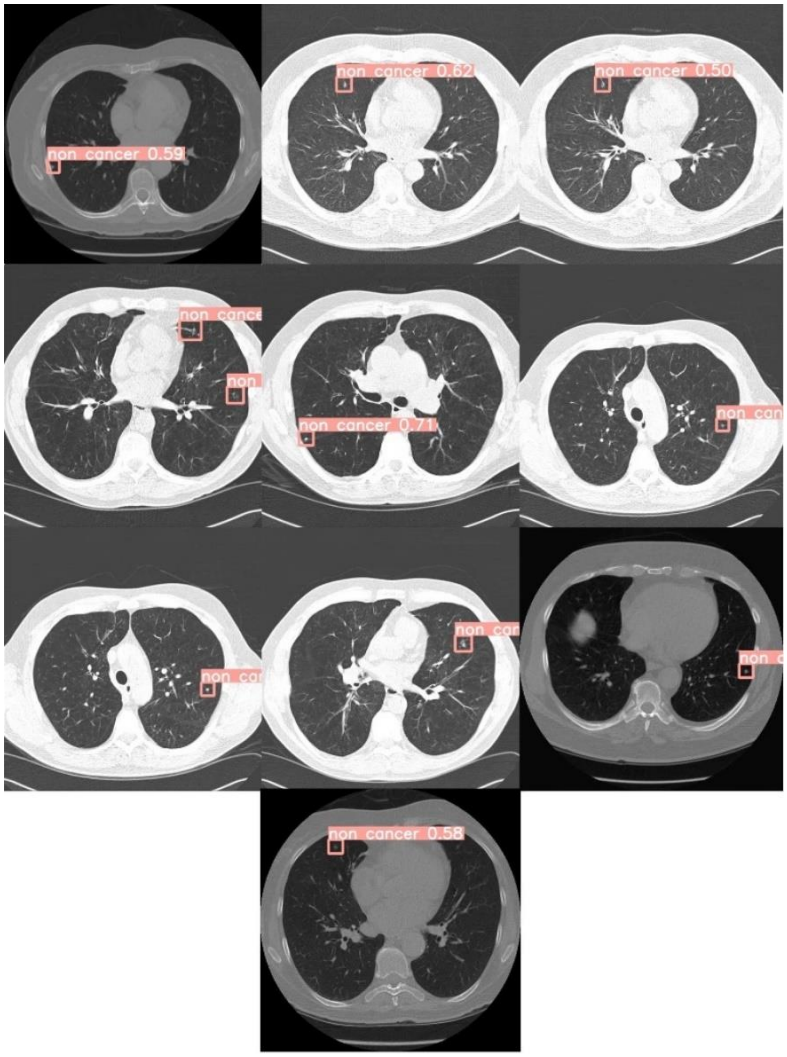


(a)

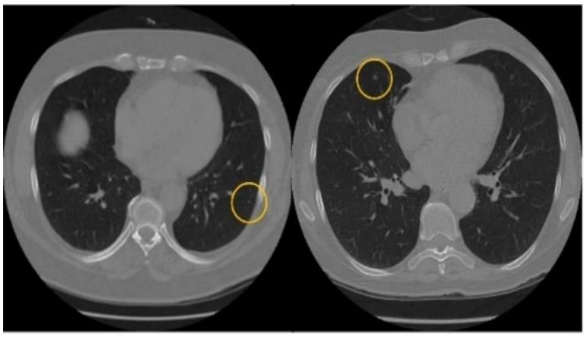


(b)

Fig. 5.29 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using YOLOv5



(a)



(b)

Fig. 5.30 (a) True Prediction (b) False Prediction/non-detection of non-cancerous nodules on test images using YOLOv5

5.4 Comparison of Implemented YOLO Models

The aim is to build a standalone module that runs the deep-learning algorithm to detect cancer and non-cancer nodules on lung CT scans. Because of the good detection speed and accuracy, the YOLOv3, YOLOv4 and YOLOv5 are selected in real-time applications and their performance and speed are compared to investigate which algorithm performs best for nodule detection. There are 15 different lung CT datasets utilized for training, testing and validation. From the results of investigations, presented in Table 5.3, it can be confirmed that all the three algorithms satisfy the requirements.

Table 5.3 Performance of nodule detection models

Model	Performance	Epochs	GFLOPs
YOLOv3	Precision: 72%, Recall: 75%, mAP: 70%, F1 Score:0.73	150	30
YOLOv4	Precision: 85%, Recall: 70%, mAP: 80%, F1 Score:0.77	150	65
YOLOv5	Precision: 90%, Recall: 80%, mAP: 85%, F1 Score:0.85	150	217

It can be seen from the Table 5.3 that YOLOv5 presents higher Precision, Recall, mAP and F1 score compared to YOLOv3 and YOLOv4, which depicts that the YOLOv5 can detect nodules more accurately compared to the other two algorithms. The higher performance of YOLOv5 compared to YOLOv4 is because YOLOv5 uses auto learning bounding boxes which improves the overall performance of the algorithm. The higher performance of YOLOv4 and YOLOv5 compared to YOLOv3 is due to YOLOv3 using Darknet53 which struggles in detecting small objects whereas YOLOv4 and YOLOv5 use CSPdarkent53 that increases the performance significantly.

YOLOv3 has a low precision and recall thus the model needs improvement. For an algorithm to be considered efficient, there must be a balance between precision and recall and that is reflected by the F1 score of the algorithm. The F1 score of YOLOv5 is higher compared to YOLOv3 and YOLOv4.

In this study, it can be seen from the value of Giga Floating Operations Per Second (GFLOP) that YOLOv3 is faster than YOLOv4 and YOLOv5. It can also be seen that YOLOv4′s speed is faster compared to YOLOv5 but slower compared to YOLOv3.

The result shows that YOLOv5 is superior in terms of performance to other YOLO models but the GFLOPs is more than the other two which leads to more time for training. Thus, considering the trade-off between the speed of operation and performance, there are two models proposed based on YOLOv5. These proposed models achieve high performance as well as they are faster in operation and they are described in next chapter.

5.5 Discussion

Real-time detection using deep learning can be used for many localization and identification tasks. In this chapter, several deep neural networks were used to detect and classify cancerous and non-cancerous nodules on the lung CT images. Initially, the networks like Faster R-CNN and SSD were trained for nodule detection and their results were compared. The results showed that Faster R-CNN obtains the highest accuracy of 95.80% while SSD has accuracy of 43.80%. Furthermore, it can be noticed that SSD requires longer training time than Faster R-CNN. The analyses suggests that the examined Faster R-CNN and SSD could localize the nodules in few samples with large size nodules while some were not predicted with correct class and in few cases the nodules were not predicted due to their very small size.

Thereby in next section, YOLO algorithms namely YOLOv3, YOLOv4 and YOLOv5 were trained. These are state-of-the-art real-time deep learning algorithms used for object detection. These algorithms are selected because of their high performance in real-time applications. The main differences between YOLOv3, YOLOv4 and YOLOv5 architecture is that YOLOv3 uses Darknet53 backbone. YOLOv4 architecture uses CSPdarknet53 as a backbone and YOLOv5 uses combination of ConvBNSiLU and C3 as a backbone.

It was found that YOLOv3 is better and faster compared to other deep learning algorithms such as Faster R-CNN and SSD. After training YOLOv4 it was known that it had better performance as compared to YOLOv3; however, the reported performance of YOLOv4 versus YOLOv5 shows that YOLOv5 performs well as compared to YOLOv4.

The comparison of YOLOv3, YOLOv4 and YOLOv5 shows that YOLOv5 outperforms YOLOv4 and YOLOv3 in terms of performance. The computational time of YOLOv3 and YOLOv4 were identical i.e., 30 and 65 GFLOPs respectively and are faster as compared to YOLOv5 which had 217 GFLOPs. However, it can be seen that the models in YOLOv4 and YOLv5 have imbalanced precision and recall as well as high computational time which results in the development of Proposed Model-1 and Proposed Model-2 which are described in next chapter.

CHAPTER-6

Automated Lung Cancer Diagnosis Using Proposed Models

Deep learning-based detection techniques currently come in two types: one-stage detectors and two-stage detectors. A two-stage detector namely Faster R-CNN performs regional proposal and classification consecutively was implemented in the previous chapter. Faster R-CNN, although popular in the past, has an inefficient learning and execution performance issue. In contrast to two-stage detector, in the one-stage detector, a regional proposal and classification are performed simultaneously. In other words, it is a method of solving classification and localization problems at the same time. In chapter 5, there were three models based on YOLO approach were implemented. In the previous chapter, it was known that the YOLOv5 outperforms Faster R-CNN, SSD, YOLOv3 and YOLOv4 in terms of precision, recall, mAP and F1 score. From Chapter-5, it can be observed that the performance achieved for Faster R-CNN, SSD, YOLOv3, YOLOv4 and YOLOv5 were low though the models are complex, have high computation time. So, to overcome these problems, two automated lung cancer diagnosis models have been proposed for the improvement in performance and low computation complexity.

The primary objective of this chapter is to create a fast-operating speed for a nodule detector by optimizing computations by reducing the number of GFLOPs required by the model. It is expected that the suggested model may be readily trained to give accurate, high-quality and convincing results in nodule detection.

Therefore, identifying nodules is a crucial step which requires a high-speed and accurate nodule detector. Therefore, this chapter provides two deep learning-based methods for detecting nodules with adequate accuracy and speed. These approaches use YOLOv5 as a baseline, then make a few modifications to predict more bounding boxes for tiny nodules and use the process in the network's backbone to provide more informative features for subsequent concatenation operations. This chapter discusses the enhancements brought on by every alteration. Ultimately, this model achieves 95% mAP, an increase of 10% over the previous best approach. In order to make the algorithm more user-friendly, a Graphical User Interface (GUI) is constructed to train and validate the implemented models.

The Proposed Model-1 and Proposed Model-2 are trained with parameters set to same value and system configuration with Windows 10 machine equipped with an Nvidia 2070 GPU. The models were implemented using TensorFlow framework. The models are trained with the pretrained checkpoint using the SGD optimizer with a learning rate of 0.01, momentum of 0.937 and weight decay of 0.005. The IoU threshold is set to 0.5. The Proposed Model-1 and Proposed Model-2 requires 120 hours and 96 hours of training respectively on an NVIDIA GeForce RTX 2070 GPU for 150 epochs with a batch size of 8.

6.1 Proposed Model-1: Modified Yolov5 Using Focus and CSP Layers

In the previous chapter it was found that the YOLOv5 performed better but had 217 GFLOPs leading to more computing time. Given this problem, a few changes were made to minimize computational complexity while improving performance.

The fundamental structure of the YOLOv5 is separated into the backbone, neck and the head parts. Backbone is a convolutional neural network designed by the aggregation of image features of varying sizes. Neck is a set of layers that mix and integrate image features prior to prediction, while head takes Neck's features and performs box and class prediction.

There is significant variation between the YOLOv5 and Proposed Model-1. The most notable characteristic of the proposed model is its Focus and CSP layers. This modified model is named as Proposed Model-1.

6.1.1 Architecture of Proposed Model-1

The Proposed Model-1 architecture is shown in Fig. 6.1. The input is resized by the network before providing to the input layer. As illustrated in Fig. 6.1, the first convolution layer of YOLOv5 is replaced by the Focus layer. The Focus layer as shown Fig. 6.2. It stacks nearby features in separate channels thus large-scale feature maps may be down sampled and merged with small-scale feature maps. This increases the identification of tiny and big nodules while simultaneously reducing the number of parameters. The focus layer was designed to minimize layers, parameters, FLOPS and Compute Unified Device Architecture (CUDA) memory while improving performance.

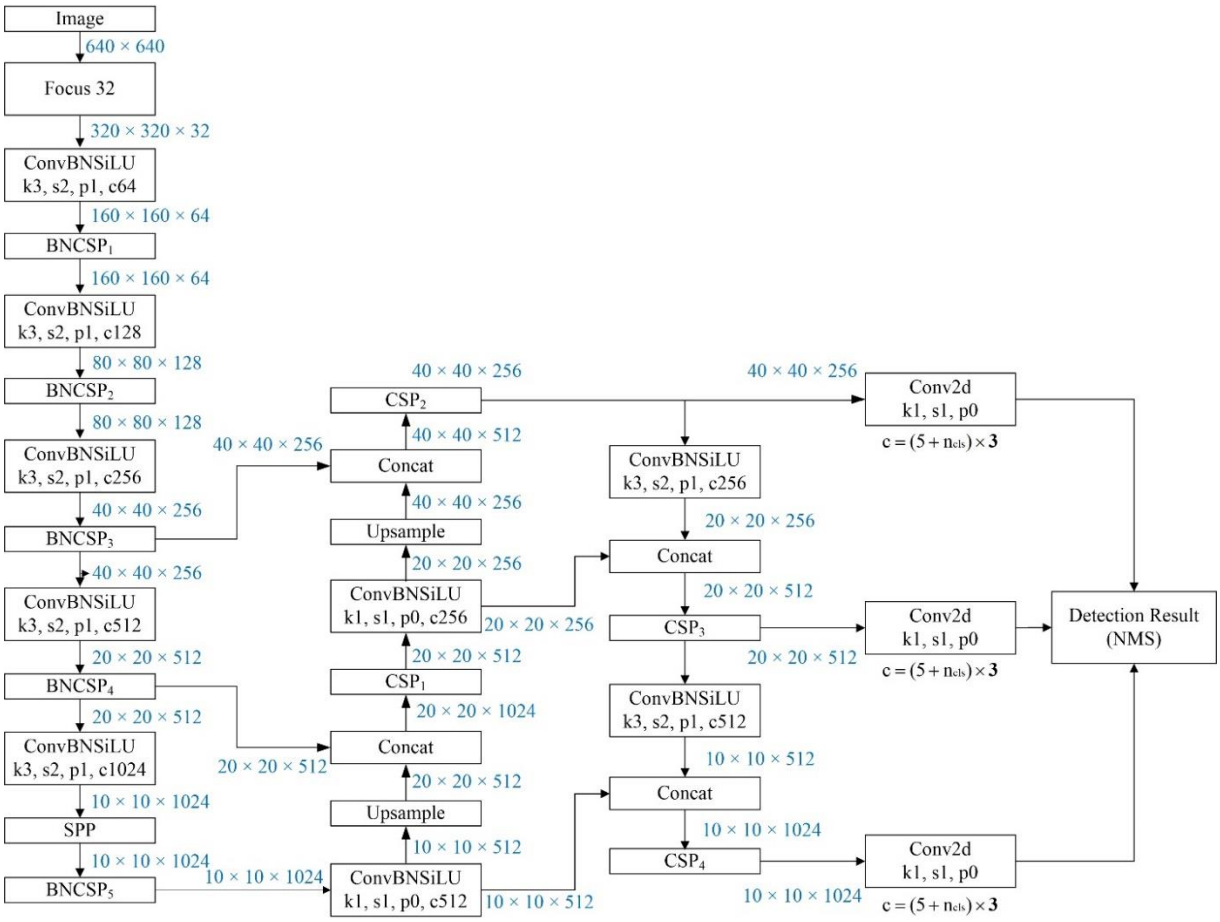


Fig. 6.1 Architecture of Proposed Model-1

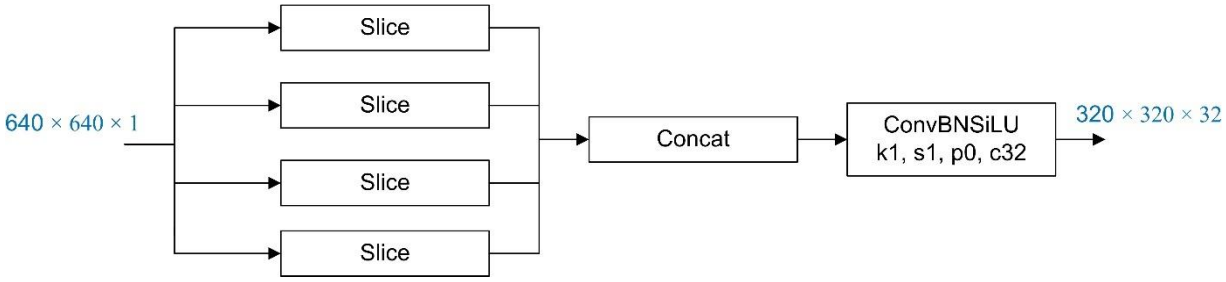


Fig. 6.2 Internal block diagram of Focus layer

A slight modification in the C3 layer in the YOLOv5 backbone is done to obtain Bottleneck CSP (BNCSP) layer illustrated in Fig. 6.3, which aggregates features to improve performance. To optimize functionality, the CSP layer extends to shallow information in the main layer, while the extra feature extraction module of ConvBNSiLu iterates to extract detailed information and functions more effectively. The internal block diagram of BottleNeck 1 used in BNCSP layers are shown in Fig. 5.26 of Chapter 5.

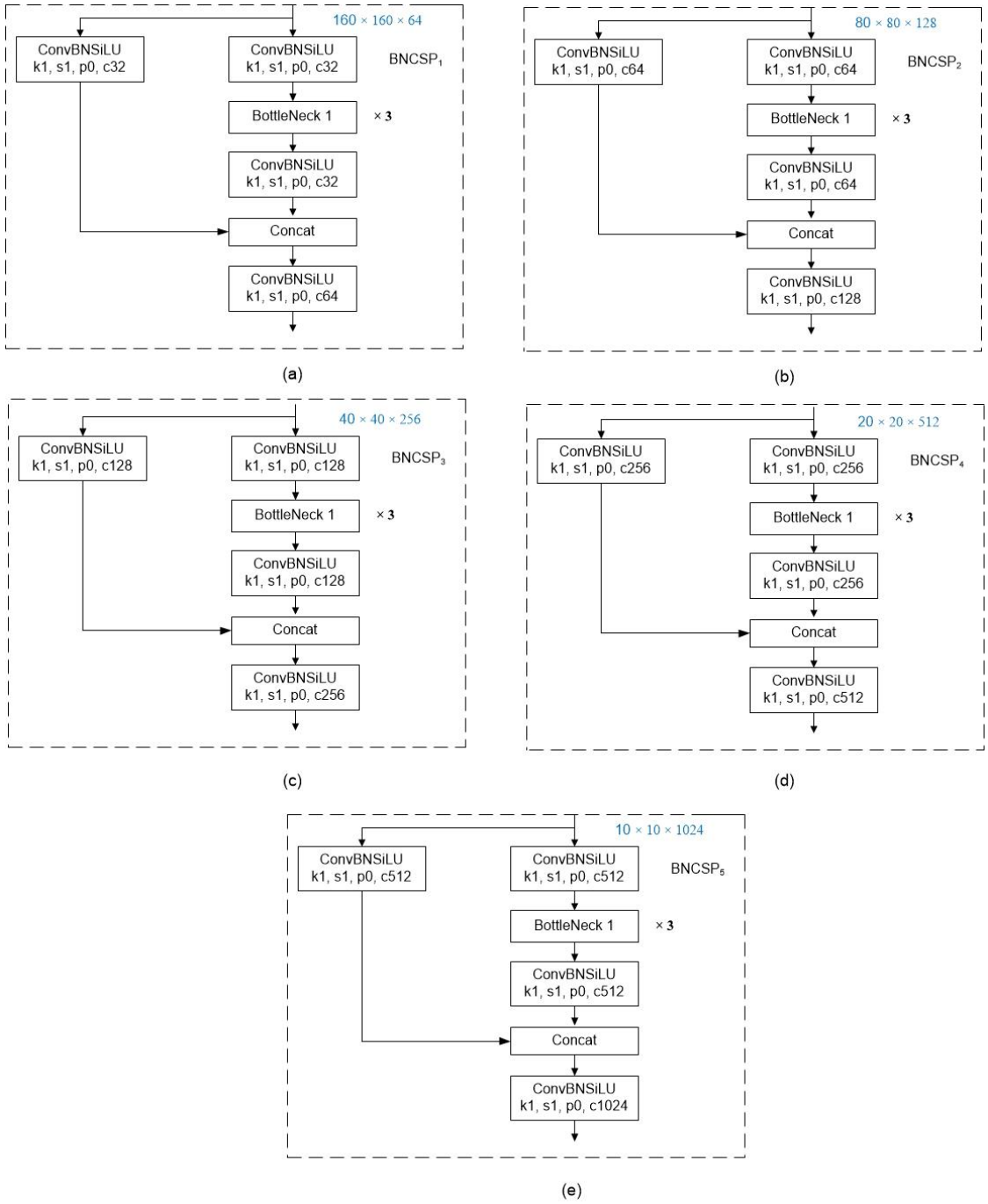


Fig. 6.3 Internal block diagram of BNCSP1 to BNCSP5

Fig. 6.4 depicts the internal structure of Spatial Pyramid Pooling (SPP) to overcome the issues with varying sizes of input image. As illustrated in Fig. 6.5, the C3 layer is replaced by CSP in the neck area of YOLOv5. This CSP structure aids in parameter reduction and addresses gradient duplication in the neck region.

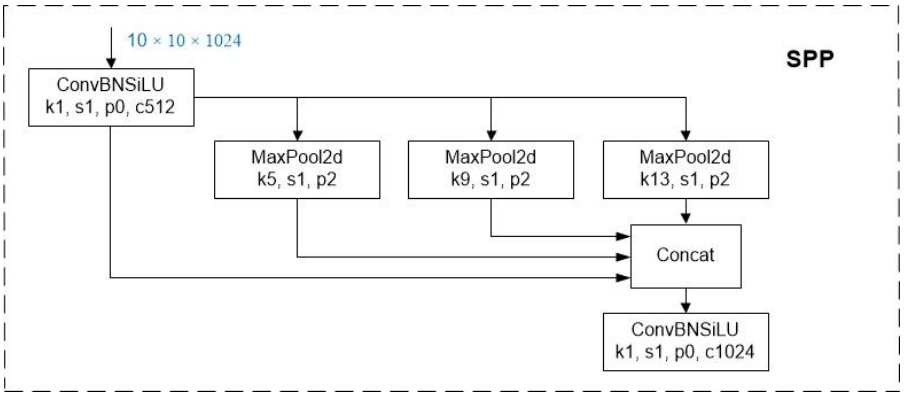


Fig. 6.4 Internal block diagram of SPP

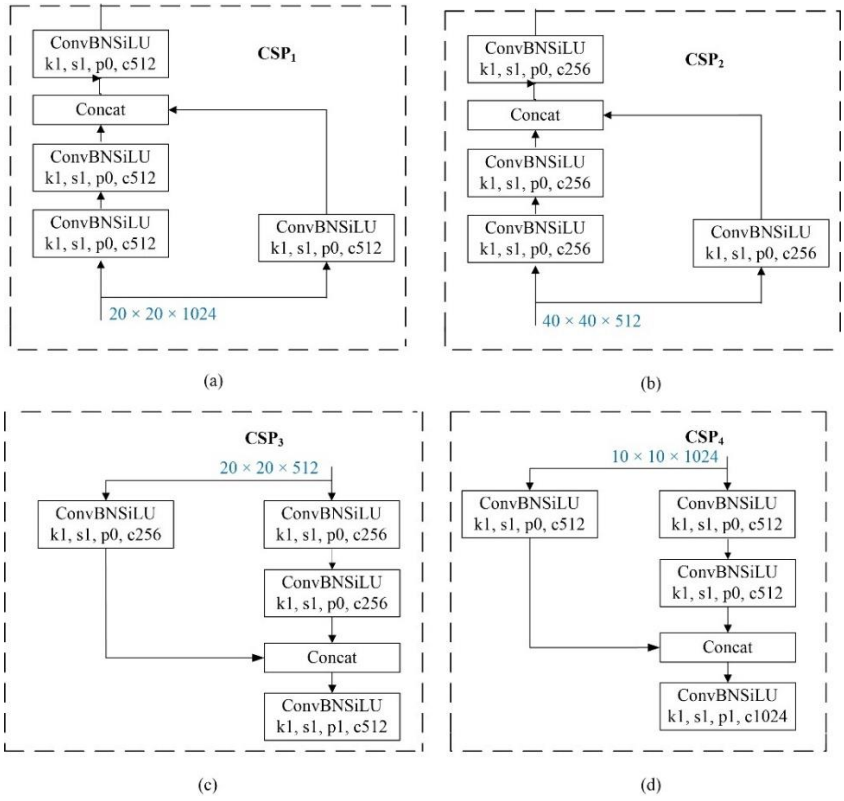


Fig. 6.5 Internal block Diagram of CSP_1 to CSP_4

Finally, the head part is made up of convolutional layers for bounding boxes and class predictions by utilizing anchor boxes. For updating the weights during backpropagation, the validation errors for bounding boxes box, object and class were determined. The Box error is calculated using IoU as the intersection of the anticipated and ground truth normalized by their union. The objectness score, or Obj error, is used to calculate the chance that a certain bounding box is an object. The Cls error is equivalent to a multi-classification score. The Obj and Cls

errors are calculated using the focal loss function, which is an extension of the cross-entropy loss function that reduces the weighting of simple instances while focusing training on hard negatives.

The model employs a compound scaling strategy that evenly increases the resolution, depth and width of all backbone, feature network and box/class prediction networks at the same time, ensuring better performance with limited computational resources.

6.1.2 Simulation Results of Proposed Model-1

The simulation results of Proposed Model-1 are shown in Fig. 6.6. The first three columns indicate model loss components, with train loss in the first row and validation loss in the second. The y axis of first three columns shows the loss and the last two columns shows the accuracy and the x axis shows the number of epochs. The precision with which an algorithm recognizes a nodule is shown by the box loss, objectness loss and classification loss. These data suggest that the two detection classes, cancer and non-cancer, are correctly recognized during training. The performance depicts that the Proposed Model-1 achieves a Precision of 85%, Recall of 82%, mAP of 85% and F1 Score of 0.84. It requires 35.2 GFLOPs.

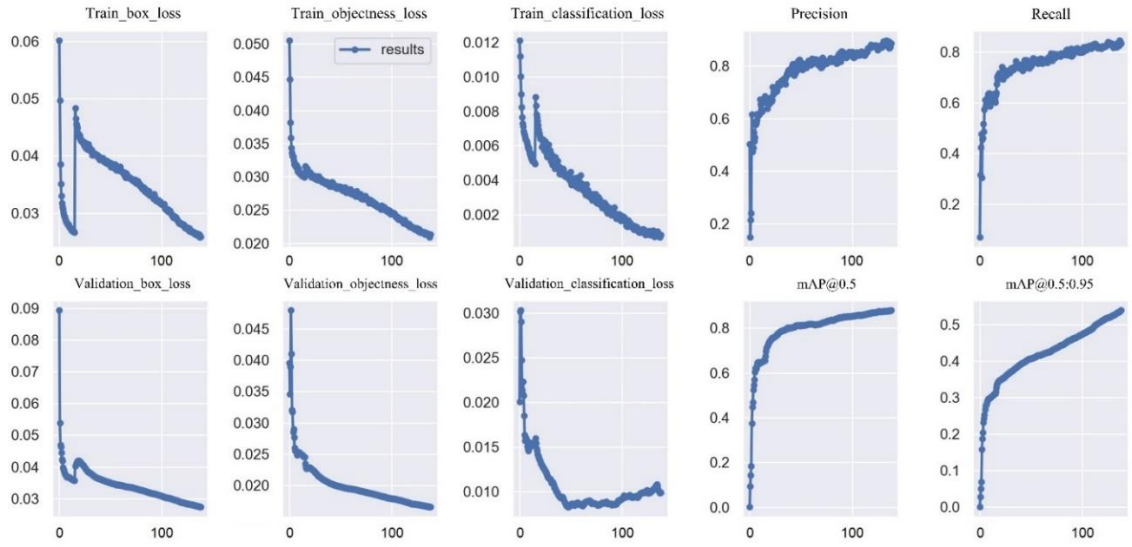
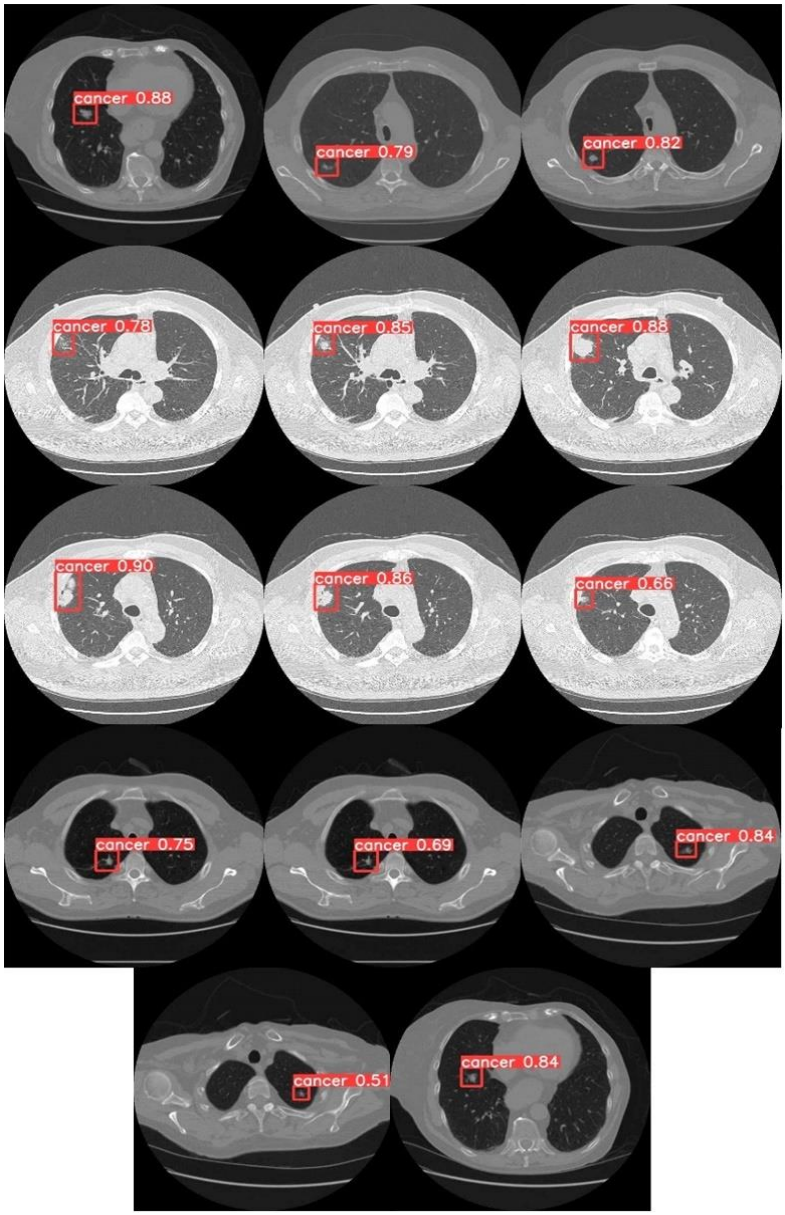


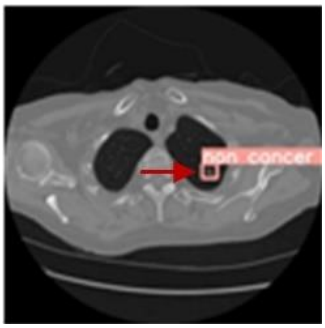
Fig. 6.6 Simulation results obtained on Proposed Model-1

The Fig. 6.7 depicts the prediction performed on Proposed Model-1 using test images containing cancerous nodules. All cancerous nodules are accurately diagnosed when compared to all previously deployed models, including Faster R-CNN, SSD, YOLOv3, YOLOv4 and

YOLOv5. In Fig. 6.8, it can be seen that few non-cancerous nodules are missed in the sample test because they are so tiny. These undetected non-cancerous nodules are annotated by a circle in Fig. 6.8 (b). The results demonstrate Proposed Model-1's inefficiency for non-cancerous datasets. Thus, there is still need for improvement in order to achieve accurate detections which is achieved by Proposed Model-2 described in next section.

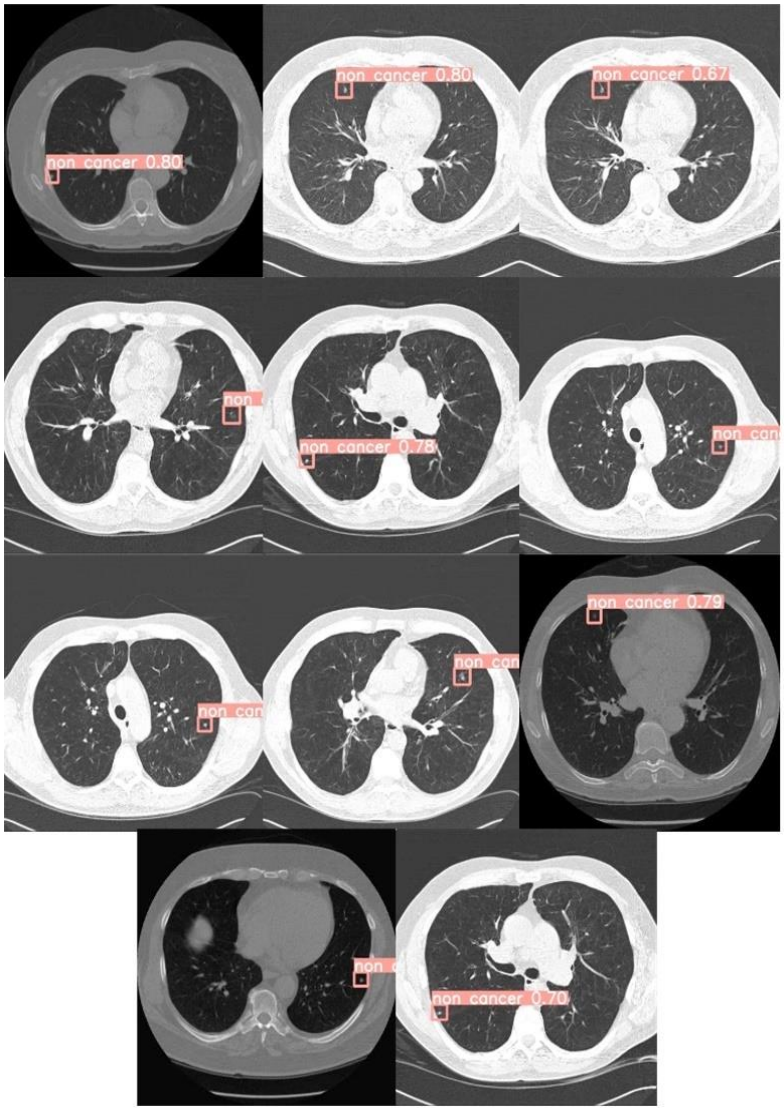


(a)



(b)

Fig. 6.7 (a) True Prediction (b) False Prediction/non-detection of cancerous nodules on test images using Proposed Model-1



(a)

(b)

Fig. 6.8 (a) True Prediction (b) non-detection of non-cancerous nodules on test images using Proposed Model-1

6.2 Proposed Model-2: Modified YOLOv5 Using Depthwise Separable Convolution Layer

Proposed Model-1 described in the previous section is superior in terms of both computation time and performance as compared to the other implemented nodule detectors in this thesis. In order to further minimize computation time i.e., the GFLOPs, the convolution layers in the Proposed Model-1's backbone are replaced with DepthWise Separable Convolution (DWSC) layers also known as DWConv. Convolutional operations using DWSC employ fewer parameters and computations while improving representational efficiency. This new model is named as Proposed Model-2.

6.2.1 Architecture of Proposed Model-2

The conventional convolution process has the effect of filtering features based on the convolutional kernels and merging them to build a new representation. Utilizing factorized convolutions known as DWConv, the filtering and combining phases may be separated into two steps for a significant reduction in computing cost. The DWConv comprises of two layers: depthwise convolutions and pointwise convolutions. In the depth-wise convolutions a single filter is applied to each input channel. The output of the depthwise layer is then linearly combined using pointwise convolution, which is a simple 1×1 convolution as shown in Fig. 6.9.

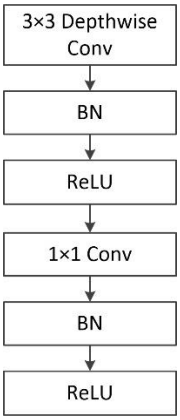


Fig. 6.9 Internal blocks of DWConv

A DWS convolution returns the standard convolutional function with a factorized 2-layer convolutional, 1 layer to space filter, and 1 layer to combine. Thus, depth-wise separable convolutional drastically diminish the unnecessary calculation and model size. By stacking the

improved pyramidal units, gradually learns lower and higher levels of feature abstraction with increasing depth and strengthens the flow and utilization of feature information to achieve a better performance. In the meantime, the gradients disappearance and overfitting phenomena of deep network are alleviated and this is the reason Proposed Model-2 achieved a better performance as compared to Proposed Model-1.

The architecture of Proposed Model-2 is shown in Fig. 6.10. As compared to Proposed Model-1, the only modification carried out is the convolution layers in the backbone are replaced by DWConv whose internal structures are shown in Fig. 6.11. The rest network structure as well as feature sizes remain intact. The prediction of the bounding box and class are carried out by the same manner as the Proposed Model-1.

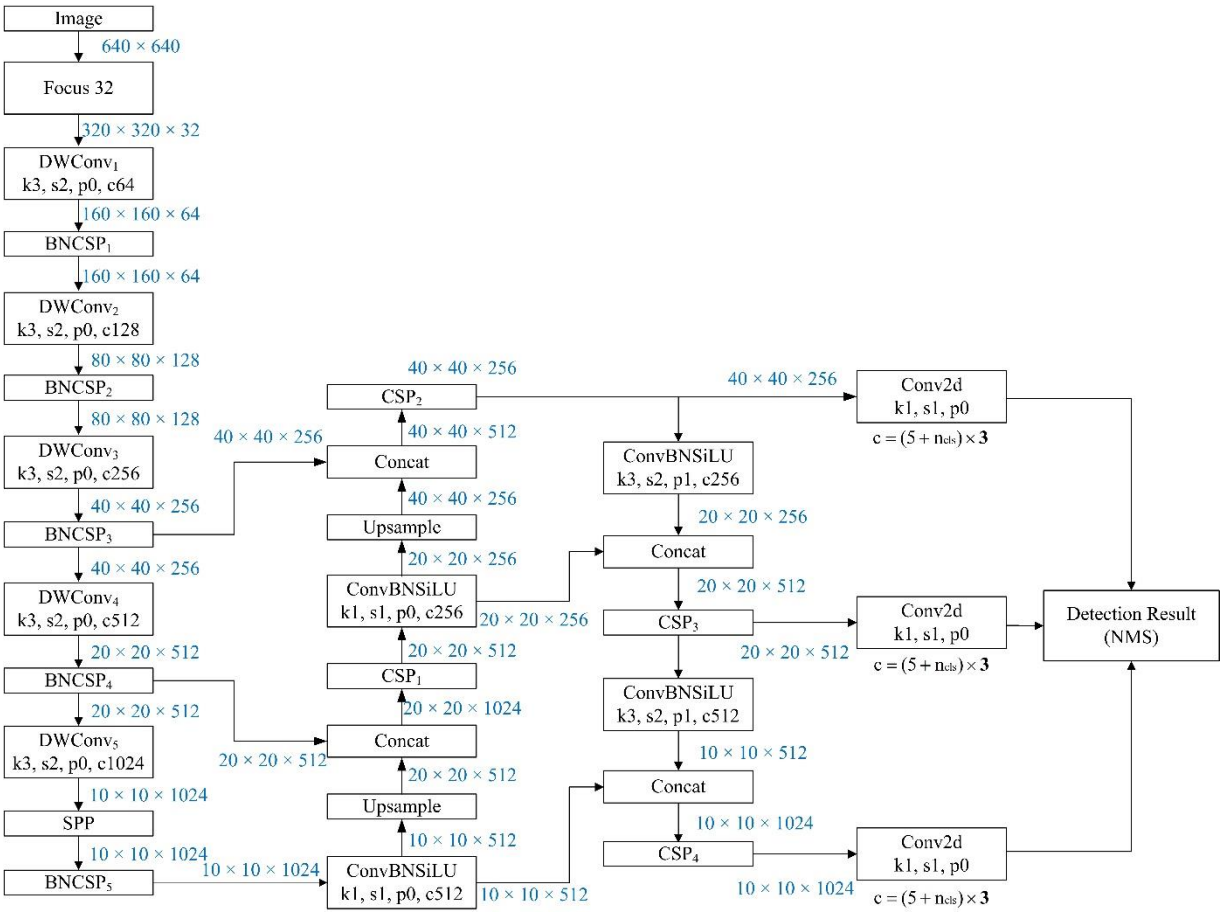


Fig. 6.10 Architecture of Proposed Model-2

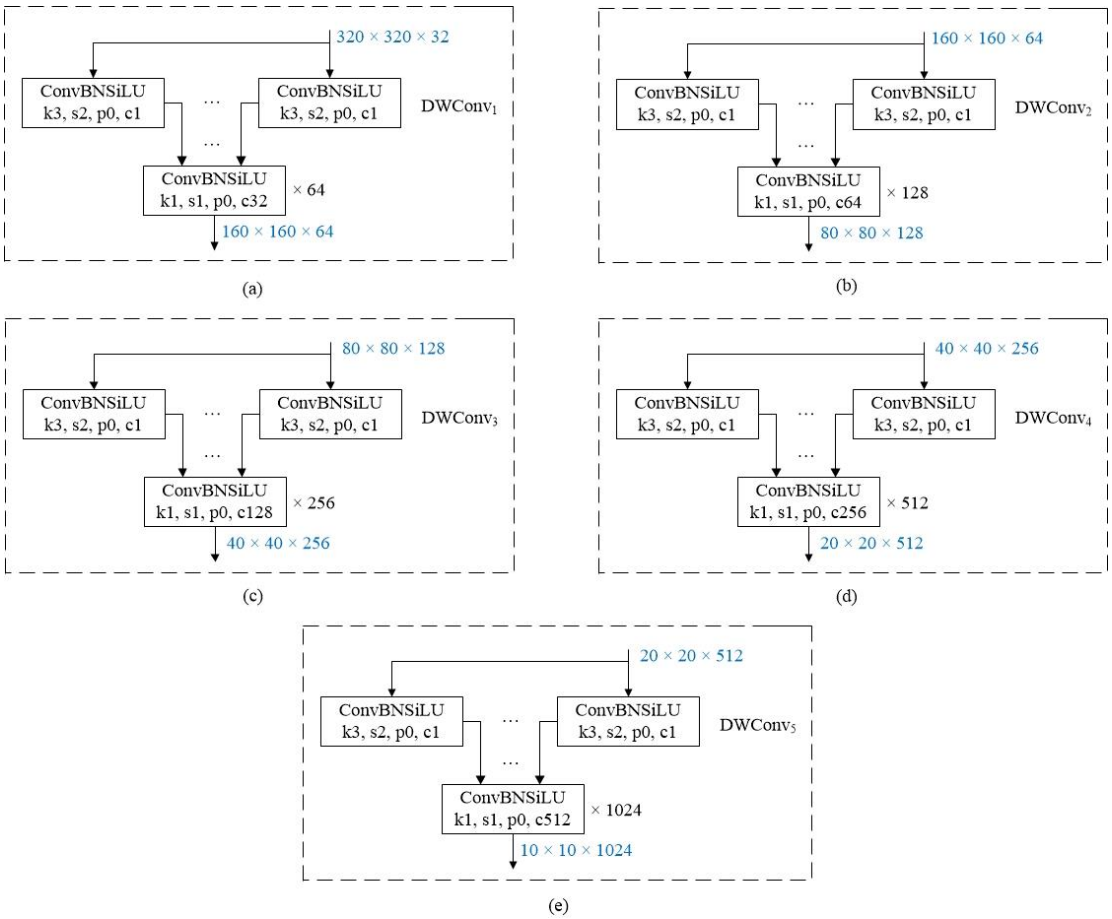


Fig. 6.11 Internal block diagram of $DWConv_1$ to $DWConv_5$

6.2.2 Simulation Results of Proposed Model-2

Fig. 6.12 depicts the performance of Proposed Model-2 with precision of 95%, a recall of 90%, a mAP of 95% and F1 score of 0.92. It requires 30.6 GFLOPs. The model was trained with the epoch set to 150. These results demonstrate that cancer and non-cancer classes are recognized properly throughout training. The box loss, objectness loss and classification loss graph are linear as compared to the previously trained YOLOv3, YOLOv4, YOLOv5 and Proposed Model-1. This signifies the performance of Proposed Model-2 for nodule detection.

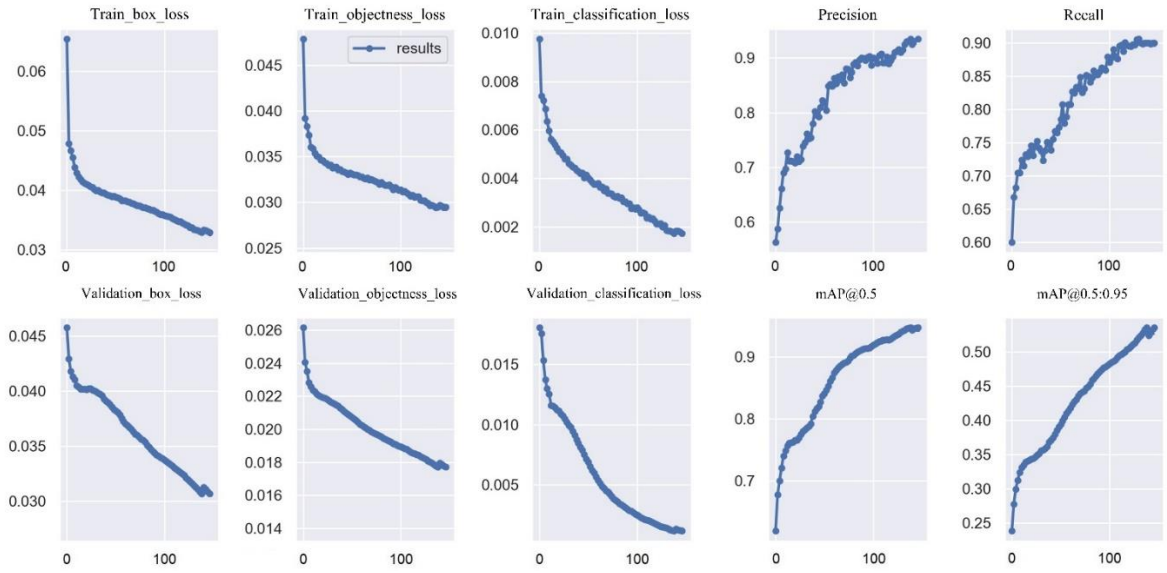


Fig. 6.12 Simulation results obtained on Proposed Model-2

The Fig. 6.13 and Fig. 6.14 depict the prediction made by Proposed Model-2 utilizing test images including cancerous and non-cancerous nodules. All cancerous and non-cancerous nodules are appropriately identified in comparison to all previously deployed models, including Faster R-CNN, SSD, YOLOv3, YOLOv4, YOLOv5 and Proposed Model-1. Even though the nodules are of varying size and types they are detected accurately. This revels the efficiency of Proposed Model-2 for lung cancer diagnosis.

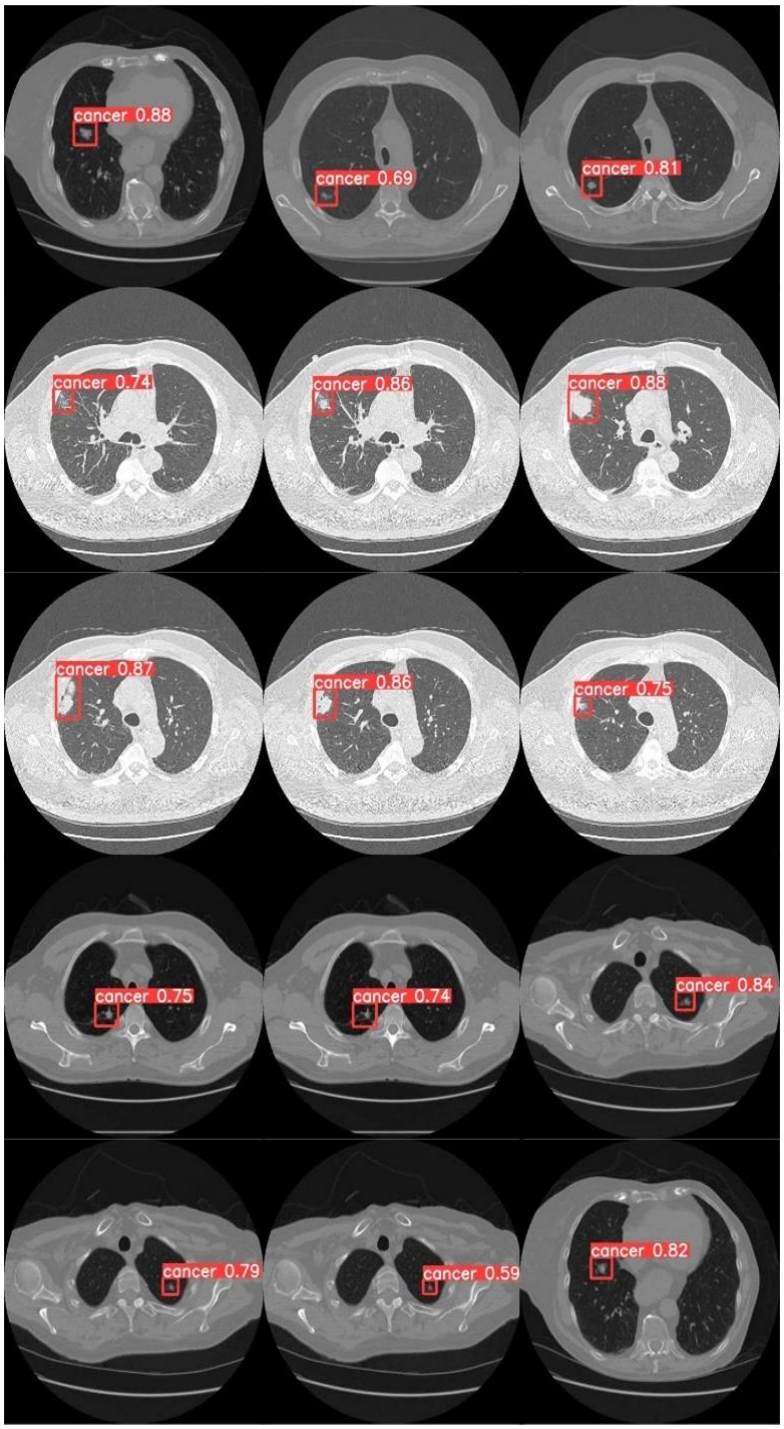


Fig. 6.13 True Prediction of cancerous nodules on test images using Proposed Model-2

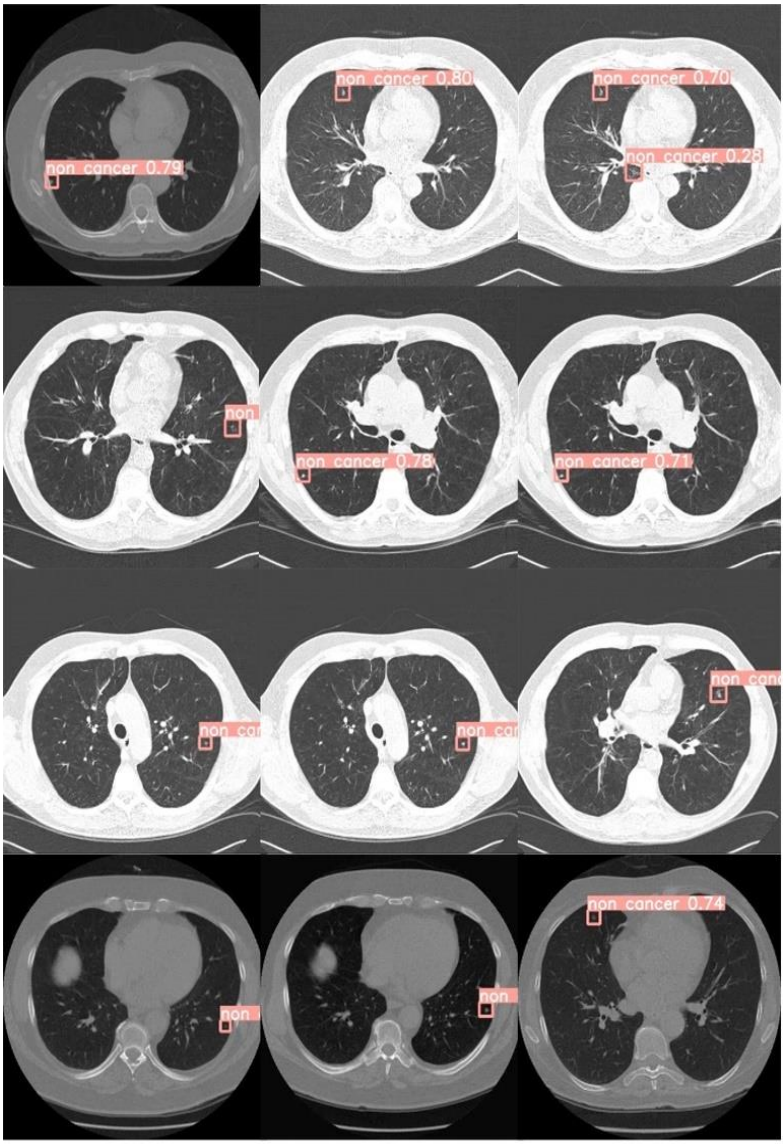


Fig. 6.14 True Prediction of non-cancerous nodules on test images using Proposed Model-2

6.3 Comparison Between YOLOv5, Proposed Model-1 and Proposed Model-2

It can be observed from Table 5.3 of chapter-5 that Yolov5 had the best output performance among YOLOv3, YOLOv4 and YOLOv5. Consequently, comparisons are made between YOLOv5 and proposed models in this chapter. Table 6.1 shows the comparison of YOLOv5, Proposed Model-1 and Proposed Model-2 in terms of precision, recall, mAP and F1 score. Another parameter for comparison is GFLOPs which indicates the computational time. The Precision of 90%, Recall of 80%, mAP of 85% and F1 Score of 0.85 have been achieved in

YOLOv5. The YOLOv5 has 217 GFLOPs so it takes much computational time. Thus, there are two models proposed for performance improvement as well as reduction in the GFLOPs which will reduce the computational complexity and hence the computing time. The Proposed Model-1 is a modification of YOLOv5 by including the Focus and CSP layers. This model achieved Precision of 85%, Recall of 82%, mAP of 85% and F1 Score of 0.84. In order to reduce the computation time of Proposed Model-1, the second model is proposed that is a modification of Proposed Model-1 by replacing convolution layers with DWSC. This Proposed Model-2 outperformed the first one, with Precision of 95%, Recall of 90%, mAP of 95% and F1 Score of 0.92.

Table 6.1 Performance Comparison between YOLOv5, Proposed Model-1 and Proposed Model-2

Model	**Performance**	**Epochs**	**GFLOPs**
YOLOv5	Precision: 90%, Recall: 80%, mAP: 85%, F1 Score: 0.85	150	217
Proposed Model-1	Precision:85%, Recall: 82%, mAP: 85%, F1 Score: 0.84	150	35.2
Proposed Model-2	Precision: 95%, Recall: 90%, mAP: 95%, F1 Score: 0.92	150	30.6

The comparison in the table is based on the best 150 epoch outcome values. The values of mAP were compared in order to objectively assess the performance of the models. The mAP value of the original YOLOv5 model is 85%, Proposed Model-1 is 85% and Proposed Model-2 is 95%. Also, the GFLOPs value revels that the Proposed Model-2 takes least computational time among all the implemented nodule detectors in this work. Overall, the Proposed Model-2 outperforms the original YOLOv5 model. As a result, the final prediction was based on the weight acquired from the trained Proposed Model-2, which was deemed to have the best performance. The sample test images were used for detections on all five YOLO implemented models in this work. The detections are illustrated in Fig. 6.15 and Fig. 6.16, where section (a) indicates the precise location of abnormality denoted by a circle. The Fig. 6.15 depicts the inefficiency of YOLOv3 and YOLOv4 in detecting cancerous nodules, while YOLOv5 and Proposed Model-1 failed to identify a few cancerous nodules. The fact that Proposed Model-2 could identify all cancerous nodules completely indicates its efficiency in predicting cancerous nodules. The Fig. 6.16 depicts the identification of non-cancerous nodules in the five YOLO models that were used. The predictions indicates that YOLOv3, YOLOv4 and YOLOv5 could

not identify non-cancerous nodules in many instances due to their tiny size. While both Proposed Models could detect all of them.

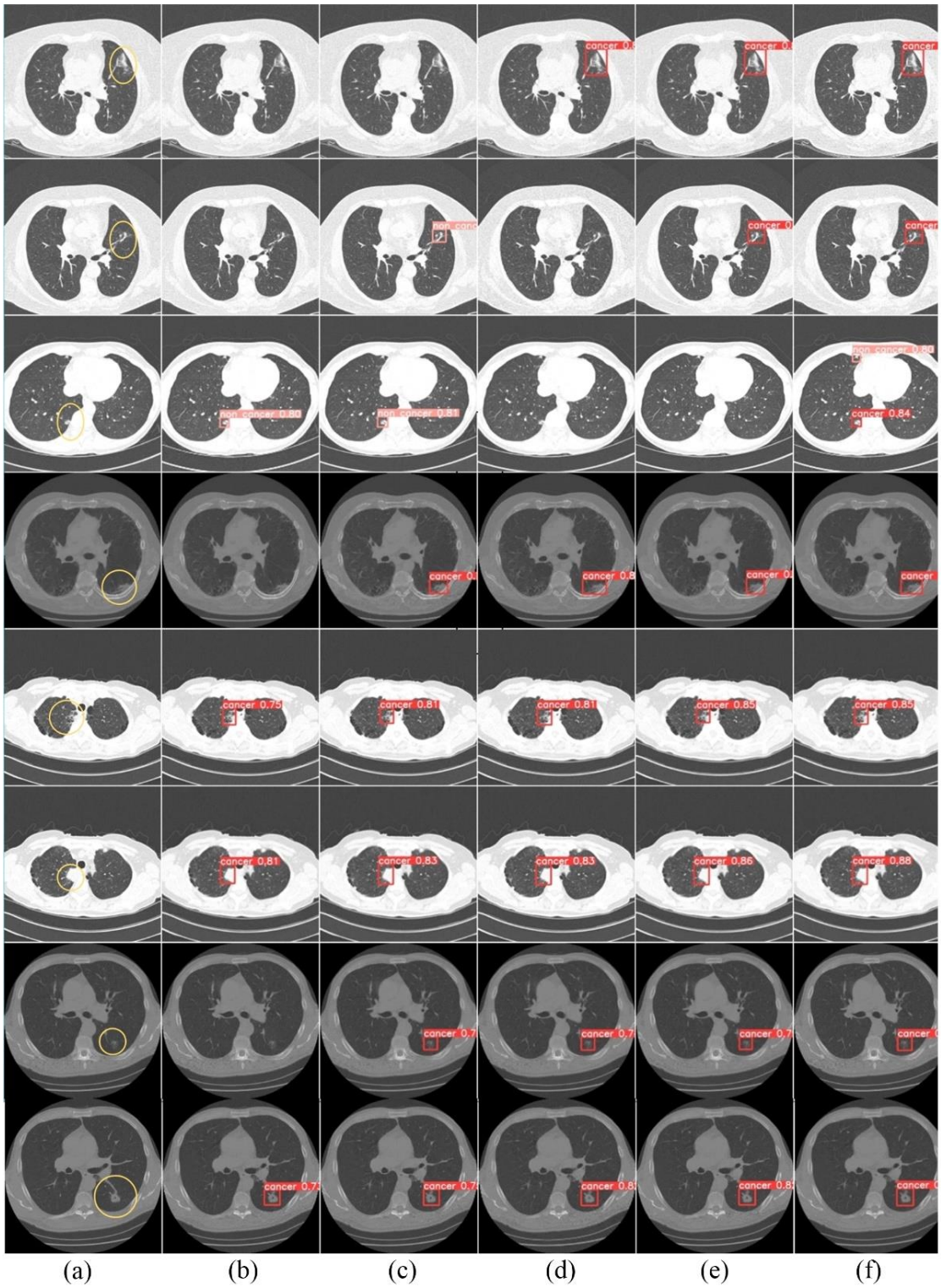


Fig. 6.15 (a) Labelled image (b) Detection on YOLOv3 (c) Detection on YOLOv4 (d) Detection on YOLOv5 (e) Detection on Proposed Model-1 (f) Detection on Proposed Model-2 for Cancerous Nodules

Fig. 6.16 (a) Labelled image (b) Detection on YOLOv3 (c) Detection on YOLOv4 (d) Detection on YOLOv5 (e) Detection on Proposed Model-1 (f) Detection on Proposed Model-2 for Non-Cancerous Nodules

6.4 Validation of Proposed Model-2

The comparison of the predictions of all five models determined that Proposed Model-2 is the best and the validation of Proposed Model-2 was performed using NLST dataset that had never been presented to the model previously. There were 100 NLST CT scans with an equal number of cancer and non-cancer scans employed for validation of Proposed Model-2 as shown in Table 6.2 and Table 6.4 respectively. This prediction is compared to the ground truth nodule location, margin and size supplied in the annotation file. It was discovered from Table 6.3 and Table 6.5 respectively that 92% of cancer predictions and 90% of non-cancer predictions are correct as per actual label. Fig. 6.17 and Fig. 6.18 illustrate image predictions demonstrating that nodules of variable size, shape, location, contrast and slices from scanners with different radiological parameter settings are well identified.

Table 6.2 Predictions on Cancer test datasets applied to Proposed Model-2

Patient ID	True Label	Prediction Label
125794	Cancer	Cancer
125797	Cancer	Cancer
125818	Cancer	Cancer
126325	Cancer	Cancer
126453	Cancer	Non-Cancer
126514	Cancer	Non-Cancer
126558	Cancer	Cancer
126625	Cancer	Cancer
126787	Cancer	Cancer
126917	Cancer	Cancer
101907	Cancer	Cancer
103787	Cancer	Cancer
103957	Cancer	Cancer
104114	Cancer	Cancer
104964	Cancer	Cancer
105120	Cancer	Cancer
105121	Cancer	Cancer
105384	Cancer	Cancer
108395	Cancer	Not Detected
108730	Cancer	Cancer
109225	Cancer	Cancer
112146	Cancer	Cancer
113235	Cancer	Cancer

113774	Cancer	Cancer
114508	Cancer	Cancer
114828	Cancer	Cancer
114955	Cancer	Cancer
115684	Cancer	Cancer
115914	Cancer	Cancer
115943	Cancer	Cancer
116009	Cancer	Cancer
116576	Cancer	Cancer
116601	Cancer	Cancer
117450	Cancer	Cancer
117708	Cancer	Cancer
117777	Cancer	Cancer
118742	Cancer	Cancer
118919	Cancer	Cancer
118990	Cancer	Not Detected
119684	Cancer	Cancer
125848	Cancer	Cancer
126055	Cancer	Cancer
126482	Cancer	Cancer
126703	Cancer	Cancer
127187	Cancer	Cancer
131582	Cancer	Cancer
132689	Cancer	Cancer
204366	Cancer	Cancer
205099	Cancer	Cancer
205501	Cancer	Cancer

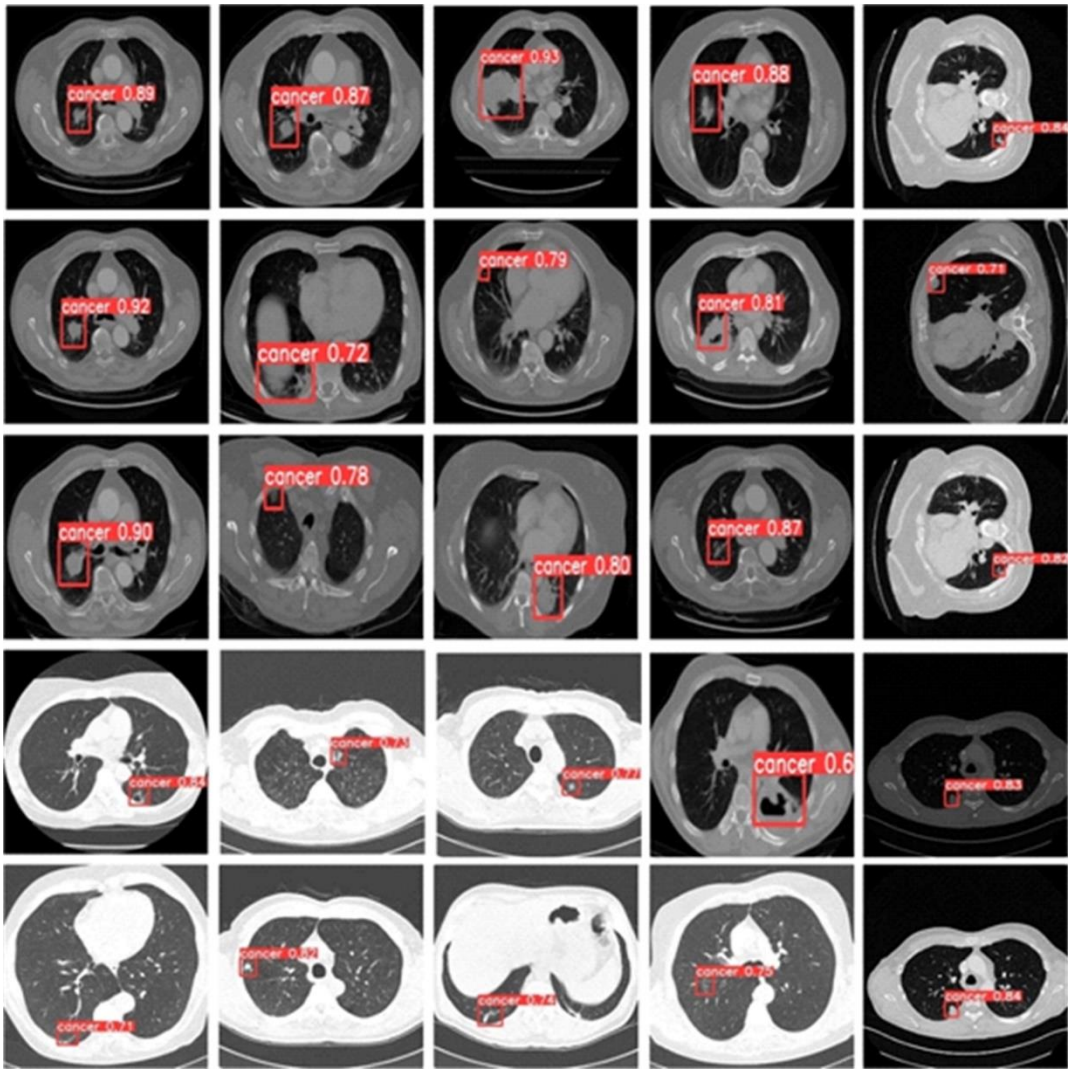


Fig. 6.17 Prediction results on cancerous datasets applied to Proposed Model-2

Table 6.3 Performance on cancerous datasets applied to Proposed Model-2

Prediction	Counts	Result
Cancer	46	0.92
Non-Cancer	2	0.04
Cancer/Non-Cancer (Both)	0	0.0
Not Detected	2	0.04

Table 6.4 Predictions on Non-Cancer test datasets applied to Proposed Model-2

Patient ID	True Label	Prediction Label
102371	Non-Cancer	Non-Cancer
102376	Non-Cancer	Non-Cancer
102394	Non-Cancer	Non-Cancer
102705	Non-Cancer	Non-Cancer
102713	Non-Cancer	Non-Cancer
102753	Non-Cancer	Non-Cancer
102755	Non-Cancer	Non-Cancer
102759	Non-Cancer	Non-Cancer
102773	Non-Cancer	Non-Cancer
102777	Non-Cancer	Non-Cancer
102803	Non-Cancer	Non-Cancer
102804	Non-Cancer	Non-Cancer
102809	Non-Cancer	Non-Cancer
102811	Non-Cancer	Non-Cancer
102815	Non-Cancer	Non-Cancer
102818	Non-Cancer	Non-Cancer
102820	Non-Cancer	Non-Cancer
102822	Non-Cancer	Non-Cancer/ Cancer (Both)
102824	Non-Cancer	Non-Cancer
102835	Non-Cancer	Non-Cancer
102837	Non-Cancer	Non-Cancer
102850	Non-Cancer	Non-Cancer
102853	Non-Cancer	Non-Cancer
102854	Non-Cancer	Non-Cancer
102867	Non-Cancer	Non-Cancer
102871	Non-Cancer	Non-Cancer
102872	Non-Cancer	Non-Cancer
102875	Non-Cancer	Non-Cancer
102877	Non-Cancer	Non-Cancer
102878	Non-Cancer	Non-Cancer/ Cancer (Both)
102888	Non-Cancer	Not Detected
102895	Non-Cancer	Non-Cancer
102908	Non-Cancer	Non-Cancer
102910	Non-Cancer	Non-Cancer
102913	Non-Cancer	Non-Cancer
102917	Non-Cancer	Non-Cancer
102918	Non-Cancer	Non-Cancer

102928	Non-Cancer	Non-Cancer
102939	Non-Cancer	Not Detected
102948	Non-Cancer	Cancer
103229	Non-Cancer	Non-Cancer
201382	Non-Cancer	Non-Cancer
201385	Non-Cancer	Non-Cancer
201411	Non-Cancer	Non-Cancer
201574	Non-Cancer	Non-Cancer
201582	Non-Cancer	Non-Cancer
201587	Non-Cancer	Non-Cancer
201632	Non-Cancer	Non-Cancer
207408	Non-Cancer	Non-Cancer
211174	Non-Cancer	Non-Cancer

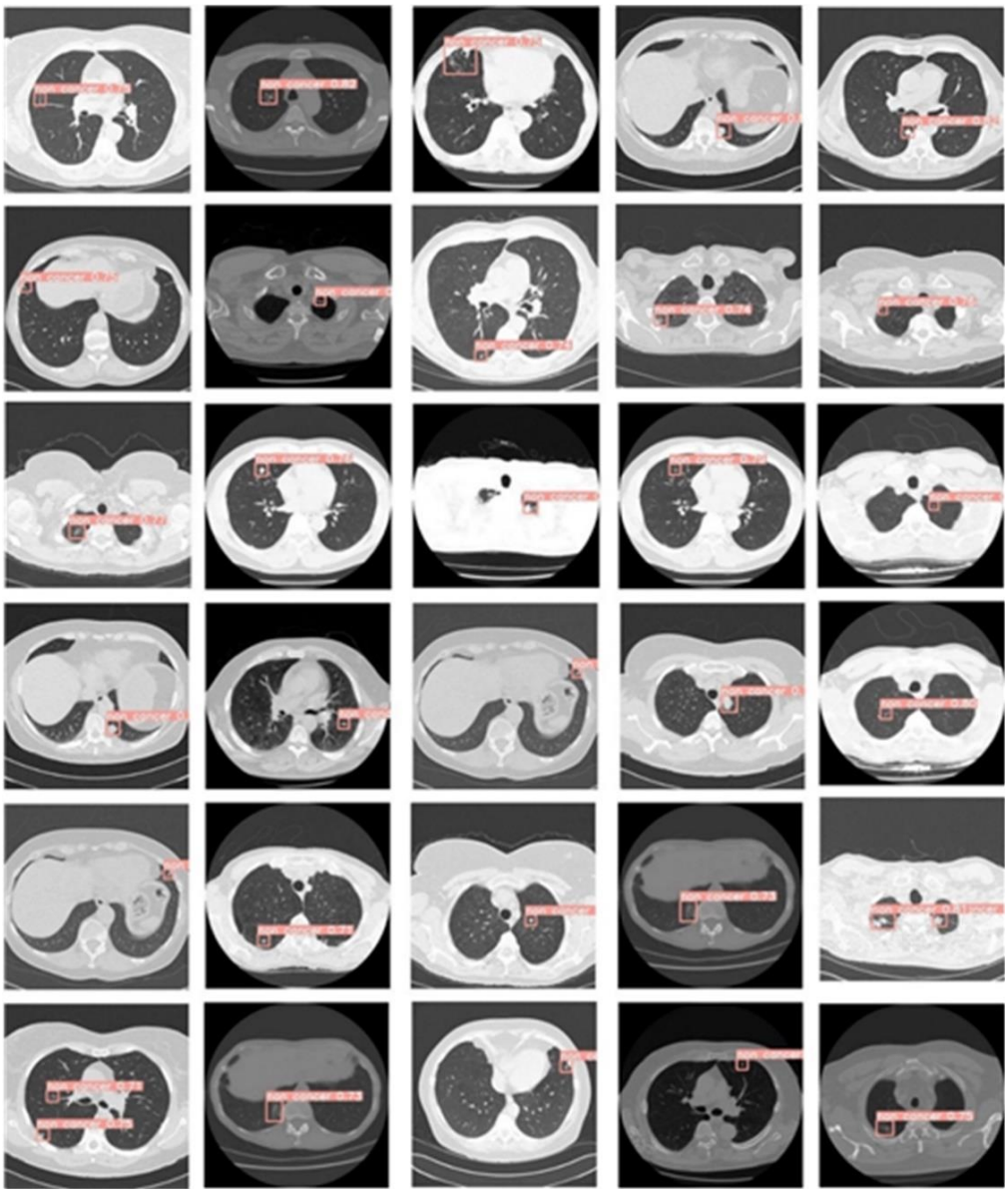


Fig. 6.18 Prediction results on non-cancerous datasets applied to Proposed Model-2

Table 6.5 Performance on non-cancerous datasets applied to Proposed Model-2

Prediction	Counts	Result
Non-Cancer	45	0.90
Non-Cancer/Cancer (Both)	2	0.04
Cancer	1	0.02
Not Detected	2	0.04

6.5 Comparison of Performance of YOLOv5, Proposed Models and Previous Work

Table 6.6 shows the comparison of the object detectors used for general objects as well as lung cancer detection. The previous work considered for comparison consists of various sources of datasets used to train different types of object detectors. The evaluation metrics used for comparison are Precision, Recall, mAP and F1 score. From the Table 6.6 it is depicted that the best performing models have precision of 91.2%, recall of 95%, mAP of 86.3% and F1 score of 0.89. The most effective Proposed Model-2 implemented in this work shows the Precision: 95%, Recall: 90%, mAP: 95% and F1Score: 0.92. The previous work used for comparison have single source of datasets which can lead to non-generalizability of the model. While the lung cancer detection models used in this work were trained on 15 different lung CT datasets. This large variations in datasets led the models to learn effectively and thus a better performance could be achieved.

Table 6.6 Comparison of Performance of proposed nodule detection model with the existing work

Title	Dataset	Model	Performance
Redmon *et al.* [199]	PASCAL VOC	YOLO	mAP: 57.9%
Redmon *et al.* [203]	PASCAL VOC and COCO	YOLOv2	mAP: 76.8%
Ren *et al.* [180]	PASCAL VOC	Faster R- CNN	mAP: 73.2%
Li *et al.* [209]	PASCAL VOC	YOLOv3	mAP: 86.3%
Zakria *et al.* [216]	Dota	YOLOv4	mAP: 75.15%
Liao *et al.* [193]	LUNA 16 and DSB2017	3-D region proposal network, leaky noisy OR Gate, backbone network, a modified U-net.	Recall: 85% Accuracy: 73.73%
Xie *et al.* [187]	LUNA16	Faster R- CNN	Recall: 86.42%
Peng *et al.* [107]	LUNA16	3D multi-scale deep convolution neural networks	Recall: 92.3%
Ding *et al.* [186]	LUNA16	Faster R-CNN	F1 Score: 0.89
Zhang *et al.* [183]	LUNA16 and TianChi competition datasets	Res2Net structure, the 3D multi-scale attention block, U-Net-like	Recall: 0. 842

		encoder-decoder, Faster R-CNN, 3D deep convolutional neural	
Wang *et al*. [184]	Tianchi AI Dataset, LUNA-16 Dataset	Faster-RCNN	Recall: 75.6%
Gu *et al*. [185]	LIDC-IDRI	Deformable convolution is proposed	mAP: 82.7%.
Su *et al*. [189]	LIDC-IDRI	Faster R-CNN	Average Precision: 91.2%.
Liu *et al*. [208]	LUNA16, SPIE-AAPM, LungTIME and HMS	3D Feature Pyramid Network (3DFPN)	Recall: 90.6%
Ardila *et al*. [190]	NLST	3D Inception-V1	Recall: 95%
Yan *et al*. [175]	DeepLesion	Faster R-CNN	Precision: 80%
Yan *et al*. [176]	DeepLesion	3D-CNN	Recall: 84.37%
Li *et al*. [177]	T2-weightedMRscans from the First Affiliated Hospital of Guangzhou Medical University	Faster-RCNN	Recall: 85.2%
Huang *et al*. [17]	NLST	A deep learning algorithm (referred to as DeepLR)	Recall: 88%
Aslani *et al*. [18]	NLST	Time-series deep learning model (DeepCAD-NLM-L)	Recall:83%
Bhatt *et al.* [77]	NLST	YOLOv4	Precision: 95%, Recall: 81%, mAP: 89.1%
YOLOv5	LIDC-IDRI, LUNA16, NSCLC-Radiogenomics, NSCLC-Radiomics, LCTSC, QIN, TCGA-LUSC, TCGA-LUAD, 4D-Lung, LungCT-Diagnosis, RIDER, DSB2017, NLST, ECLAP, Anti-PD 1	YOLOv5	Precision: 90%, Recall: 80%, mAP: 85%, F1 Score: 0.85
Proposed Model-1	LIDC-IDRI, LUNA16, NSCLC-Radiogenomics, NSCLC-Radiomics, LCTSC, QIN, TCGA-LUSC, TCGA-LUAD, 4D-Lung, LungCT-Diagnosis, RIDER, DSB2017, NLST, ECLAP, Anti-PD 1	Modified YOLOv5 using Focus and CSP layers	Precision: 85%, Recall: 82%, mAP: 85%, F1 Score: 0.84
Proposed Model-2	LIDC-IDRI, LUNA16, NSCLC-Radiogenomics, NSCLC-Radiomics, LCTSC, QIN, TCGA-LUSC, TCGA-LUAD, 4D-Lung, LungCT-Diagnosis, RIDER, DSB2017, NLST, ECLAP, Anti-PD 1	Modified YOLOv5 using DWSC layer	Precision: 95%, Recall: 90%, mAP: 95%, F1 Score: 0.92

6.6 Graphical User Interface for Lung Nodule Diagnosis

There were five nodule detectors trained in previous chapter based on deep learning approaches. In order to increase the performance of YOLOv5 on the objective of nodule detection, two models based on YOLOv5 network alterations are presented in this chapter. After comparative analysis, it was found that the Proposed Model-2 is the best model for nodule diagnosis. In order to make the nodule diagnosis user friendly, a Lung Cancer Detection GUI is implemented in this study. To improve nodule detection results, users may simply alter the threshold based on their own requirements. In comparison to conventional approaches such as manual monitoring, the technology used in this study will assist the radiologist in detecting nodules on CT images.

Identifying Graphical User Interface (GUI) components in GUI images is a domain-specific object detection task. It supports numerous software engineering tasks, including GUI animation and testing. Studies that have already been conducted on the recognition of GUI elements directly adopt sophisticated techniques from the computer vision field, including both deep learning models and older ones that depend on conventional image processing features. Taking knowledge from previous work, a new GUI is designed specifically for the nodule detectors implemented in this study for the diagnosis of lung cancer.

Fig. 6.19 depicts the GUI training and validation procedures for nodule detectors. In the first step, the Lung cancer detection GUI, seen in Fig. 6.20, is launched. Consequently, the nodule detector built in Chapters 5 and 6 are chosen. A new project is formed by specifying its name, which generates a directory with sub-directories for storing the required data and configuration files along with the hyperparameters settings in accordance with the implemented nodule detector in this work. The detector is then trained and the checkpoints will be saved for use in generating the inference graph. By configuring the IoU threshold and number of classes, the trained model may be applied to test images for nodule detection.

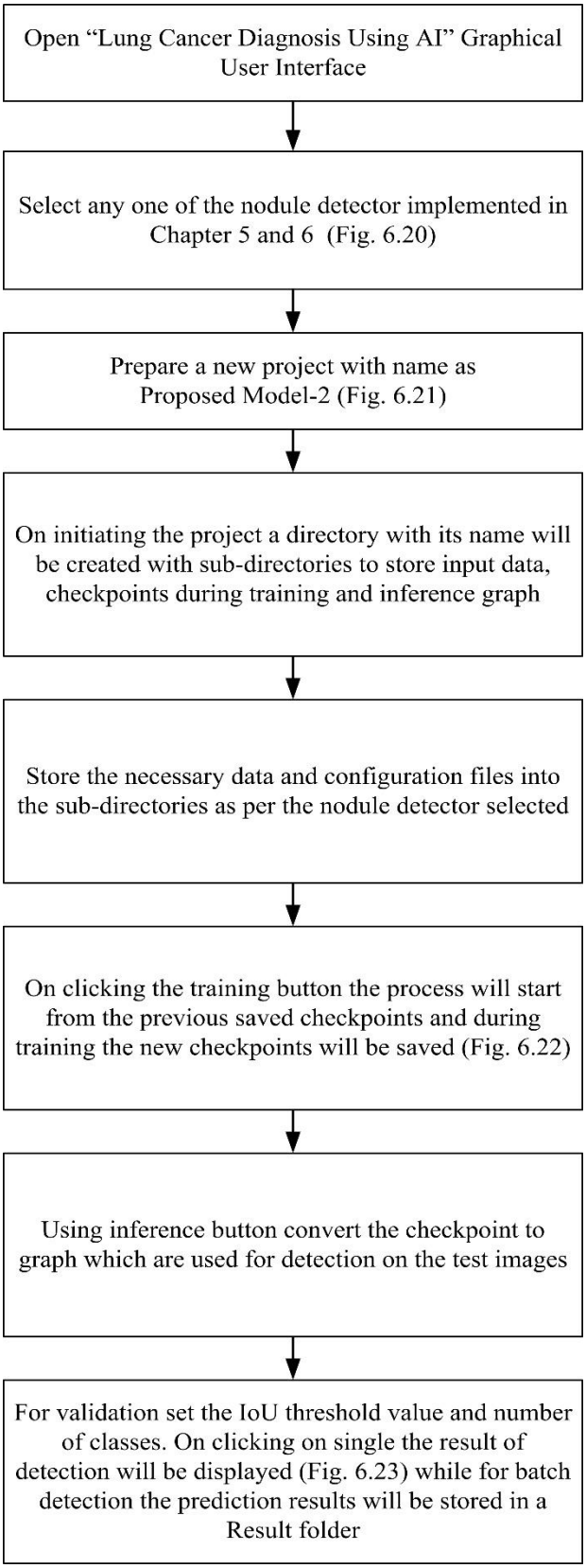
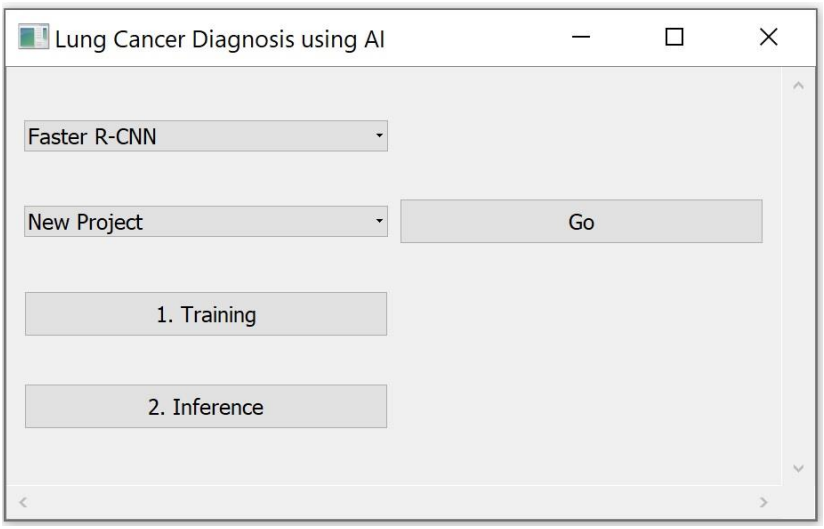
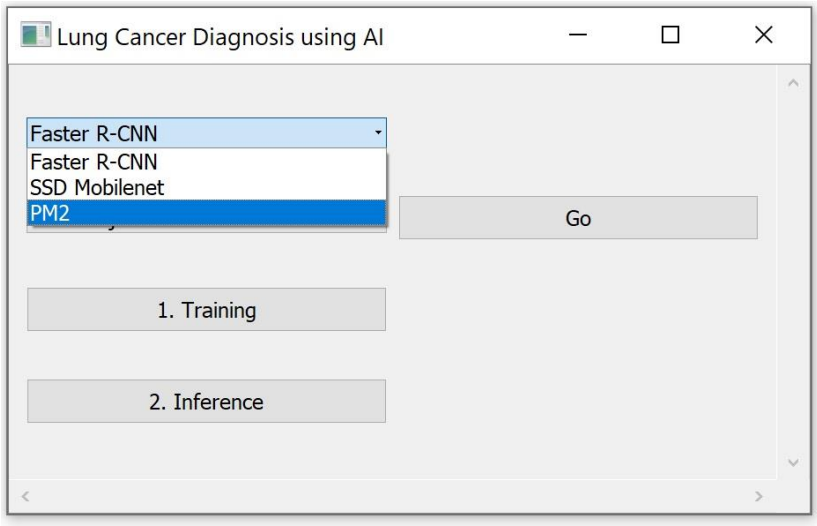


Fig. 6.19 Steps to use the GUI for Nodule Detection



(a)



(b)

Fig. 6.20 (a) Nodule Detection GUI (b) Selection of nodule detector

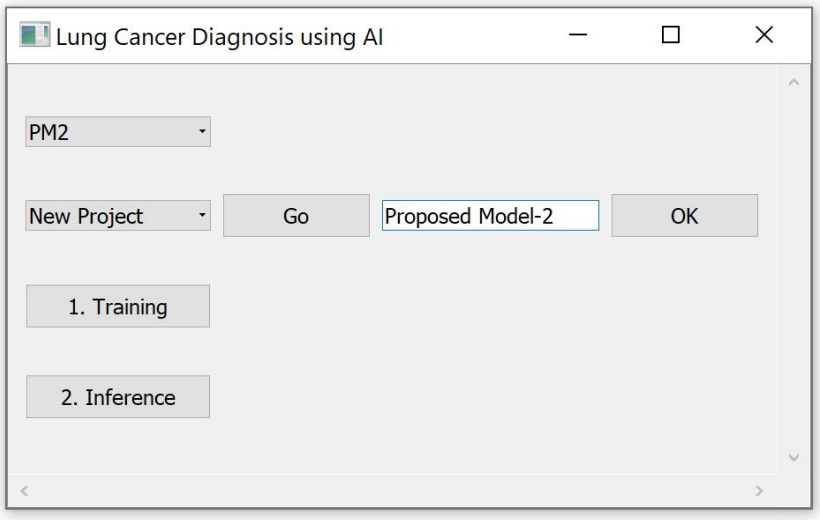


Fig. 6.21 Selection of nodule detector and project

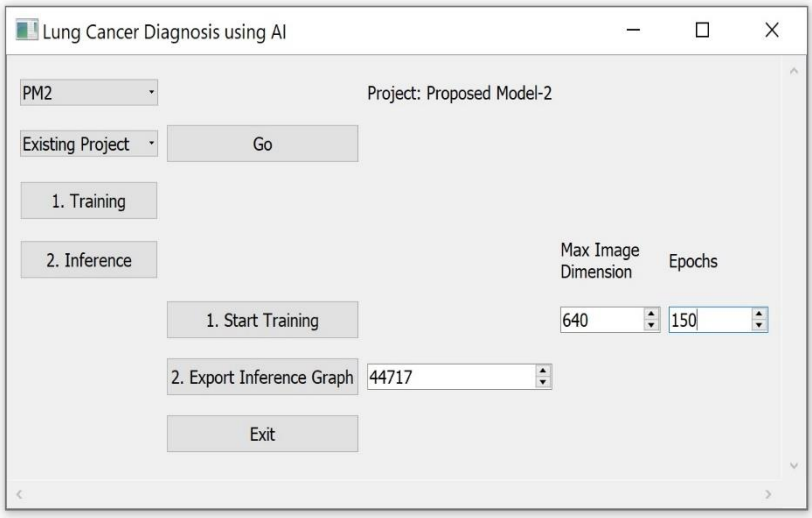


Fig. 6.22 Train nodule detector

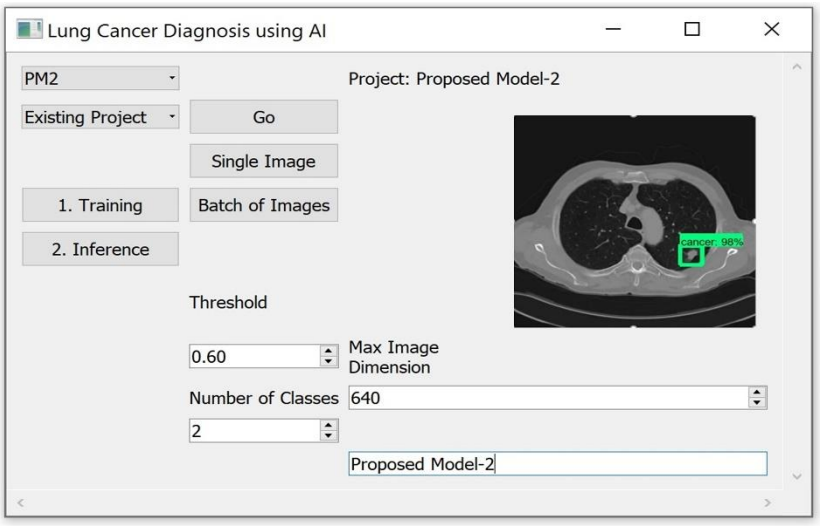


Fig. 6.23 Prediction on test data

6.7 Discussion

In previous chapter, nodule identification was initially performed using Faster R-CNN, a two-stage anchor-box-based deep learning technique, as well as the one-stage SSD. However, both model's performance and detection results were poor. Following that, three YOLO models were trained: YOLOv3, YOLOv4 and YOLOv5. The comparison in the preceding chapter demonstrated that YOLOv5 outperformed all other nodule detectors. Because of a significant number of GFLOPs, the YOLOv5 had a high computation time. Thus, this chapter describes two automated nodule identification models that are modifications of YOLOv5, resulting in improved performance and efficient prediction on test images. In addition, the suggested model required much less training time than the other implemented nodule detectors. Finally, the highest performing Proposed Model-2 was chosen for validation. In addition, to make the nodule identification framework more user-friendly, a GUI is built that can be used to train nodule detectors and utilized for lung cancer diagnosis.

CHAPTER-7
Conclusion and Future Scope

7.1 Conclusion

It is known from previous research that detecting lung cancer in its early stages using LDCT scans may increase survival rates and save medical costs. Existing LDCT screening produce vast amounts of screening data, but they face difficulties such as high over-diagnosis rates, high costs and increasing radiation exposure. These difficulties emphasize the necessity for clinicians to be assisted with an automated lung cancer diagnostic system in order to eliminate false positive results and make the screening more cost-effective. It underlines the need of providing individualized screening recommendations for people with different risk factors in order to save money and decrease radiation exposure while retaining high cancer detection efficiency. The precise and accurate detection of pulmonary nodules is one of the most effective methods for reducing lung cancer-related mortality. As a result, enhanced approaches to lower the false positive rate must be implemented. Thus, in this work methodologies are implemented to help clinicians for the lung cancer diagnosis.

Initial implementations of lung cancer classification algorithms in this study were based on CBIR, hybrid approach and DNNs. The outcomes of these models were presented in Chapter 3 and 4. The CBIR approach was presented in this work to classify CT slices as cancer and non-cancer based on a machine learning framework consisting of SURF, k-means and SVM algorithms. This approach offered accuracy of 98.56%. Another approach based on the hybrid model was designed for lung nodule classification. This model is a combination of deep learning and machine learning approach. The developed model consisted of ResNet-50 and SVM to differentiate cancer and non-cancer. The accuracy achieved in this method was 96.73%. It was discovered that the strategies resulted in improved performance. However, since these models were trained on just one data set i.e., LUNA16, the approaches may not be generalized adequately to other CT scan datasets generated by other radiological scan settings. Moreover, these techniques required separate procedures for feature extraction and

classification. So, in order to implement an end-to-end model involving a single step, seven DNNs namely MobileNet, SqueezeNet, AlexNet, GoogLeNet, ResNet-18, VGG-16 and VGG-19 were implemented for lung cancer classification in chapter 4. Finally, based on the challenges and drawbacks of these networks, there was a novel model implemented which incorporates the domain knowledge into the design of deep learning framework to enable model interpretation and improve lung cancer diagnosis. This model was named as SVGG as it was derived from concept of VGG. These DNN models were trained on the LUNA16 and DSB2017 datasets and they demonstrated excellent robustness for lung cancer classification. The DNNs can be used to classify lung nodules, but it can also be used as an effective feature extractor. Among these DNNs, the SVGG showed the best performance for the lung cancer classification with accuracy of 99.60%.

The aforementioned three methodologies were implemented for lung cancer classification. While the automated diagnosis of lung cancer needed a detection method that determines the precise location and likelihood of the abnormality being cancerous or non-cancerous. The main objective of this work was to develop an automated lung cancer diagnosis system. So, to achieve this, there were five nodule detectors namely Faster R-CNN, SSD, YOLOv3, YOLOv4 and YOLOv5 are trained and the performance of these networks were shown in chapter 5. Initially, Faster R-CNN and SSD were trained for nodule detection and their results were compared. The results showed that the Faster R-CNN achieved the highest accuracy of 95.80% while SSD had an accuracy of 43.80%. Though the models were complex and took long time to be trained but performance was low. So, YOLO algorithms namely YOLOv3, YOLOv4 and YOLOv5 were trained. The comparison of YOLOv3, YOLOv4 and YOLOv5 proved that YOLOv5 outperformed YOLOv3 and YOLOv4 in terms of performance. The Precision of 90%, Recall of 80%, mAP of 85% and F1 Score of 85% was achieved in YOLOv5.

The mentioned five nodule detectors were complicated, had high computation time and low performance during validation. So, to address these issues, two automated lung cancer diagnosis models with improved performance and reduced computational complexity have been developed. The first proposed approach was a YOLOv5 modification that incorporated the Focus and CSP layers. This model obtained 85% of precision, 82% of recall, 85% of mAP and F1 score of 0.83. To lower the computation time of the Proposed Model-1, the second model was presented, which was a refinement of the first proposed model that replaced convolution

layers in the backbone with DWSC. The Proposed Model-2 outperformed the first with a precision of 95%, recall of 90%, mAP of 95% and F1 score of 0.92.

To address the challenge of a lack of well-labelled medical datasets, 14 separate CT scan datasets were employed to train the seven nodule detection algorithms listed above. The performance of Proposed Model-2 outperformed all other trained models. To validate Proposed Model-2, 100 NLST CT scans with an equal number of cancer and non-cancer were employed. According to the actual label, the prediction accuracy on the test dataset was 90% for non-cancer and 92% for cancer. This was possible because the Proposed Model-2 was robust for feature diagnosis, especially for difficult-to-detect nodules in CT scan images.

Finally, a graphical user interface (GUI) for lung cancer detection was also developed to make the nodule identification framework more user-friendly.

7.2 Future Work

This study indicates the effectiveness of using a novel nodule identification paradigm based on unified deep classification and detection. This study reports on very promising nodule classification and detection results. This work was accomplished utilizing CT scans for lung cancer diagnosis, but in the future, other types of finer-grained semantic information, such as fusing more context information about the nodules, such as connections with surrounding blood vessels and information about the patient, such as the medical history report, could be integrated to build a better nodule knowledge base for automated diagnosis systems.

The accurate nodule volume measurement, as well as the extraction and integration of the nodule diagnostic for the development of radiological text reports, may all be included in future work.

The proposed nodule detector may assist radiologists in locating all types of nodules using a single unified framework. It might be used as an initial screening tool, with detection findings sent to additional specialized systems trained on certain kinds of nodules. This study may also be used to identify different sorts of medical images in the future to forecast the correct disease for effective diagnosis. Furthermore, future research might entail analyzing it on additional clinical data and promoting it in clinical practice with the help of radiologists and surgeons. This work may be used to a variety of detecting problems in the future.

References

[1] https://bio.libretexts.org [Online]

[2] https://www.lungevity.org [Online]

[3] F. Bray et al., "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries", *CA: A Cancer Journal Clinicians.*, vol. 68, pp. 394-424, 2018. doi: 10.3322/caac.21492.

[4] AS Tsao et al., "Scientific advances in lung cancer 2015", *Journal Thoracic Oncology*, vol. 11, no. 5, pp. 613-638, 2016. doi: 10.1016/j.jtho.2016.03.012.

[5] https://www.cancer.org/cancer/non-small-celllung-cancer/about/key-statistics.html. [Online].

[6] SG Spiro and JC Porter, "Lung cancer-where are we today? Current advances in staging and nonsurgical treatment", *American Journal Respiratory and Critical Care Medicine*, vol. 166, no. 9, pp. 1166-1196, 2002. doi: 10.1164/rccm.200202-070SO.

[7] PM Ellis and R. Vandermeer, "Delays in the diagnosis of lung cancer", *Journal Thoracic Disease*, vol. 3, no. 3, pp. 183-188, 2011. doi: 10.3978/j.issn.2072-1439.2011.01.01.

[8] R.L. Siegel, K.D. Miller and A. Jemal, "Cancer statistics, 2020", *CA: A Cancer Journal Clinicians*, vol. 70, no. 1, pp. 7-30, 2020. doi: 10.3322/caac.21590.

[9] R.L. Siegel et al., "Cancer statistics, 2021", *CA: A Cancer Journal Clinicians*, vol. 71, no. 1, pp. 7-33, 2021. doi: 10.3322/caac.21654.

[10] R.L. Siegel et al., "Cancer statistics, 2022", *CA: A Cancer Journal Clinicians*, vol. 72, no. 1, pp. 7-33, 2022. doi: 10.3322/caac.21708.

[11] Cancer Tomorrow, International agency for research on cancer, *World Health Organization, GLOBOCAN 2018*. [Online].

[12] JK Leader et al., "Pulmonary nodule detection with low-dose CT of the lung: Agreement among radiologists", *American Journal Roentgenology*, vol. 185, no. 4, pp. 973-978, 2005. doi: 10.2214/AJR.04.1225.

[13] A. Panunzio and P. Sartori, "Lung cancer and radiological imaging", *Current Radiopharmaceuticals*, vol. 13, no. 3, pp. 238-242, 2020. doi: 10.2174/1874471013666200523161849.

[14] SE Verleden et al., "Small airways pathology in idiopathic pulmonary fibrosis: a retrospective cohort study", *The Lancet Respiratory Medicine*, vol. 8, no. 6, pp. 573-584, 2020. doi: 10.1016/S2213-2600(19)30356-X.

[15] H. Rusinek et al., "Pulmonary nodule detection: Low-dose versus conventional CT", *Radiology*, vol. 209, no. 1, pp. 243-249, 1998. doi: 10.1148/radiology.209.1.9769838.

[16] SB Lo et al., "Computer-aided detection of lung nodules on CT with a computerized pulmonary vessel suppressed function", *American Journal Roentgenology*, vol. 210, pp. 480-488, 2018. doi: 10.2214/AJR.17.18718.

[17] P. Huang et al., "Prediction of lung cancer risk at follow-up screening with low-dose CT: a training and

validation study of a deep learning method", *Lancet Digit Health*, vol. 1, no. 7, pp. 353-362, 2019. doi: 10.1016/S2589-7500(19)30159-1.

[18] S. Aslani et al., "Enhancing cancer prediction in challenging screen-detected incident lung nodules using time-series deep learning", *Computer Vision and Pattern Recognition (CVPR), arXiv:2203.16606*.

[19] H. Fujitaa et al., "Computer-aided diagnosis: the emerging of three CAD systems induced by Japanese health care needs", *Computer Methods and Programs in Biomedicine*, vol. 92, no. 3, pp. 238-248, 2008. doi: 10.1016/j.cmpb.2008.04.003.

[20] K. Doi, "Computer-aided diagnosis in medical imaging: historical review, current status and future potential", *Computerized Medical Imaging and Graphics*, vol. 31, no. 4-5, pp. 198-211, 2007. doi: 10.1016/j.compmedimag.2007.02.002.

[21] N. Lee et al., "Potential of computer-aided diagnosis to improve CT lung cancer screening", In *Proc. IEEE Reviews in Biomedical Engineering*, vol. 2, pp. 136-146, 2009. doi: 10.1109/RBME.2009.2034022.

[22] P.Y.M. Anshad and S.S. Kumar, "Recent methods for the detection of tumor using computer aided diagnosis-A review", In *Proc. IEEE International Conference on Control, Instrumentation, Communication and Computational Technologies*, pp. 1014-1019, 2014. doi: 10.1109/ICCICCT.2014.6993108.

[23] B. van Ginneken, C.M.S. Prokop and M. Prokop, "Computer-aided diagnosis: how to move from the laboratory to the clinic", *Radiology*, vol. 261, no. 3, pp. 719-732, 2011. doi: 10.1148/radiol.11091710.

[24] M. Firmino et al., "Computer-aided detection system for lung cancer in computed tomography scans: review and future prospects", *BioMedical Engineering*, 2014. doi: 10.1186/1475-925X13-41.

[25] A. El-Baz et al., "Review article computer-aided diagnosis systems for lung cancer: challenges and methodologies", *International Journal of Biomedical Imaging*, pp. 1-46, 2013. doi: 10.1155/2013/942353.

[26] S.G. Armato et al., "The lung image database consortium (LIDC) and image database resource initiative (IDRI): A completed reference database of lung nodules on CT scans", *Med. Phys.*, vol. 38, no. 2, pp. 915-931, 2011. doi: 10.1118/1.3528204.

[27] A.A.A. Setio et al., "Automatic detection of large pulmonary solid nodules in thoracic CT images", *Med. Phys.*, vol. 42, no. 10, pp. 5642–5653, 2015. doi: 10.1118/1.4929562.

[28] K. Murphy et al., "A large-scale evaluation of automatic pulmonary nodule detection in chest CT using local image features and k-nearest-neighbour classification", *Medical Image Analysis*, vol. 13, no. 5, pp. 757–770, 2009. doi: 10.1016/j.media.2009.07.001.

[29] A.A.A. Setio et al., "Validation, comparison and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge", *Medical Image Analysis*, vol. 42, pp. 1-13, 2017. doi: 10.1016/j.media.2017.06.015.

[30] K. Kuan et al., "Deep learning for lung cancer detection: Tackling the Kaggle data science bowl 2017 challenge", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9, 2017. arxiv.org/abs/1705.09435.

[31] National Lung Screening Trial Research Team, "Data from the National Lung Screening Trial (NLST) [Data set]", *The Cancer Imaging Archive*, 2013. doi: 10.7937/TCIA.HMQ8-J677.

[32] S. Bakr et al., "A radiogenomic dataset of non-small cell lung cancer", *Sci Data*, vol. 5, pp. 180-202. 2018. doi: 10.1038/sdata.2018.202.

[33] I. Fornacon-Wood et al., "Radiomics as a personalized medicine tool in lung cancer: Separating the hope from the hype", *Lung Cancer*, vol. 146, pp. 197-208, 2020. doi: 10.1016/j.lungcan.2020.05.028.

[34] J. Yang et al., “Data from lung CT segmentation challenge”, *The Cancer Imaging Archive*, 2017. doi: 10.7937/K9/TCIA.2017.3R3FVZ08.

[35] LP Clarke et al., “Quantitative imaging for evaluation of response to cancer therapy”, *Translational Oncology*, vol. 2, no. 4, pp. 195–197, 2009. doi: 10.1593/tlo.09217.

[36] S. Kirk et al., “The cancer genome atlas lung squamous cell carcinoma collection (TCGA-LUSC) (Version 4) [Data set]”, *The Cancer Imaging Archive*, 2016. doi: 10.7937/K9/TCIA.2016.TYGKKFMQ.

[37] B. Albertina et al., “Radiology data from the cancer genome atlas lung adenocarcinoma [TCGA-LUAD] collection”, *The Cancer Imaging Archive*, 2016. doi: 10.7937/K9/TCIA.2016.JGNIHEP5.

[38] G.D. Hugo et al., “A longitudinal four-dimensional computed tomography and cone beam computed tomography dataset for image-guided radiation therapy research in lung cancer”, *Med. Phys.*, vol. 44, pp. 762–771, 2017. doi: 10.1002/mp.12059.

[39] R.V. Adiraju and S. Elias, “A survey on lung CT datasets and research trends”, *Research on Biomedical Engineering*, vol. 37, pp. 403–418, 2021. doi: 10.1007/s42600-021-00138-3.

[40] S. Armato et al., “The reference image database to evaluate response to therapy in lung cancer (RIDER) project: a resource for the development of change-analysis software”, *Clinical Pharmacology & Therapeutics*, vol. 84, no. 4, pp. 448-456, 2008. doi: 10.1038/clpt.2008.161.

[41] ELCAP Dataset. Accessed: 2019. http://www.via.cornell.edu/databases/lungdb.html. [Online].

[42] P. Madhavi, S. Patel and A.S. Tsao, “Data from anti-PD-1 immunotherapy lung [Data set]”, *The Cancer Imaging Archive*, 2019. doi: 10.7937/tcia.2019.zjjwb9ip.

[43] F. Kanavati et al., “Weakly-supervised learning for lung carcinoma classification using deep learning”, *Scientific Report*, vol. 10, no. 9297, 2020. doi: 10.1038/s41598-020-66333-x.

[44] SG Armato et al., “Lung cancer: performance of automated lung nodule detection applied to cancers missed in a CT screening program”, *Radiology*, vol. 225, no. 3, pp. 685-692, 2002. doi: 10.1148/radiol.2253011376.

[45] J.R.F. da Silva Sousa et al., “Methodology for automatic detection of lung nodules in computerized tomography images”, *Computer Methods and Programs in Biomedicine*, vol. 98, no. 1, pp. 1–14, 2010. doi: 10.1016/j.cmpb.2009.07.006.

[46] D. Cascio et al., “Automatic detection of lung nodules in CT datasets based on stable 3D mass-spring models”, *Computers in Biology and Medicine*, vol. 42, no. 11, pp. 1098–1109, 2012. doi: 10.1016/j.compbiomed.2012.09.002.

[47] T. Messay, R.C. Hardie and S.K. Rogers, “A new computationally efficient CAD system for pulmonary nodule detection in CT imagery”, *Medical Image Analysis*, vol. 14, no. 3, pp. 390–406, 2010. doi: 10.1016/j.media.2010.02.004.

[48] G. Wei et al., “A content-based image retrieval scheme for identifying lung nodule malignancy levels”, In *29th Chinese Control and Decision Conference (CCDC)*, pp. 3127-3130, 2017. doi: 10.1109/CCDC.2017.7979045.

[49] N. Malviya, N. Choudhary and K. Jain, “Content based medical image retrieval and clustering based segmentation to diagnose lung cancer”, *Advances in Computational Sciences and Technology*, vol. 10, no. 6, pp. 1577-1594, 2017.

[50] Y. Yoon, T. Hwang and H. Lee, “Prediction of radiographic abnormalities by the use of bag-of-features and

convolutional neural networks", *The Veterinary Journal*, vol. 237, pp. 43-48, 2018. doi: 10.1016/j.tvjl.2018.05.009.

[51] M. Bedo et al., "Endowing a content-based medical image retrieval system with perceptual similarity using ensemble strategy", *Journal of Digital Imaging*, vol. 29, no. 1, pp. 22–37, 2016. doi: 10.1007/s10278-015-9809-1.

[52] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos", In *9th IEEE International Proc. on Computer Vision*, vol. 2, pp. 1470-1477, 2003. doi: 10.1109/ICCV.2003.1238663.

[53] J. Philbin et al., "Object retrieval with large vocabularies and fast spatial matching", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-8, 2007. doi: 10.1109/CVPR.2007.383172.

[54] H. Lodhi et al., "Text classification using string kernels", *Journal of Machine Learning Research*, vol. 2, pp. 419–444, 2002.

[55] J. Yang et al., "Evaluating bag-of-visual words representations in scene classification", In *Proc. International Workshop on Multimedia Information Retrieval*, pp. 197–206, 2007. doi: 10.1145/1290082.1290111.

[56] E. Nowak, F. Jurie and B. Triggs, "Sampling strategies for bag-of-features image classification", In A. Leonardis, H. Bischof, A. Pinz (eds) *Computer vision – ECCV of Lecture Notes in Computer Science*, vol. 3954, pp. 490–503, 2006. doi: 10.1007/11744085_38.

[57] G. Csurka et al., "Visual categorization with bags of keypoints", *Workshop on Statistical Learning in Computer Vision, ECCV*, pp. 1-16, 2004.

[58] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories", In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 524-531, 2005. doi: 10.1109/CVPR.2005.16.

[59] M. Gangeh et al., "A texton-based approach for the classification of lung parenchyma in CT images", In *T. Jiang, N. Navab, J.P.W. Pluim, M.A. Viergever (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010, Lecture Notes in Computer Science*, vol. 6363, pp. 595-602, 2010. doi: 10.1007/978-3-642-15711-0_74.

[60] U. Avni et al., "X-ray image categorization and retrieval using patch-based visual words representation", In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 350–353, 2009. doi: 10.1109/TMI.2010.2095026.

[61] J. Caicedo, A. Cruz and F. Gonzalez, "Histopathology image classification using bag of features and kernel functions", In *C. Combi, Y. Shahar, A. Abu-Hanna (eds.) AIME 2009. LNCS,* pp. 126–135, 2009. doi: 10.1007/978-3-642-02976-9_17.

[62] S.H. Raza et al., "Automated classification of renal cell carcinoma subtypes using bag-of-features", In *IEEE International Conference of Engineering in Medicine and Biology Society (EMBC)*, pp. 6749–6752, 2010. doi: 10.1109/IEMBS.2010.5626009.

[63] S.D. Bhatt and H.B. Soni, "Image retrieval using bag-of-features for lung cancer classification", In *6th IEEE International Conference on Inventive Computation Technologies (ICICT)*, pp. 531-536, 2021. doi: 10.1109/ICICT50816.2021.9358499.

[64] T. Lindeberg, "Feature detection with automatic scale selection", *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998. doi: 10.1023/A:1008045108935.

[65] D. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. doi: 10.1023/B:VISI.0000029664.99615.94.

[66] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005. doi: 10.1109/TPAMI.2005.188.

[67] Z. Songtao, Li Chao and Li Liqing, "An improved method for eliminating false matches", In *2nd International Conference on Image, Vision and Computing*, pp. 133-137, 2017. doi: 10.1109/ICIVC.2017.7984533.

[68] L. Kabbai et al., "Image matching based on LBP and SIFT descriptor", In *12th IEEE International Multi-Conference on Systems, Signals & Devices (SSD15)*, pp. 1-6, 2015. doi: 10.1109/SSD.2015.7348116.

[69] A. Krishna, P.C.S. Rao and C.Z. Basha, "Efficient computerized lung cancer detection using bag of words", In *7th International Conference on Smart Structures and Systems (ICSSS)*, pp. 1-4, 2020. doi: 10.1109/ICSSS49621.2020.9202039.

[70] H. Bay, T. Tuytelaars and L.V. Gool, "Speeded-up robust features (SURF)", *Computer Vision and Image Understanding*, vol. 110, pp. 346–359, 2008. doi: 10.1016/j.cviu.2007.09.014.

[71] P. Nanglia et al., "Comparative investigation of different feature extraction techniques for lung cancer detection system", In: A. Luhach, D. Singh, PA Hsiung, K. Hawari, P. Lingras, P. Singh (eds) *Advanced Informatics for Computing Research. ICAICR*, vol. 955, pp. 296-307, 2019. doi: 10.1007/978-981-13-3140-4_27.

[72] P. Sangamithraa and S. Govindaraju, "Lung tumour detection and classification using EK-Mean clustering", In *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 2201-2206, 2016. doi: 10.1109/WiSPNET.2016.7566533.

[73] V. Kananm and V. Jagan, "Detection of lung cancer using image segmentation", *International Journal of Electrical Engineering & Technology*, vol. 11, no. 2, pp. 7-16, 2020.

[74] V. Suma, "A novel Information retrieval system for distributed cloud using hybrid deep fuzzy hashing algorithm", *Journal of Information Technology and Digital World*, vol. 2, no. 3, pp. 151-160, 2020. doi: 10.36548/jitdw.2020.3.003

[75] S. Potghan, R. Rajamenakshi and A. Bhise, "Multi-layer perceptron based lung tumor classification", In *2nd IEEE International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 499-502, 2018. doi: 10.1109/ICECA.2018.8474864.

[76] C. Fu and D. Cai, "EFANNA: An extremely fast approximate nearest neighbor search algorithm based on KNN graph", *Computer Vision and Pattern Recognition (CVPR)*, 2016. arXiv:1609.07228. doi: 10.48550/arXiv.1609.07228.

[77] S.D. Bhatt et al., "Diagnosis of pulmonary nodules on CT images using YOLOv4", *International Journal of Online and Biomedical Engineering (IJOE)*, vol. 18, no. 5, pp. 131–146, 2021. doi: 10.3991/ijoe.v18i05.29529.

[78] S. Ganesan, T.S. Subashini and K. Jayalakshmi, "Classification of x-rays using statistical moments and SVM", In *IEEE Proc.International Conference on Communications and Signal Processing*, pp. 1109-1112, 2014. doi: 10.1109/ICCSP.2014.6950020.

[79] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998. doi: 10.1023/A:1009715923555.

[80] Y. Ma and G. Guo, "Support vector machines applications", *Lausanne: Springer Science and Business Media*, 2014. doi: 10.1007/978-3-319-02300-7.

[81] D.P. Kaucha et al., "Early detection of lung cancer using SVM classifier in biomedical image processing", In *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pp. 3143-3148, 2017. doi: 10.1109/ICPCSI.2017.8392305.

[82] S. Makaju et al., "Lung cancer detection using CT scan images", *Proc. Computer Science*, vol. 125, pp. 107-114, 2018. doi: 10.1016/j.procs.2017.12.016.

[83] J. Yao et al., "Computer-aided diagnosis of pulmonary infections using texture analysis and support vector machine classification", *Academic Radiology*, vol. 18, no. 3, pp. 306-314, 2011. doi: 10.1016/j.acra.2010.11.013.

[84] H. Shao, L. Cao and Y. Liu, "A detection approach for solitary pulmonary nodules based on CT images", In *Proc. of 2nd International Conference on Computer Science and Network Technology*, pp. 1253-1257, 2012. doi: 10.1109/ICCSNT.2012.6526151.

[85] F. Ciompi et al., "Towards automatic pulmonary nodule management in lung cancer screening with deep learning", *Scientific Report*, vol. 7, no. 46479, pp. 1-10, 2017. doi: 10.1038/srep46479.

[86] S.D. Bhatt et al., "Automated system for lung nodule classification based on ResNet-50 and SVM", In *3rd IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT-2022)*, pp. 1-5, 2022. doi: 10.1109/ICICT55121.2022.10064515.

[87] GE Hinton, S. Osindero and Y-W Teh, "A fast learning algorithm for deep belief nets", *Neural Computation*, vol. 18, no. 7, pp. 1527-1554, 2006. doi: 10.1162/neco.2006.18.7.1527.

[88] Y. Bengio et al., "Greedy layer-wise training of deep networks", In *Proc.the 19th International Conference on Neural Information Processing Systems (NIPS'06)*, pp. 153-160, 2006.

[89] T. Mikolov et al., "Empirical evaluation and combination of advanced language modeling techniques", In *International Speech Communication Association* (*INTERSPEECH) ISCA*, pp 605–608, 2011.

[90] GE Dahl et al., "Phone recognition with the mean-covariance restricted Boltzmann machine", *Advances in Neural Information Processing Systems*, vol. 1, pp. 469-477, 2010.

[91] L. Deng et al., "Binary coding of speech spectrograms using a deep auto-encoder", In *11th Annual Conference of the International Speech Communication Association* (*INTERSPEECH) ISCA*, pp. 1692-1695, 2010.

[92] D. Yu, F. Seide and G. Li, "Conversational speech transcription using context-dependent deep neural networks", In *29th International Conference on Machine Learning*, pp. 1-2, 2012.

[93] A. Krizhevsky, I. Sutskever and GE Hinton, "ImageNet classification with deep convolutional neural networks", *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017. doi: 10.1145/3065386.

[94] A. Karpathy et al., "Large-scale video classification with convolutional neural networks", In *Proc. IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1725-1732, 2014. doi: 10.1109/CVPR.2014.223.

[95] Y. Bengio, A. Courville and P. Vincent, "Representation learning: A review and new perspectives", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, 2013. doi: 10.1109/TPAMI.2013.50.

[96] Y. Bengio, "Learning deep architectures for AI", *Foundations and trends in Machine Learning*, vol. 2, no.

1, pp. 1-127, 2009. doi: 10.1561/2200000006.

[97] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436-444, 2015. doi: 10.1038/nature14539.

[98] V. Makde et al., "Deep neural network-based classification of tumourous and non-tumourous medical images", *Information and Communication Technology for Intelligent Systems*, vol. 84, pp. 199-206, 2017. doi: 10.1007/978-3-319-63645-0_22.

[99] R. Paul et al., "Combining deep neural network and traditional image features to improve survival predication accuracy for lung cancer patients from diagnostic CT", In *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 2570-2575, 2016. doi: 10.1109/SMC.2016.7844626.

[100] P. Monkam et al., "CNN models discriminating between pulmonary micro-nodules and non-nodules from CT images", *BioMedical Engineering Online*, vol. 17, no. 1, pp. 1-16, 2018. doi: 10.1186/s12938-018-0529-x.

[101] P. Monkam et al., "Ensemble learning of multiple-view 3D-CNNs model for micro-nodules identification in CT images", *IEEE Access*, vol. 7, pp. 5564–5576, 2019. doi: 10.1109/ACCESS.2018.2889350.

[102] M. Kafi, M. Maleki and N. Davoodian, "Functional histology of the ovarian follicles as determined by follicular fluid concentrations of steroids and IGF-1 in Camelus dromedarius", *Research in Veterinary Science*, vol. 99, pp. 37-40, 2015. doi: 10.1016/j.rvsc.2015.01.001.

[103] H-C Shin et al., "Deep convolutional neural networks for computer aided detection: CNN architectures, dataset characteristics and transfer learning", *IEEE Transactions Medical Imaging*, vol. 35, pp. 1285-1298, 2016. doi: 10.1109/TMI.2016.2528162.

[104] F. Ciompi et al., "Automatic classification of pulmonary peri-fissural nodules in computed tomography using an ensemble of 2D views and a convolutional neural network out-of-the-box", *Medical Image Analysis*, vol. 26, no. 1, pp. 195-202, 2015. doi: 10.1016/j.media.2015.08.001.

[105] D. Kumar, A. Wong and D.A. Clausi, "Lung nodule classification using deep features in CT images", In *12th IEEE Conference on Computer and Robot Vision*, pp. 133-138, 2015. doi: 10.1109/CRV.2015.25.

[106] M.Z. Rehman et al., "Lungs cancer nodules detection from CT scan images with convolutional neural networks", In *R. Ghazali, N. Nawi, M. Deris, M., J. Abawajy (eds) Recent Advances on Soft Computing and Data Mining. Advances in Intelligent Systems and Computing*, pp. 382-391, 2020. doi: 10.1007/978-3-030-36056-6_36.

[107] H. Peng, H. Sun and Y. Guo, "3D multi-scale deep convolutional neural networks for pulmonary nodule detection", *PLoS ONE*, vol. 16, no. 1, pp. 1-14, 2021. doi: 10.1371/journal.pone.0244406.

[108] W. Shen et al., "Multi-scale convolutional neural networks for lung nodule classification", *Information Processing in Medical Imaging*, vol. 24, pp. 588-599, 2015. doi: 10.1007/978-3-319-19992-4_46.

[109] J. Lyu and S. Ling, "Using multi-level convolutional neural network for classification of lung nodules on CT images", In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 686-689, 2018. doi: 10.1109/EMBC.2018.8512376.

[110] A.A.A. Setio et al., "Pulmonary nodule detection in CT images: false positive reduction using multi-view convolutional networks", *IEEE Transactions Medical Imaging*, vol. 35, no. 5, pp. 1160–1169, 2016. doi: 10.1109/TMI.2016.2536809.

[111] KL Hua et al., "Computer-aided classification of lung nodules on computed tomography images via deep learning technique", *OncoTargets and Therapy*, vol. 8, pp. 2015-2022, 2015. doi: 10.2147/OTT.S80733.

[112] W. Shen et al., "Multi-crop convolutional neural networks for lung nodule malignancy suspiciousness classification", *Pattern Recognition*, vol. 61, pp. 663-673, 2017. doi: 10.1016/j.patcog.2016.05.029.

[113] M. Sandler et al., "MobileNetv2: inverted residuals and linear bottlenecks", In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510-4520, 2018. doi: 10.1109/CVPR.2018.00474.

[114] A.G. Howard et al., "MobileNets: efficient convolutional neural networks for mobile vision applications", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9, 2017. doi: 10.48550/arXiv.1704.04861.

[115] M.S. Rahman, P.C. Shill and Z. Homayra, "A new method for lung nodule detection using deep neural networks for CT images", In *International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 1-6, 2019. doi: 10.1109/ECACE.2019.8679439.

[116] A. Souid, S. Nizar and S. Hedi, "Classification and predictions of lung diseases from chest X-rays using MobileNet V2", *Applied Sciences*, vol. 11, no. 6:2751, pp. 1-16, 2021. doi: 10.3390/app11062751.

[117] Q. Zhang et al., "Lung nodule diagnosis on 3D computed tomography images using deep convolutional neural networks", *Procedia Manufacturing*, vol. 39, pp. 363-370, 2019. doi: 10.1016/j.promfg.2020.01.375.

[118] NI Forrest et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 mb model size", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-13, 2016. doi: 10.48550/arXiv.1602.07360.

[119] M. Polsinelli, L. Cinque and G. Placidi, "A light CNN for detecting COVID-19 from CT scans of the chest", *Pattern Recognition Letter*, vol. 140, pp. 95-100, 2020. doi: 10.1016/j.patrec.2020.10.001.

[120] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition", *Neural computation*, vol. 1, no. 4, pp. 541- 551, 1989. doi: 10.1162/neco.1989.1.4.541.

[121] Y. LeCun et al., "Learning algorithms for classification: A comparison on handwritten digit recognition", *Neural networks Stat Mech Perspect*, pp. 261-276, 1995.

[122] N. Khehrah et al., "Lung nodule detection in CT images using statistical and shape-based features", *Journal Imaging*, vol. 6, no. 2, pp. 1-6, 2020. doi: 10.3390/jimaging6020006.

[123] Y. Xie et al., "Fusing texture, shape and deep model-learned information at decision level for automated classification of lung nodules on chest CT", *Information Fusion*, vol. 42, pp. 102-110, 2018. doi: 10.1016/j.inffus.2017.10.005.

[124] I. Nasser and S. Abu-Naser, "Lung cancer detection using artificial neural network", *International Journal of Engineering and Information Systems*, vol. 3, no. 3, pp. 17–23, 2019.

[125] M.B.A. Miah and M.A. Yousuf, "Detection of lung cancer from CT image using image processing and neural network", In *Proc. International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, pp. 1-6, 2015. doi: 10.1109/ICEEICT.2015.7307530.

[126] S. Zhang et al., "Computer-aided diagnosis (CAD) of pulmonary nodule of thoracic CT image using transfer learning", *Journal Digital Imaging*, vol. 32, no. 6, pp. 995–1007, 2019. doi: 10.1007/s10278-019-00204-4.

[127] Y. Bengio, P. Simard and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult", *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994. doi: 10.1109/72.279181.

[128] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", In *Proc. 13th International Conference on Artificial Intelligence and Statistics, PMLR*, vol. 9, pp. 249-256, 2010.

[129] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions", *International Journal of Uncertainty, Fuzziness and Knowledge-Based System,* vol. 6, no. 2, pp. 107-116, 1998. doi: 10.1142/S0218488598000094.

[130] V. Nair and GE Hinton, "Rectified linear units improve restricted Boltzmann machines", In *Proc. 27th International Conference on Machine Learning (ICML)*, pp. 807-814, 2010. doi: 10.5555/3104322.3104425.

[131] Y. LeCun et al., "Efficient backprop", *Neural Networks: Tricks of the Trade*, vol. 7700, pp. 9–50, 1998. doi: 10.1007/978-3-642-35289-8_3.

[132] A.M. Saxe, J.L. McClelland and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks", *Neural and Evolutionary Computing,* pp. 1-22, 2013. doi: 10.48550/arXiv.1312.6120.

[133] GE Dahl, TN Sainath and GE Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout", In *IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8609-8613, 2013. doi: 10.1109/ICASSP.2013.6639346.

[134] N. Srivastava et al., "Dropout: a simple way to prevent neural networks from overfitting". *Journal of Machine Learning Research,* vol. 15, no. 1, pp. 1929-1958, 2014. doi: 10.5555/2627435.2670313.

[135] P. Gupta and A.P. Shukla, "Improving accuracy of lung nodule classification using AlexNet model", In *Proc. International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES),* pp. 1–6, 2021. doi: 10.1109/ICSES52305.2021.9633903.

[136] X. Zhao et al., "Agile convolutional neural network for pulmonary nodule classification using CT images", *International Journal of Computer Assisted Radiology*, vol. 13, no. 4, pp. 585–595, 2018. doi: 10.1007/s11548-017-1696-0.

[137] H.F. Al-Yasriy et al., "Diagnosis of lung cancer based on CT scans using CNN", *IOP Conference Series: Materials Science and Engineering*, vol. 928, pp. 1-9, 2020. doi: 10.1088/1757-899X/928/2/022035.

[138] A. Agarwal, K. Patni and D. Rajeswari, "Lung cancer detection and classification based on AlexNet CNN", In *Proc. 6th International Conference on Communication and Electronics Systems (ICCES)*, pp. 1390-1397, 2021. doi: 10.1109/ICCES51350.2021.9489033.

[139] A. Bhandary et al., "Deep-learning framework to detect lung abnormality-A study with chest X-ray and lung CT scan images", *Pattern Recognition Letters*, vol. 129, pp. 271–278, 2020. doi: 10.1016/j.patrec.2019.11.013.

[140] C. Szegedy et al., "Going deeper with convolutions", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 1–9. 2015. doi: 10.1109/CVPR.2015.7298594.

[141] C. Szegedy et al., "Rethinking the Inception architecture for computer vision", In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818-2826, 2016. doi: 10.1109/CVPR.2016.308.

[142] R.S. Bose and S.V. Kumar, "Hand gesture recognition using faster R-CNN Inception-V2 model", In *Proc. Advances in Robotics 2019 (AIR 2019), Association for Computing Machinery*, no.19, pp. 1–6, 2020. doi: 10.1145/3352593.3352613.

[143] M. Lin, Q. Chen and S. Yan, "Network in network", *Neural and Evolutionary Computing,* pp. 1–10, 2013. doi: 10.48550/arXiv.1312.4400.

[144] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal

covariate shift", In *32nd International Conference on Machine Learning (ICML)*, 2015. doi: 10.48550/arXiv.1502.03167.

[145] YN Dauphin, H. Vries and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization", *Advances in Neural Information Processing Systems*, pp. 1504-1512, 2015. doi: 10.48550/arXiv.1502.04390.

[146] T. Fang, "A novel computer-aided lung cancer detection method based on transfer learning from GoogLeNet and median intensity projections", In *IEEE International Conference on Computer and Communication Engineering Technology (CCET)*, pp. 286-290, 2018. doi: 10.1109/CCET.2018.8542189.

[147] MS AL-Huseiny and AS Sajit, "Transfer learning with GoogLeNet for detection of lung cancer", *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 2, pp. 1078-1086, 2021. doi: 10.11591/ijeecs.v22.i2.pp1078-1086.

[148] G. Zheng, G. Han and N.Q. Soomro, "An Inception module CNN classifiers fusion method on pulmonary nodule diagnosis by signs", *Tsinghua Science and Technology*, vol. 25, no. 3, pp. 368–383, 2019. doi: 10.26599/TST.2019.9010010.

[149] K. He et al., "Deep residual learning for image recognition", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016. doi: 10.1109/CVPR.2016.90.

[150] K. He and J. Sun, "Convolutional neural networks at constrained time cost", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5353-5360, 2015. doi: 10.1109/CVPR.2015.7299173.

[151] T-Y Lin et al., "Microsoft coco: common objects in context", In *European Conference on Computer Vision*, vol. 8693, pp 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.

[152] S. Bhatia, Y. Sinha and L. Goel, "Lung cancer detection: A deep learning approach", *Soft Computing for Problem Solving*, vol. 817, pp. 699-705, 2019. doi: 10.1007/978-981-13-1595-4_55.

[153] T.K. Sajja, R.M. Devarapalli and H.K. Kalluri, "Lung cancer detection based on CT scan images by using deep transfer learning", *Traitement Signal*, vol. 36, pp. 339–344, 2019. doi: 10.18280/ts.360406.

[154] A. Nibali, Z. He and D. Wollersheim, "Pulmonary nodule classification with deep residual networks", *International Journal of Computer Assisted Radiology and Surgery*, vol. 12, no. 10, pp. 1799–1808, 2017. doi: 10.1007/s11548-017-1605-6.

[155] L. Haibo et al., "An improved YOLOv3 algorithm for pulmonary nodule detection", In *Proc. IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, pp. 1068–1072, 2021. doi: 10.1109/IMCEC51613.2021.9482291.

[156] S. Cui et al., "Development and clinical application of deep learning model for lung nodules screening on CT images", *Scientific Reports*, vol. 10, no. 1, pp. 13657, 2020. doi: 10.1038/ s41598-020-70629-3.

[157] M. Saric et al., "CNN-based method for lung cancer detection in whole slide histopathology images", In *4th IEEE International Conference on Smart and Sustainable Technologies (SpliTech)*, pp. 1-4, 2019. doi: 10.23919/SpliTech.2019.8783041.

[158] K. Simonyan, A. Vedaldi and A. Zisserman, "Deep inside convolutional networks: visualising image classification models and saliency maps", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2013. doi: 10.48550/arXiv.1312.6034.

[159] M. Ranzato et al., "Unsupervised learning of invariant feature hierarchies with applications to object recognition", In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*

(CVPR), pp 1-8, 2007. doi: 10.1109/CVPR.2007.383157.

[160] L. Geng et al., "Lung segmentation method with dilated convolution based on VGG-16 network", *Computer Assisted Surgery*, vol. 24 (Suppl. 2), pp. 27-33, 2019. doi: 10.1080/24699322.2019.1649071.

[161] S. Pang et al., "VGG16-T: A novel deep convolutional neural network with boosting to identify pathological type of lung cancer in early stage by CT images", *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, pp. 771-780, 2020. doi: 10.2991/ijcis.d.200608.001.

[162] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-14, 2015. arXiv:1409.1556.

[163] J. Dehmeshki et al., "Automated detection of lung nodules in CT images using shape-based genetic algorithm", *Computerized Medical Imaging and Graphics*, vol. 31, no. 6, pp. 408–417, 2007. doi: 10.1016/j.compmedimag.2007.03.002.

[164] X. Ye et al., "Shape-based computer-aided detection of lung nodules in thoracic CT images", *IEEE Transactions Biomedical Engineering*, vol. 56, no. 7, pp. 1810–1820, 2009. doi: 10.1109/TBME.2009.2017027.

[165] WJ Choi and TS Choi, "Automated pulmonary nodule detection based on three-dimensional shape-based feature descriptor", *Computer Methods and Programs in Biomedicine*, vol. 113, no. 1, pp. 37-54, 2014. doi: 10.1016/j.cmpb.2013.08.015.

[166] A. Elnakib et al., "Early lung cancer detection using deep learning optimization", *International Journal of Online and Biomedical Engineering (IJOE)*, vol. 16, no. 6, pp. 82-94, 2020. doi: 10.3991/ijoe.v16i06.13657.

[167] S.D. Bhatt and H.B. Soni, "Improving classification accuracy of pulmonary nodules using simplified deep neural network", *The Open Biomedical Engineering*, vol. 15 (Suppl-2- M7), pp. 180-189, 2021. doi: 10.2174/1874120702115010180.

[168] Y. Zhu et al., "Attention CoupleNet: fully convolutional attention coupling network for object detection", *IEEE Transactions Image Processing*, vol. 28, no. 1, pp. 113-126, 2019. doi: 10.1109/TIP.2018.2865280.

[169] R. Girshick et al., "Rich feature hierarchies for accurate object detection and semantic segmentation", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580-587, 2014. doi: 10.1109/CVPR.2014.81.

[170] J.U. Kim and Y.M. Ro, "Attentive layer separation for object classification and object localization in object detection", In *IEEE International Conference on Image Processing (ICIP)*, pp. 3995-3999, 2019. doi: 10.1109/ICIP.2019.8803439.

[171] C. Peng et al., "Graphical representation for heterogeneous face recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 301-312, 2015. doi: 10.1109/TPAMI.2016.2542816.

[172] C. Peng et al., "Face recognition from multiple stylistic sketches: Scenarios, datasets and evaluation", *Pattern Recognition*, vol. 84, pp. 262-272, 2016. doi: 10.1016/j.patcog.2018.07.014.

[173] C. Chen et al., "Deepdriving: learning affordance for direct perception in autonomous driving", In *IEEE International Conference on Computer Vision*, pp. 2722-2730, 2015. doi: 10.1109/ICCV.2015.312.

[174] N. Ganatra, "A comprehensive study of applying object detection methods for medical image analysis", In *8th International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 821-826, 2021. doi: 10.1109/INDIACom51348.2021.00147.

[175] K. Yan et al., "DeepLesion: automated deep mining, categorization and detection of significant radiology

image findings using large-scale clinical lesion annotations", *Computer Vision and Pattern Recognition (CVPR),* pp. 1-9, 2017. doi: 10.48550/arXiv.1710.01766.

[176] K. Yan, M. Bagheri and R.M. Summers, "3D context enhanced region-based convolutional neural network for end-to-end lesion detection", In *A. Frangi, J. Schnabel, C. Davatzikos, C. Alberola-López, G. Fichtinger (eds) Medical Image Computing and Computer Assisted Intervention - MICCAI 2018. Lecture Notes in Computer Science,* vol. 11070, pp. 511-519, 2018. doi: 10.1007/978-3-030-00928-1_58.

[177] Z. Li et al., "R-FCN++: towards accurate region-based fully convolutional networks for object detection", In *Proc. AAAI Conference on Artificial Intelligence,* vol. 32, no. 1, 2018. doi: 10.1609/aaai.v32i1.12265.

[178] P.F. Felzenszwalb et al., "Object detection with discriminatively trained part-based models", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 32, no. 9, pp. 1627-1645, 2010. doi: 10.1109/TPAMI.2009.167.

[179] TY Lin et al., "Focal loss for dense object detection", In *IEEE International Conference on Computer Vision,* pp. 2999-3007, 2017. doi: 10.1109/ICCV.2017.324.

[180] S. Ren et al., "Faster R-CNN: towards real-time object detection with region proposal networks", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2017. doi: 10.1109/TPAMI.2016.2577031.

[181] G. Subrata, "A deeper look at how Faster-RCNN works", 2018. whatdhack.medium.com/a-deeper-look-at-how-faster-rcnn-works-84081284e1cd [Online].

[182] H. Yanagisawa, T. Yamashita and H. Watanabe, "A study on object detection method from manga images using CNN", In *IEEE International Workshop on Advanced Image Technology (IWAIT),* pp. 1–4, 2018. doi: 10.1109/IWAIT.2018.8369633.

[183] H. Zhang, Y. Peng and Y. Guo, "Pulmonary nodules detection based on multi-scale attention networks", *Scientific Reports,* vol. 12, no. 1466, pp. 1-14, 2022. doi: 10.1038/s41598-022-05372-y.

[184] T. Wang et al., "Few-shot adaptive Faster R-CNN", In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 7166-7175, 2019. doi: 10.1109/CVPR.2019.00734.

[185] J. Gu, Z. Tian and Y. Qi, "Pulmonary nodules detection based on deformable convolution", *IEEE Access,* vol. 8, pp. 16302-16309, 2020. doi: 10.1109/ACCESS.2020.2967238.

[186] J. Ding et al., "Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks", In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 559–567, 2017. doi: 10.1007/978-3-319-66179- 7-64.

[187] H. Xie et al., "Automated pulmonary nodule detection in CT images using deep convolutional neural networks", *Pattern Recognition*, vol. 85, pp. 109–119, 2019. doi: 10.1016/j.patcog.2018.07.031.

[188] W. Zhu et al., "DeepLung: deep 3D dual path nets for automated pulmonary nodule detection and classification", In *IEEE Winter Conference on Applications of Computer Vision (WACV),* pp. 673–681, 2018. doi: 10.1109/WACV.2018.00079.

[189] Y. Su, D. Li and X. Chen, "Lung nodule detection based on faster R-CNN framework", *Computer Methods and Programs in Biomedicine,* vol. 200, 2021. doi: 10.1016/j.cmpb.2020.105866.

[190] D. Ardila et al., "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography", *Nature Medicine,* vol. 25, no. 6, pp. 954-961, 2019. doi: 10.1038/s41591-019-0447-x.

[191] Y. Li et al., “Lung nodule detection with deep learning in 3D thoracic MR images”, *IEEE Access*, vol. 7, pp. 37822–37832, 2019. doi: 10.1109/ACCESS.2019.2905574.

[192] J. Wang et al., “Pulmonary nodule detection in volumetric chest CT scans using CNNs-based nodule-size-adaptive detection and classification”, *IEEE Access*, vol. 7, pp. 46033–46044, 2019. doi: 10.1109/ACCESS.2019.2908195.

[193] F. Liao et al., “Evaluate the malignancy of pulmonary nodules using the 3-D deep leaky noisy-or network”, *IEEE Transactions Neural Networks*, vol. 30, no. 11, pp. 3484-3495, 2019. doi: 10.1109/TNNLS.2019.2892409.

[194] C. Tong et al., “Pulmonary nodule detection based on ISODATA improved Faster R-CNN and 3D-CNN with focal loss”, *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 16, no. 1, pp. 1–9, 2020. doi: 10.1145/3365445.

[195] R. Fan, SI Kamata and Y. Chen, “Pulmonary nodule detection using improved faster R-CNN and 3D ResNet”, In *Proc. 13th International Conference on Digital Image Processing (ICDIP)*, 2021. doi: 10.1117/12.2599884.

[196] A. Neubeck and L. Van Gool, “Efficient non-maximum suppression”, In *18th International Conference on Pattern Recognition (ICPR'06)*, pp. 850–855, 2006. doi: 10.1109/ICPR.2006.479.

[197] W. Liu et al., “SSD: single shot multibox detector”, In *Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. Lecture Notes in Computer Science*, vol. 9905, pp. 21–37, 2016. doi: 10.1007/978-3-319-46448-0_2.

[198] N. Khosravan and U. Bagci, ‘‘S4ND: single-shot single-scale lung nodule detection”, In *A. Frangi, J. Schnabel, C. Davatzikos, C. Alberola-López, G. Fichtinger (eds) Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. Lecture Notes in Computer Science*, vol. 11071, pp. 794–802, 2018. doi: 10.1007/978-3-030-00934-2_88.

[199] J. Redmon et al., “You only look once: unified, real-time object detection”, In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, 2016. doi: 10.1109/CVPR.2016.91.

[200] Q. Cheng et al., “A survey and analysis on automatic image annotation”, *Pattern Recognition*, vol. 79, pp. 242–259, 2018. doi: 10.1016/j.patcog.2018.02.017.

[201] MA Al-Masni et al., “Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system”, *Computer Methods and Programs in Biomedicine*, vol. 157, pp. 85-94, 2018. doi: 10.1016/j.cmpb.2018.01.017.

[202] R.S. Sindhu et al., “Using YOLO based deep learning network for real time detection and localization of lung nodules from low dose CT scans”, *Medical Imaging: Computer-Aided Diagnosis*, vol. 10575, pp. 1-9, 2018. doi: 10.1117/12.2293699.

[203] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger”, In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6517-6525, 2017. doi: 10.1109/CVPR.2017.690.

[204] C.M. Yang, Y. Choo and S. Park, “Semi-automatic image and video annotation system for generating ground truth information”, In *IEEE International Conference on Information Networking (ICOIN)*, pp. 821–824, 2018. doi: 10.1109/ICOIN.2018.8343233.

[205] X-Z Li et al., “YOLO V2 network with asymmetric convolution kernel for lung nodule detection of CT image”, *Chinese Journal of Biomedical Engineering*, vol. 38, no. 4, pp. 401-408, 2019. doi: 10.3969/j.issn.0258-8021.2019.04.3.

[206] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-6, 2018. doi: 10.48550/arXiv.1804.02767.

[207] T-Yi Lin et al., "Feature pyramid networks for object detection", In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944, 2017. doi: 10.1109/CVPR.2017.106.

[208] J. Liu et al., "Accurate and robust pulmonary nodule detection by 3D feature pyramid network with self-supervised feature learning", *Image and Video Processing, pp. 1-15, 2019. arXiv:1907.11704*.

[209] Z. Li et al., "Lightweight ship detection methods based on YOLOv3 and denseNet", *Mathematical Problems in Engineering*, pp. 1-10, 2020. doi: 10.1155/2020/4813183.

[210] G.H. Aly et al., "YOLO based breast masses detection and classification in full-field digital mammograms", *Computer Methods and Programs in Biomedicine*, vol. 200, no. 105823, 2021. doi: 10.1016/j.cmpb.2020.105823.

[211] K. Xu, H. Jiang and W. Tang, "A new object detection algorithm based on YOLOv3 for lung nodules", In *Proc. 6th International Conference on Computing and Artificial Intelligence*, pp. 233-239, 2020. doi: 10.1145/3404555.3404609.

[212] C-Y Wang, A. Bochkovskiy and H-Y Mark Liao, "YOLOv4: optimal speed and accuracy of object detection", *Computer Vision and Pattern Recognition (CVPR)*, pp. 1-17, 2020. doi: 10.48550/arXiv.2004.10934.

[213] K. He et al., "Spatial pyramid pooling in deep convolutional networks for visual recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904-1916, 2015. doi: 10.1109/TPAMI.2015.2389824.

[214] L. Shu et al., "Path aggregation network for instance segmentation", In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8759–8768, 2018. doi: 10.1109/CVPR.2018.00913.

[215] C-Y Wang, A. Bochkovskiy and H-Y Mark Liao, "Scaled-YOLOv4: scaling cross stage partial network", In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13024-13033, 2021. doi: 10.1109/CVPR46437.2021.01283.

[216] Z. Zakria et al., "Multiscale and direction target detecting in remote sensing images via modified YOLOv4", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1039-1048, 2022. doi: 10.1109/JSTARS.2022.3140776.

[217] C-M Yan and C. Wang, "Automatic detection and localization of pulmonary nodules in CT images based on YOLOv5", *Journal of Computers*, vol. 33, no. 3, pp. 113-123, 2022. doi: 10.53106/199115992022063303009.

[218] Y. Zehua et al., "Research on non-pooling YOLOv5 based algorithm for the recognition of randomly distributed multiple types of parts", *Sensors*, vol. 22, no. 23: 9335, 2022. doi: 10.3390/s22239335.

[219] K. Liu, "STBi-YOLO: A real-time object detection method for lung nodule recognition", *IEEE Access*, vol. 10, pp. 75385-75394, 2022. doi: 10.1109/ACCESS.2022.3192034.

[220] J. Chen et al., "Object detection for graphical user interface: old fashioned or deep learning or a combination?", In *Proc. 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020)*, pp. 1202–1214, 2020. doi: 10.1145/3368089.3409691.

List of Publications

1. S.D. Bhatt and H.B. Soni, “Improving classification accuracy of pulmonary nodules using simplified deep neural network”, *The Open Biomedical Engineering*, vol. 15 (Suppl2- M7), pp. 180-189, 2021. doi: 10.2174/1874120702115010180. [Scopus]
2. S.D. Bhatt et al., “Diagnosis of pulmonary nodules on CT images using YOLOv4”, *International Journal of Online and Biomedical Engineering (IJOE)*, vol. 18, no. 05, pp. 131–146, 2022. doi: 10.3991/ijoe.v18i05.29529. [Scopus, Clarivate Analytics ESCI]
3. S.D. Bhatt and H.B. Soni, “Image retrieval using bag-of-features for lung cancer classification”, In *6th International IEEE Conference on Inventive Computation Technologies (ICICT)*, pp. 531-536, 2021. doi: 10.1109/ICICT50816.2021.9358499.
4. S.D. Bhatt et al., “Automated system for lung nodule classification based on ResNet-50 and SVM”, In *3rd IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT-2022)*, pp. 1-5, 2022. doi: 10.1109/ICICT55121.2022.10064515.