A PROJECT REPORT ON

COVID-19 DETECTION
BY LUNG CT SCAN USING DEEP NEURAL NETWORK

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING
(COMPUTER ENGINEERING)

SUBMITTED BY

MCEBO PATEGUANA	**B150024210**
PRASHANT SHARMA	**B150024217**
THAWATCHAI YANGO	**B150024220**

DEPARTMENT OF COMPUTER ENGINEERING

MAEER'S
MAHARASHTRA INSTITUTE OF TECHNOLOGY
PAUD RD-KOTHRUD, PUNE 411038

SAVITRIBAI PHULE PUNE UNIVERSITY
2020-2021

CERTIFICATE

This is to certify that the project report entitles

"COVID-19 DETECTION
BY LUNG CT SCAN USING DEEP NEURAL NETWORK"

Submitted by

MCEBO PATEGUANA	**B150024210**
PRASHANT SHARMA	**B150024217**
THAWATCHAI YANGO	**B150024220**

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Prof. Vandana Jagtap** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering (Computer Engineering).**

(Prof. Vandana Jagtap)
Guide
Department of Computer Engineering

(Dr.V.Y.Kulkarni)
Head,
Department of Computer Engineering

(Dr. L. K. Kshirsagar)
Principal,
MAEER'S Maharashtra Institute of Technology, Pune 38

Place: Pune
Date:

ACKNOWLEDGEMENT

It gives us immense pleasure to express my sincere gratitude to all the helping hands who have guided us in the completion of this project. It was a great learning experience for us to work on our project. As student of MIT PUNE in Faculty of Engineering and technology (FET), under Savitribai Phule Pune University (SPPU), Pune; We, **Mcebo Pateguana, Prashant Sharma** and **Thawatchai Yango** got an opportunity of doing work on project. We took up the project with strong determination and tried to put in the best of our effort

The project done by us is about "Disease Detection using Lung CT scan". This work is supported in a Covid-19 detection using Convolutional Neural Network and this project could not have been completed without the help of our faculty guide **Prof.Vandana Jagtap** for providing the facility to carry out the research work.

MCEBO PATEGUANA B150024210

PRASHANT SHARMA B150024217

THAWATCHAI YANGO B150024220

ABSTRACT

Medical data growth is in healthcare communities, accurate analysis of medical data benefit early disease detection, patient care and community services. However, the analysis of patients is depends on accuracy of diagnosis and then treatment as well. The increasing number of cases of confirmed coronavirus disease (COVID-19) in China is striking.

The purpose of this study was to investigate the relation between chest CT findings and the clinical conditions of COVID-19 pneumonia. Among those who develop symptoms, most (about 80%) recover from the disease without needing hospital treatment. About 15% become seriously ill and require oxygen and 5% become critically ill and need intensive care. Complications leading to death may include respiratory failure, acute respiratory distress syndrome (ARDS), sepsis and septic shock, thromboembolism, and/or multi organ failure, including injury of the heart, liver or kidneys. In rare situations, children can develop a severe inflammatory syndrome a few weeks after infection.

Proposed method not only detects the availability of NOVEL CORONA but also it tracks the treatment progress. In Second generation, number of architectures or algorithms is present for classification problem. In other languages we have to start from scratch, but for MATLAB and Python this is another case. Simply calling those functions and changing the input argument, you test.

Due to available built in commands, design and development time get reduced. With minimal Mathematics behind deep learning, we can design and test various architectures of neural network.

TABLE OF CONTENTS

LIST OF ABBREVATIONS i

LIST OF FIGURES ii

LIST OF TABLES iii

CHAPTER TITLE PAGE NO.

Sr. No. Title of Chapter Page No.

01 Introduction 1

1.1 Introduction 1

1.2 Overview 1

1.3 Motivation 1

1.4 Project Goals 2

1.5 Problem Definition & Objectives 2

1.6 Project Scope & Limitations 3

1.7 Deep Learning Integration 3

1.8 Advantages 3

02 Literature Survey 4

2.1 Background 4

2.2 Table of Literature Survey 5-7

2.3 Methodology for literature survey 1 7-8

2.4 Methodology for literature survey 2 8-9

2.5 Methodology for literature survey 3 9-10

2.6 Methodology for literature survey 4 10-11

2.7 Methodology for literature survey 5 11-12

03 Software Requirements Specification 13

3.1 Assumptions and Dependencies 13

3.2 Functional Requirements 13

3.2.1 System Feature 1(Functional Requirement) 13

3.2.2 System Feature2 (Functional Requirement) 13

3.3 External Interface Requirements 13

3.3.1 User Interfaces 13-14

3.3.2 Hardware Interfaces 14

3.3.3 Software Interfaces 14

3.3.4 Communication Interfaces 14
3.4 Nonfunctional Requirements 14
3.4.1 Performance Requirements 14-15
3.4.2 Safety Requirements 15
3.4.3 Security Requirements 15
3.4.4 Software Quality Attributes 15
3.5 System Requirements 16
3.5.1 Database Requirements 16
3.5.2 Software Requirements 16
3.5.3 Hardware Requirements 16
3.6 Analysis Models: SDLC Model to be applied 16
3.6.1 Waterfall Mode 16
3.6.1.1 Requirement analysis 17
3.6.1.2 System Design 17
3.6.1.3 Implementation 17
3.6.1.4 Testing 17
3.6.1.5 Deployment 17
3.6.1.5 Maintenance 17-18
04 System Design 19
4.1 System Architecture 19
4.1.1 Proposed System Architecture 19-20
4.1.2 System Analysis Proposed Architecture 20-21
4.2 Data Flow Diagrams 21
4.3.1 DFD Level 0 22
4.3.2 DFD Level 1 22
4.3 Entity Relationship Diagrams 23
4.4 UML Diagrams 23
4.4.1 Use case diagram 23-24
4.4.2 Activity diagrams 24-25
4.4.3 Class Diagram 26
4.4.4 Sequence Diagram 27
4.4.5 Component Diagram 28

4.4.6	Deployment Diagram	28
4.4.7	State Diagram	29
4.5	User Interface Design	30
4.5.1	Registration Form Page	30
4.5.2	Login Form Page	30
4.5.3	Covid-19 Test Page	31
4.5.4	User Report in PDF	31
05	**Project Plan**	32
5.1	System Implementation Plan	32
06	**Project Implementation**	33
6.1	Overview of Project Modules	33
6.2	Tools and Technologies Used	34-35
6.3	Algorithm Details	35
6.3.1	Convolutional Neural Networks	35-37
6.3.2	ReLU	38
6.3.3	Pooling	38
6.3.4	GoogLeNet Architecture Layers and Parameters	39-44
6.3.5	Training Accuracy	44
6.3.6	Image Insert and Enhancement	45-46
07	**Software Testing**	47
7.1	Type of Testing	47
7.1.1	Unit Testing	47
7.1.2	Integration Testing	47-49
7.2	Test cases & Test Results	49-53
08	**Results**	54
8.1	Outcomes	54-56
8.2	Screen Shots	57-61
09	**Conclusions**	62
9.1	Conclusions	62
9.2	Limitations	62
9.3	Future Work	62
9.4	Advantages	62

9.5 Application 62

Appendix A: Problem statement feasibility assessment using, satisfy ability analysis and NP Hard, NP-Complete or P type using modern algebra and relevant mathematical models 63-65

Appendix B: Details of the papers referred in IEEE format (given earlier) Summary of the above paper in not more than 3-4 lines. Here you should write the seed idea of the papers you had referred for preparation of this project report in the following format. 66

Example:

Thomas Noltey, Hans Hanssony, Lucia Lo Belloz,"Communication Buses for Automotive Applications" In Proceedings of the 3rd Information Survivability Workshop (ISW-2007), Boston, Massachusetts, USA, October 2007. IEEE Computer Society.

Appendix C: Plagiarism Report 67

LIST OF ABBREVATIONS

ABBREVIATION	ILLUSTRATION
CNN	Conventional Neural Network

LIST OF FIGURES

FIGURE	ILLUSTRATION	PAGE NO.
3.1	Waterfall Model	18
4.1	Block Diagram of Proposed System	19
4.2	System Analysis Proposed Architecture	20
4.3	DFD Level – 0	22
4.4	DFD Level – 1	22
4.5	ER Diagram	23
4.6	Use case Diagram	24
4.7	Activity Diagram	25
4.8	Class Diagram	26
4.9	Sequence Diagram	27
4.10	Component Diagram	28
4.11	Deployment Diagram	28
4.12	User State Diagram	29
4.13	Admin State Diagram	29
4.14	Registration Form Page	30
4.15	Login Form Page	30
4.16	Covid-19 Test Page	31
4.17	User Report in PDF	31
6.1	A 3x3 convolution of depth 1 performed over a 5x5 input feature map, also of depth	36
6.2	Left: A 5x5 input feature map (depth 1). Right: a 3x3 convolution (depth 1)	37
6.3	Illustrates the end-to-end Structure of a convolutional neural network.	37
6.4	Left: Max pooling performed over a 4x4 feature map with a 2x2 filter and stride of 2. Right: the output of the max pooling operation. Note the resulting feature map is now 2x2, preserving only the maximum values from each tile.	38

6.5	GoogLeNet Architecture	39-41
6.6	GoogLeNet Parameters	42-44
6.7	Training Accuracy Graph	44
6.8	Original Image Input	45
6.9	Resizing Image	45
6.10	Median Image	46
6.11	Gray Conversions	46
8.1	Accuracy for all stages	54
8.2	Accuracy for normal stage	55
8.3	Accuracy for normal I	55
8.4	Accuracy for stage II	56
8.5	Accuracy for stage sever	56
8.6	User Registration UI	57
8.7	User Login UI	57
8.8	Input Images UI	58
8.9	Resized Images UI	58
8.10	Median Images UI	59
8.11	Gray Images UI	59
8.12	Covid-19 Detection UI	60
8.13	Accuracy UI	60
8.14	Report Covid Template PDF	61

LIST OF TABLES

FIGURE	ILLUSTRATION	PAGE NO.
1	Literature Survey	5-7
2	Project plan	32

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

- Medical data growth in healthcare communities, accurate analysis of medical data benefits of early disease detection, patient care and community services.
- However, Novel Corona virus has taken large attention of the entire globe. Everyone joined the battle to fight the Corona virus.
- As a part of society we develop the software for Corona detection using CNN.
- So we are proposing diagnosis system based on machine learning for giving promising solution with high accuracy.
- Patients with confirmed COVID-19 pneumonia have typical imaging features that can be helpful in early screening of highly suspected cases and in evaluation of the severity and extent of disease.

1.2 OVERVIEW

- The system requires a user to upload a Lung CT Scan. The system then processes the file and displays the images to the user. The user then chooses which scan he or she wants to predict then the application pre-processes the Lung CT Scan and infers the image to the predictive model. The output or prediction of the model is then displayed to the user which will display Covid-19 stage I,II and Sever stage or Non- Covid-19 to the User.

1.3 MOTIVATION

- The Corona outbreak has put significant pressure on imaging departments, to test hundreds of peoples per day. Patients and doctors typically have to wait a few hours to get the CT results, but our system is improving the CT diagnosis speed for each case; and each minute saved is critical to decrease the chance of cross-contamination at the hospital.

- The shortage of strict laboratory requirements for the use of the RT-PCR detection kit, to confirm the 2019-nCoV diagnosis, is a major problem. Proposed system can help with limited medical resources to immediately screen out suspected Corona virus-infected patients for further diagnosis and treatment.
- The battle against this epidemic is one being fought by all clinicians and countries, and We as a part of society is fully committed to support these efforts, wherever needed, and aspires to "Use the most advanced AI technology to serve the most fundamental needs."

1.4 PROJECT GOALS

- **System: Upload CT Scans** The system should be capable of getting Lung CT Scans from Users that will be utilized by the Deep Learning Model.
- **System: Detection of Lung Cancer** The system should be able to detect the Covid-19 within the Lung CT scan images that users have uploaded.
- **System: Display Results** The system should be able to give information that our user can appropriately understand and gain insight from it.

1.5 PROBLEM DEFINITION AND OBJECTIVES

- With minimum processing time to avoid the spread of Covid-19 virus in community
- To develop a system that detects Novel corona symptoms with maximum precision.
- To give most promising tool that can acceptable by all the doctors.
- To provide higher accuracy over previous research
- To give most promising tool that can acceptable by all the doctors.

1.6 PROJECT SCOPE AND LIMITATIONS

- Now day's Covid-19 are growing rapidly by busy and stress full life's.
- Patients with confirmed COVID-19 pneumonia have typical imaging features that can be helpful in early screening of highly suspected cases and in evaluation of the severity and extent of disease.
- The necessary tools required for early detection of these diseases are still not readily available in most populations globally.
- It is one the most dangerous disease among men and women and early identification and treatment is the best available option for the infected people.
- In future work we are going to invent such strong system which can accurately work on large medical dataset with number of diseases.
- Large number of training dataset is required for more accuracy

1.7 DEEP LEARNING INTEGRATION

Integrating deep learning models into applications using Python is different compared to standard machine learning algorithms. Standard machine learning algorithms can be serialized into a file and loaded into an application. Deep learning models use Tensorflow as a framework. Integrating the model requires additional code to ensure that the model runs sequentially on the same thread as the application. This is further explained in the implementation chapter.

1.8 ADVANTAGES

- To reduce death rate by early detection of Covid-19 in the world.
- To give unique solution for covid-19 diseases.
- To give most promising tool that can acceptable by all the doctors.

CHAPTER 2
LITERATURE SURVEY

2.1 BACKGROUND

A novel corona virus has resulted in person-to-person transmission but as far as we know, it causing corona virus disease 2019 (COVID-19) can also be from an asymptomatic carrier with no coved symptoms. Till now there is no report about any clinically approved antiviral medicine or vaccines that are effective against COVID-19. For NOVEL CORONA Patients CT scan have some symptoms such as, Reticulation, Ground Glass Opacities and Consolidation of lung tissue. To detect such symptoms is challenging. Traditional methods such as segmenting the Region of interest t, then extracting features of those part and then classify using some pretrained classifier is time consuming process. Recent studies on Deep learning change the process of detection by providing the image to this second generation neural networks which are capable of extracting features and classifying itself. Also CORONA is spreading fast, so this overall development should be completed within little duration. The Corona outbreak has put significant pressure on imaging departments, to test hundreds of peoples per day. Patients and doctors typically have to wait a few hours to get the CT results, but our system is improving the CT diagnosis speed for each case; and each minute saved is critical to decrease the chance of cross-contamination at the hospital. It has spread rapidly across the world, bringing massive health, economic, environmental and social challenges to the entire human population. At the moment, WHO recommends that people should wear face masks to avoid the risk of virus transmission and also recommends that a social distance of at least 2m be maintained between individuals to prevent person-to person spread of disease.

2.2 TABLE OF LITERATURE SURVEY

Sr.No	Paper Name	Paper Concept	Advantage	Disadvantage
1	Wei Zhao, ZhengZhong et al. **"Relation Between Chest CT Findings and Clinical Conditions of Corona virus Disease (COVID-19) Pneumonia: A Multicenter Study,"**	The purpose of this study was to investigate the relation between chest CT findings and the clinical conditions of COVID-19 pneumonia.	The health care industry generates a huge amount of data daily. However, most of it is not effectively used. Efficient tools to extract knowledge from these databases for clinical detection.	Covid diseases when aggravated spiral way beyond control. Skin diseases are complicated and take away lots from last year.
2	Wang W, Xu Y et al. **"Detection of SARS-CoV-2 in Different Types of Clinical Specimens. JAMA"**	This paper states that bio-distribution of SARS-CoV-2 among different tissues of inpatients with corona virus disease 2019 (COVID-19) diagnosed based on symptoms and radiology and	Early detection of Covid will help to recover the patient and it will not spread. This algorithm can discriminate	Time consuming due to use of SVM.

		confirmed by SARS-CoV-2 detection.	benign.	
3	Tao Ai, Zhenlu Yang et al. **"Correlation of Chest CT and RT-PCR Testing in Corona virus Disease 2019(COVID-19)"**	In this system experimented with diagnostic accuracy of CT using RT-PCR for SARS-CoV-2 as reference standard and investigated reasons for discordant results between the two tests.	This method employed deep neural network (DNN) gives better accuracy.	Lack in its performance when executed in local machines due to less amount of computational power.
4	Laura Matrajt, Tiffany Leung **"Evaluating the Effectiveness of Social Distancing Interventions to Delay or Flatten the Epidemic Curve of Coronavirus Disease"**	This System uses reduced contacts of adults >60 years of age, adults 20-59 years of age, and children <19 years of age for 6 weeks. Our results suggest interventions started earlier in the epidemic delay the epidemic curve and interventions started later flatten the epidemic curve.	Current literature usually handles this task through hand crafted feature creation and selection	Significantly increase accuracy Deep Learning method outperforms other c.
5	ShashiYadav "**Deep Learning based Safe Social**	Hence, this study proposes an automated method	A multi-layer perception neural	Manual segmentation is a time-

	Distancing and Face Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence"	that can identify dermatology and segment it across all image slices in volumetric images.	network is adopted as a classifier, and a bounding 3D-box-based genetic algorithm is used to identify the location of pathological tissues in the images.	consuming task highly prone to human error.

Table 1.LITERATURE SURVEY

2.3 Methodology for literature survey 1

Wei Zhao, ZhengZhong, XingzhiXie, Qizhi Yu and Jun Liu **"Relation Between Chest CT Findings and Clinical Conditions of Coronavirus Disease (COVID-19) Pneumonia: A Multicenter Study,"** American Journal of Roentgenology: 1-6. 10.2214/AJR.20.22976

Wei Zhao et.al [1] states that In December 2019, a series of cases of pneumonia of unknown causation emerged in Wuhan, Hubei, China, and quickly raised intense attention around the world [1]. A novel bat-origin coronavirus, 2019 novel corona-virus, was identified by means of deep sequencing analysis [2]. The virus, named severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [3], is phylogenetically closest to bat SARS-like coronavirus but in a separate clade, which means that a novel coronavirus is spreading [2]. As of February 17, 2020, 72,436 laboratory-confirmed cases were consecutively reported in 31 provinces (municipalities and regions) in China, including 11,741 severe cases, 1868 fatal cases,

and 6242 suspected cases [4}The study had several limitations. First, only 101 patients with confirmed COVID-19 were included; negative results and infections with other viruses were not included in the analyses. Comprehensive investigation of the imaging features of patients with negative results and other virus infections may help us to differentiate COVID-19 pneumonia from other lung infections and then to screen patients with highly suspected cases. Second, we did not evaluate follow-up CT findings in our study. Exploring the CT changes and comparing them with the clinical parameters may help us monitor and predict outcome and support clinical decision making.

Conclusion

Patients with confirmed COVID-19 pneumonia have typical imaging features that can be helpful in early screening of highly suspected cases and in evaluation of the severity and extent of disease. Most patients with COVID-19 pneumonia have GGO or mixed GGO and consolidation and vascular enlargement in the lesion. Lesions are more likely to have peripheral distribution and bilateral involvement and be lower lung predominant and multifocal. CT involvement score can help in evaluation of the severity and extent of the disease.

2.4 Methodology for literature survey 2

Wang W, Xu Y, Gao R, **"Detection of SARS-CoV-2 in Different Types of Clinical Specimens. JAMA"** Published online March 11, 2020. doi:10.1001/jama.2020.3786

Wang W et.al [2] proposed that in this examination, SARS-CoV-2 was recognized in examples from numerous destinations of 205 patients with COVID-19, with lower respiratory lot tests regularly testing positive for the infection. Significantly, the live infection was identified in defecation, inferring that SARS-CoV-2 might be sent by the fecal course. A little level of blood tests had positive PCR test results, proposing that contamination now and then might be fundamental. Transmission of the infection by respiratory and extra respiratory courses may help clarify the fast spread of illness. What's more, trying of examples from different destinations may improve the affectability and diminish bogus negative test outcomes. Two more modest

investigations detailed the presence of SARS-CoV-2 in butt-centric or oral swabs and blood from 16 patients in Hubei Province,3 and viral burden in throat swabs and sputum from 17 affirmed case.

Conclusion

Out of the sample collected 19% had severe illness. Most of the patients had a fever, dry cough, and fatigue. While for the patients on whom mRNA was collected in 7 patients' excreted virus in the respiratory tract. Here we conclude that early detection can help in stopping the spread

There were 1070 specimens collected from 205 patients with COVID-19 who were a mean age of 44 years (range, 5-67 years) and 68% male. Most of the patients presented with fever, dry cough, and fatigue; 19% of patients had severe illness. Bronchoalveolar lavage fluid specimens showed the highest positive rates (14 of 15; 93%), followed by sputum (72 of 104; 72%), nasal swabs (5 of 8; 63%), fibro bronchoscope brush biopsy (6 of 13; 46%), pharyngeal swabs (126 of 398; 32%), feces (44 of 153; 29%), and blood (3 of 307; 1%). None of the 72 urine specimens tested positive

2.5 Methodology for literature survey 3

Tao Ai, Zhenlu Yang, HongyanHou, Chenao Zhan, Chong Chen, WenzhiLv, Qian Tao, Ziyong Sun, Liming Xia ," **Correlation of Chest CT and RT-PCR Testing in Coronavirus Disease 2019 (COVID-19) in China: A Report of 1014 Cases**" Feb 26 2020https://doi.org/10.1148/radiol.2020200642

Tao Ai,et.al [3] introducing Key Results. The positive rates of reverse-transcription polymerase chain reaction (RT-PCR) assay and chest CT in our cohort were 59% (601 of 1014 patients) and 88% (888 of 1014 patients), respectively, for the diagnosis of patients suspected of having coronavirus disease 2019 (COVID-19) hest CT accuracy in diagnosing COVID-19 during the peak of the Italian epidemic: A retrospective correlation with RT-PCR testing. trategy of combining initial RT-PCR and chest CT was analysed to ... confirming COVID-19 infection, incorporating

multiple RT- Materials and Methods This study included 1014 patients in Wuhan, China, who underwent both chest CT and RT-PCR tests between January 6 and February 6, 2020. With use of RT-PCR as the reference standard, the performance of chest CT in the diagnosis of COVID-19 was assessed. In addition, for patients with multiple RT-PCR assays, the dynamic conversion of RT-PCR results (negative to positive, positive to negative) was analyzed as compared with serial chest CT scans for those with a time interval between RT-PCR tests of 4 days or more.

Conclusion

- Out of 1014 patients, 601 were Covid-19 positive 888 were CT scan+
- Out of 413 patients with negative RT-PCR results, 308 had positive chest CT findings. And out of that 308, 48% or 103 were highly likely positive cases.
- So we draw the following conclusion that chest CT scan has a high sensitivity for the diagnosis of covid-19 and should be used more often.
- Pregnant women infected with COVID-19 need to more intensive attention. Currently, there is limited data on COVID-19 with pregnancy and found mostly during late pregnancy which is risks for mother and fetus. During late pregnancy of COVID-19 infected mother may risk of adverse obstetrical outcomes. A multi-disciplinary team contact should be adopted for dealing with these patients as it allows effectively sharing knowledge and expertise as well as responsibility. However, repeated testing, contact tracing, and self-isolation may control the spread of SARS-CoV2 and COVID-19 infection until specific treatment either by vaccine or drugs are available.

2.6 Methodology for literature survey 4

Laura Matrajt, Tiffany Leung**"Evaluating the Effectiveness of Social Distancing Interventions to Delay or Flatten the Epidemic Curve of Coronavirus Disease"** pmc article doi: 10.3201/eid2608.201093. Epub 2020 Apr 28.

Tiffany Leungand LauraMatrajt [2] introducing that a numerical model to examine the adequacy of social separating mediations in a moderate sized city. Intercessions decreased contacts of grown-ups >60 years old, grown-ups 20-59 years

old, and youngsters <19 years old for about a month and a half. Our outcomes recommend mediations began before in the plague defer the pandemic bend and intercessions began later straighten the pestilence bend. We noticed that, while social removing mediations were set up, most new cases, hospitalizations, and passing's were deflected, even with humble decreases in contact among grown-ups. Nonetheless, when intercessions finished, the pestilence bounced back. Our models propose that social removing can give critical opportunity to expand medical care limit however should happen related to testing and contact following of all speculated cases to alleviate infection transmission.

2.7 Methodology for literature survey 5

ShashiYadav"**Deep Learning based Safe Social Distancing andFace Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence"** International Journal for Research in Applied Science & Engineering Technology (IJRASET) 2020

ShashiYadavet.al[5] introducing to establish safe climate that adds to public security, we propose a productive PC vision put together methodology centered with respect to the continuous mechanized observing of individuals to recognize both safe social removing and face veils in broad daylight puts by executing the model on raspberry pi4 to screen action and identify infringement through camera. After identification of penetrate, the raspberry pi4 imparts ready sign to control focus at state police base camp and furthermore offer alert to public. In this proposed framework current profound learning calculation have been blended in with mathematical methods for building a vigorous modular which covers three parts of location, following, and approval. Accordingly, the proposed framework favors the general public by saving time and helps in bringing down the spread of Covid. It tends to be actualized adequately in current circumstance when lockdown is facilitated to investigate people in open social events, shopping centers, and so on Robotized investigation decreases labor to assess people in general and furthermore can be utilized in any place.

Conclusion

In this paper, we proposed an approach that uses computer vision and MobileNet V2 architecture to help maintain a secure environment and ensure individuals protection by automatically monitoring public places to avoid the spread of the COVID-19 virus and assist police by minimizing their physical surveillance work in containment zones and public areas where surveillance is required by means of camera feeds with raspberry pi4 in real-time.

Thus, this proposed system will operate in an efficient manner in the current situation when the lockout is eased and helps to track public places easily in an automated manner. We have addressed in depth the tracking of social distancing and the identification of face masks that help to ensure human health. The implementation of this solution was successfully tested in real-time by deploying model in raspberry pi4. The solution has the potential to significantly reduce violations by real-time interventions, so the proposed system would improve public safety through saving time and helping to reduce the spread of coronavirus. This solution can be used in places like temples, shopping complex, metro stations, airports, etc.

CHAPTER 3
SOFTWARE REQUIREMENTS SPECIFICATION

3.1 ASSSUMTIONS AND DEPENDENCIES

Assumptions:-

Reliable identification and classification of covid-19 requires pathological test, namely, needle biopsy specimen and analysis by experienced pathologists as it involves human judgment of several factors and a combination of experiences, a decision support system is desirable in this case. Existing system does not provide diagnosis system which helps to doctors.

Dependencies:-

Algorithm is using structured and unstructured data from hospital. To the best of our knowledge, none of the existing work focused on both data types in the area of medical big data analytics. It is reducing death rate by wrong diagnosis using giving accurate diagnosis.

3.2. FUNCTIONAL REQUIREMENTS

3.2.1 System Feature 1(Functional Requirement)

- Covid-19 detection according to CT scan Images

3.2.2 System Feature2 (Functional Requirement)

- Database -SQLite3

3.3 EXTERNAL INTERFACE REQUIRMENTS

3.3.1 User Interfaces

Python

Python interface is being actively developed right now. There are many algorithms and many functions that compose or support those algorithms. Open CV is written natively in C++ and has a template interface that works seamlessly with STL containers.

Image Processing

Read and Write Images. Detection of images and its features. Detection of shapes like Circle, rectangle etc in a image, Detection of coin in images. Text recognition in images.e.g. Reading Number Plates.Modifying image quality and colors.

3.3.2 Hardware Interfaces

To run our project we required a hardware system which is feasible for our project like Intel I3 processor, 2 GB RAM, 20GB Hard disk .We also need standard keyboard, Mouse, LED Monitor.

3.3.3 Software Interfaces

The system can use Microsoft as the operating system platform. System also makes use of certain GUI tools. To run this application we need JDK and above as python platform. To store data we need SQLite3 database.

3.3.4 Communication Interfaces

1. Disease detection System
2. User disease image data set
3. Pre-processing unit
4. Feature vector generation using CNN
5. Classified results in the form of predictions
6. Open-CV for image processing

3.4 NONFUNCTIONAL REQUIREMENTS

3.4.1 Performance Requirements

- **Performance –**
  Performance of our system fast as compare other system and response time is quick.
- **Availability-**
  Availability of data is also requirement for performing any operations.

- **Maintainability-**
  In this system we can maintain data of user's images.
- **Security-**
  In this system user information is store in the form of images, so our system is secure.
- **Usability-**
  This system is very useful in assistive tool for medical sector.

3.4.2 Safety Requirements

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available.

3.4.3 Security Requirements

No security needed in the system due to in this system user information is store in the form of images, so our system is secure.

3.4.4 Software Quality Attributes

- **Capacity**
  Capacity of project according to data it is very less.
- **Availability**
  Proposed system will available on python application.
- **Reliability**
  System is reliable for disease prediction.
- **Security**
  User when login to system that time users mail id and password match accurately.

3.5 SYSTEM REQUIREMENTS

3.5.1 Database Requirements

- Database – SQLite3

3.5.2 Software Requirements (Platform Choice)

- Operating System - Windows 7 or above , Mac OSX or above
- Programming Language - Python 3.7
- Software Version - Python IDLE 3.7
- GUI - PyQT4, tkinter
- IDE - Py-Charm or any compatible
- Python Libraries - Tensorflow 2.2, Keras, Matplotlib, Reportlab, OpenCV, Scikit-learn, db-sqlite3 etc,

3.5.3 Hardware Requirements

- Processor - Inter Core I3 / AMD Rizen 3 or above
- CPU Speed - Minimum 2 GHz, Recommended 3GHz or more.
- Memory(RAM) - Minimum2 GB, Recommended 4 GB or more
- Hard Drive - Minimum 20 GB, Recommended 40 GB or more
- Key Board - Standard Windows / Mac Keyboard
- Mouse - Standard Mouse
- Monitor - LED, LCD Monitor

3.6 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED

3.6.1 Waterfall Model –

The waterfall model is a sequential model that is used in the software development processes, where the process is seen flowing steadily downwards through the phase of Requirement Gathering and Analysis, System Design, Implementation, Testing, Deployment and Maintenance.

3.6.1.1. Requirement analysis:

Here requirements are gathered means which kind of dataset is required. Then what are functional requirement of system. Document is prepared, and then use cases are designed. In our system we gather all information of Admin and user and functionality of each module.

3.6.1.2. System Design:

In this stage, hardware and software requirement to design the system is decided. It uses above mentioned hardware and software requirements. We design the of Admin and user module. Design the according to functionality of each module.

3.6.1.3. Implementation:

In this stage, system is developed module wise. In this system consist of mainly 2 modules that is

1. Admin 2. User

3.6.1.4. Testing:

In this stage, all developed software's are installed and they are tested in different ways against the system requirements. In this stage we check all this module is working properly or not with proper authentication. Disease prediction proper or not as well as stage prediction proper or not.

3.6.1.5. Deployment:

In this deployment stage we deployed the new functionality of each module like Crop Yields Predictions, Crop Suggestion to farmer; Dynamic Assistance and Online E-Mart modules .We deploy all system with proper functions.

3.6.1.6. Maintenance:

According to software's new version and their use, they need to be updated. , some predefined machine learning libraries need to be used. This system is easy to maintain

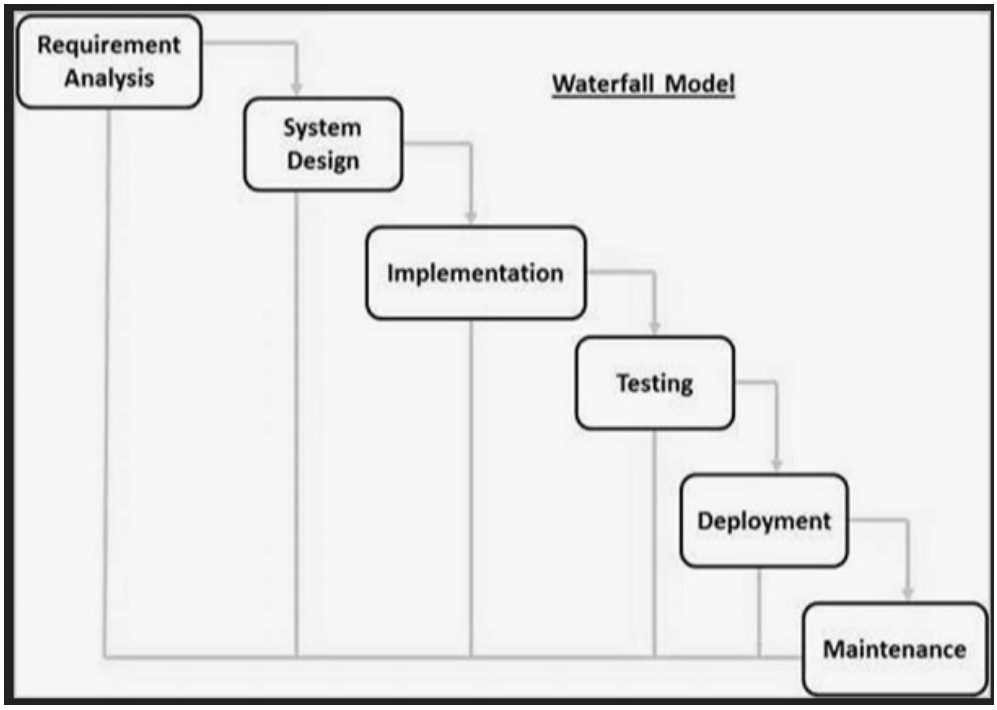


Fig 3.1.Waterfall module

CHAPTER 4
SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

4.1.1 Proposed System Architecture

In a proposed system, we are proposing experiment on detection of covid-19 disease using lungs of CT scan images with limited set of supervised data.

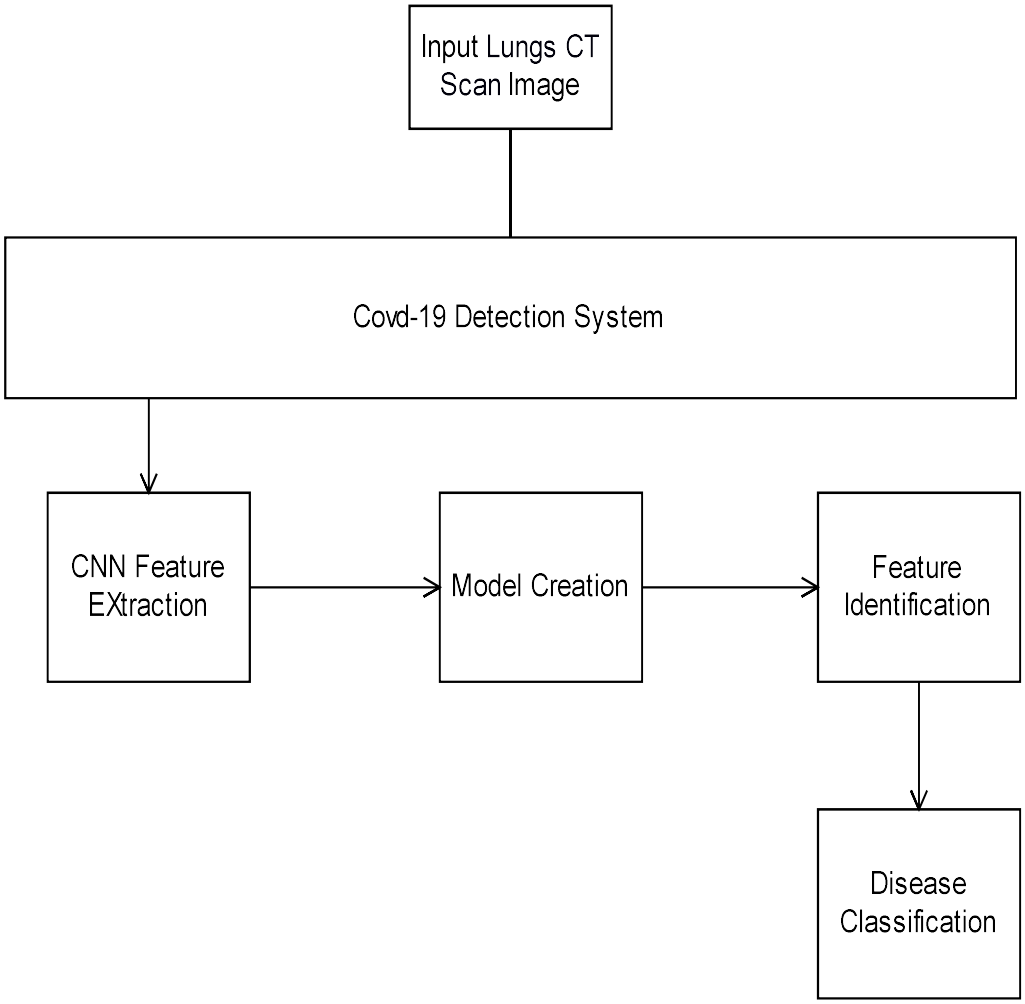


Fig 4.1 Block Diagram of Proposed System

We propose a new Convolutional neural network based multimodal disease risk prediction model for limited covid-19 with higher accuracy. We are going to solve accuracy issue in diagnosis of Covid-19 with accurate stage predictions. We also work on detections by machine evaluations depends on sizes in mm. Admin and users

are two modules include in our system. Admin first gather the information about diseases in the form of images. After gathering of information like preprocessing on the data, training of the data, model generation according to the features of the data. User inserts the image for checking Covid-19 diseases. Using our proposed system, we predict the disease according to different type of stages. We are trying to develop system for multi disease detection and stages predictions gives early detection and saves lots of life's by reducing death rate by covid-19 diseases.

- We are going to invent system which will help to medical research and diagnosis tool.
- Our work is based on machine learning techniques for disease detection framework.
- Modules:
  1. Covid-19 Detection
- Features
  Covid-19 detections according to Lungs of CT scan Images

4.1.2 System Analysis Proposed Architecture

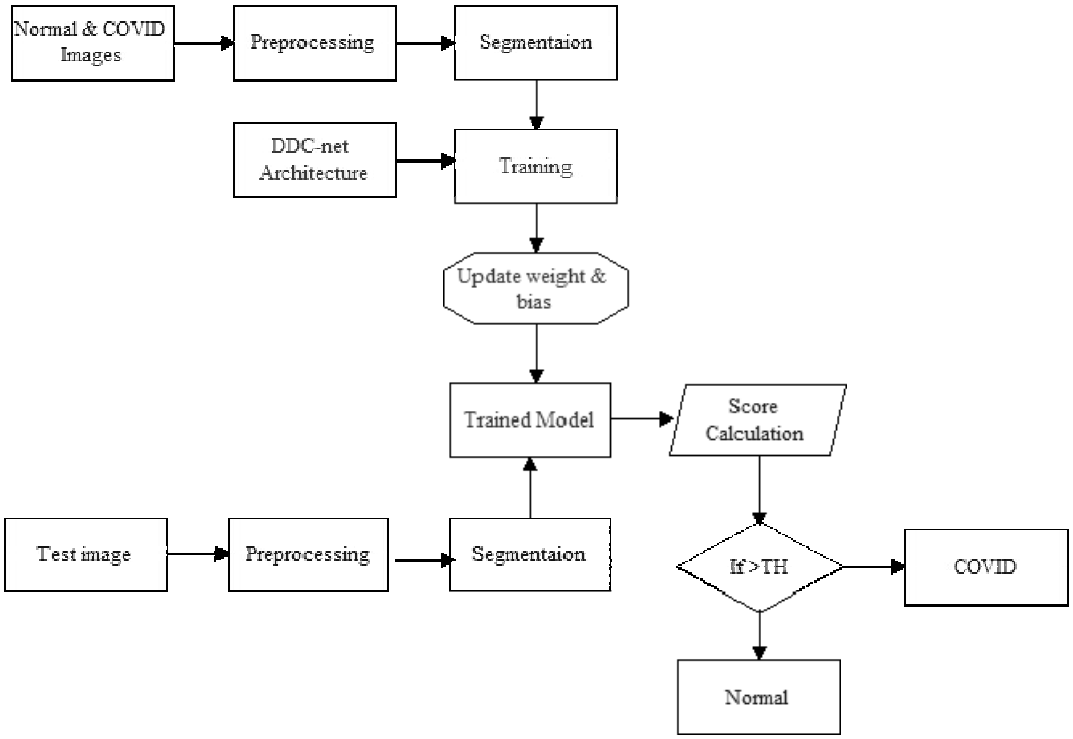


Fig 4.2 System Analysis Proposed Architecture

Proposed method takes Lung CT scan as input. It process on input image using median filter. After that it extracts the region of interest. Then our deep dense network will look for any symptoms for corona such as glass opacity. If it found any of the trained symptom then it will gives result for COVID costiveness. The accuracy of any Deep Network depends on the training dataset. For our model we used normal Lung CT scan from LIDC Dataset and Corona image are taken from web. As there is privacy issues of corona images. Also in this situation no one is ready to make those dataset public. A new artificial intelligence-powered deep learning model will help radiologists to distinguish COVID-19 from community-acquired pneumonia and other lung diseases in chest CT imaging. Proposed model will help with a growing workload to be able to focus on complex clinical cases. With its deep learning algorithms, it automatically highlights abnormalities, segments anatomies. Proposed model gives better accuracy for Dataset. For real time imagery large dataset is needed.

4.2 DATA FLOW DIAGRAM

A **data flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about process timing or whether processes will operate in sequence or in parallel, unlike a traditional structured flowchart which focuses on control flow, or a UML activity workflow diagram, which presents both control and data, flows as a unified model.

4.3.1 DFD LEVEL 0

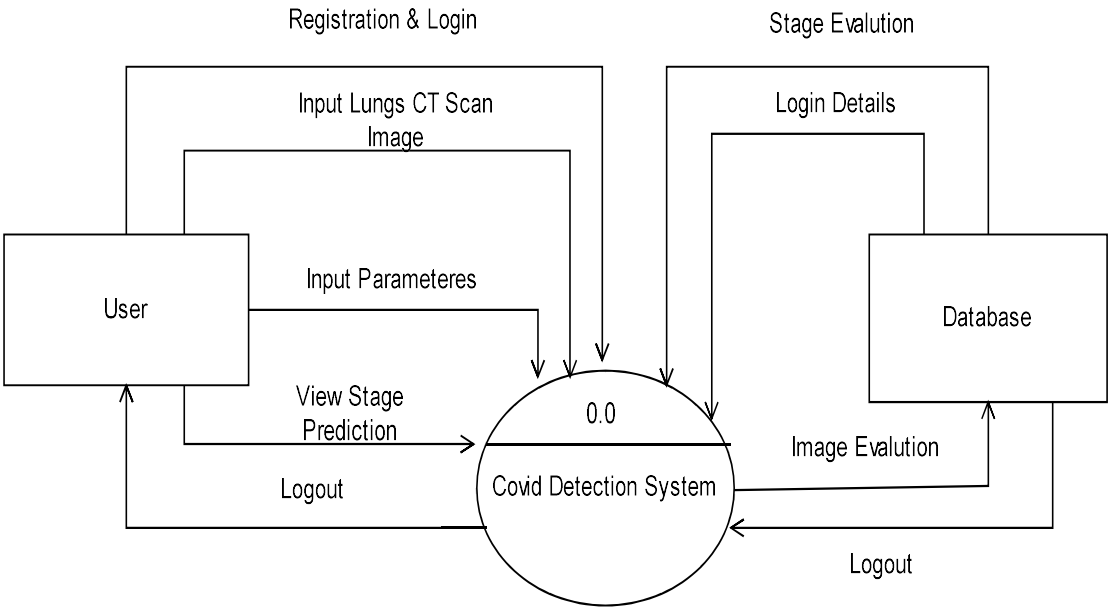


Fig 4.3 DFD level 0

4.3.2 DFD LEVEL 1

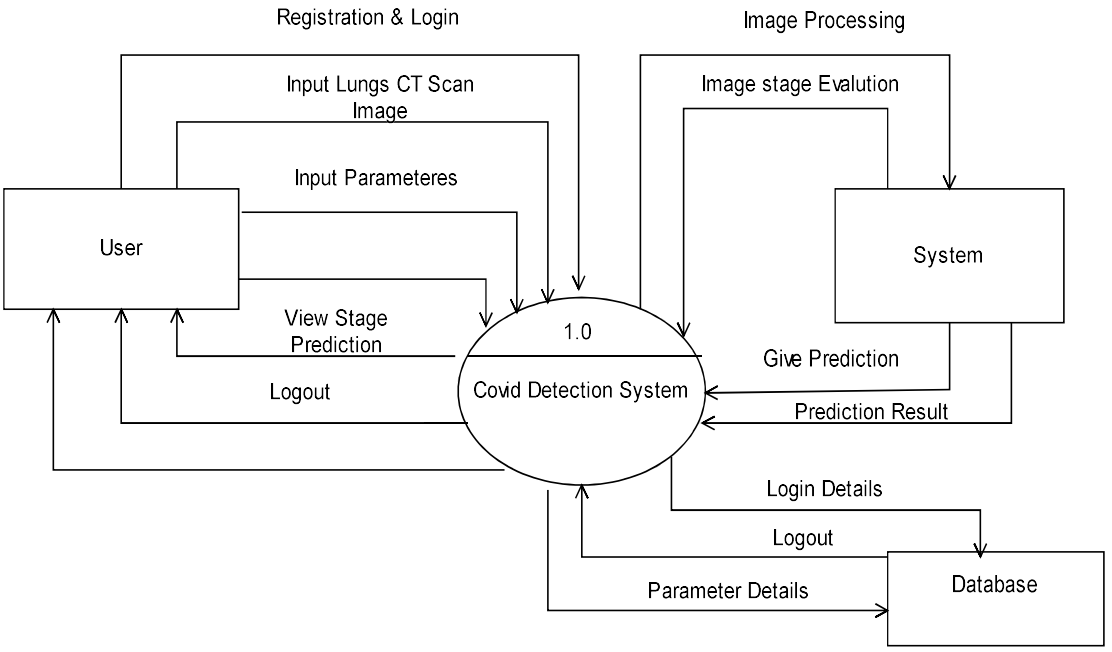


Fig 4.4 DFD 1 level

4.3 ENTITY RELATIONSHIP DIAGRAMS

An **entity–relationship model (ER model for short)** describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

In software engineering, an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure which can be implemented in a database, typically a relational database.

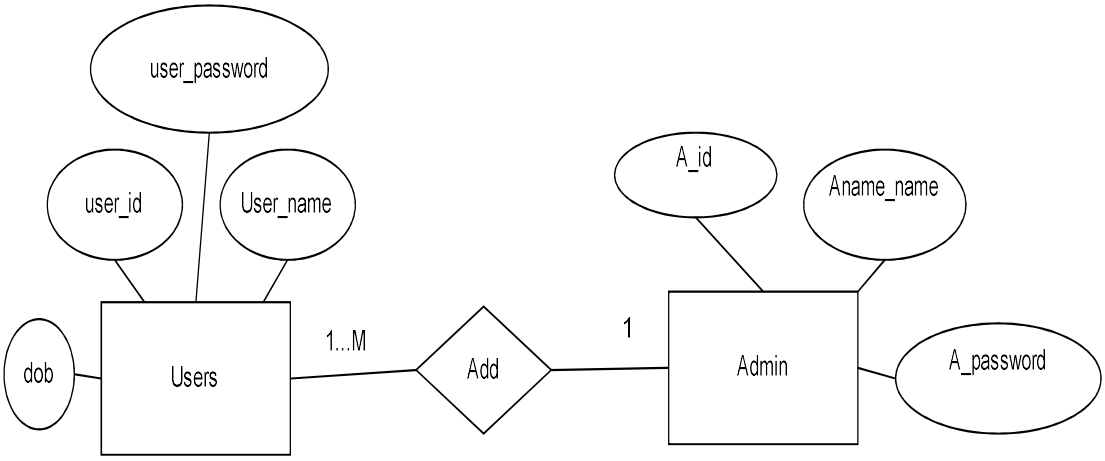


Fig 4.5 ER- Diagram

4.4 UML DIAGRAMS

4.4.1 Use case diagram:-

A use case diagram is a graphical representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can show the different types of users of a system and the various ways in which they interact with the system. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to gather its functionality use cases are prepared and actors are identified. The purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interaction among the actors.
-

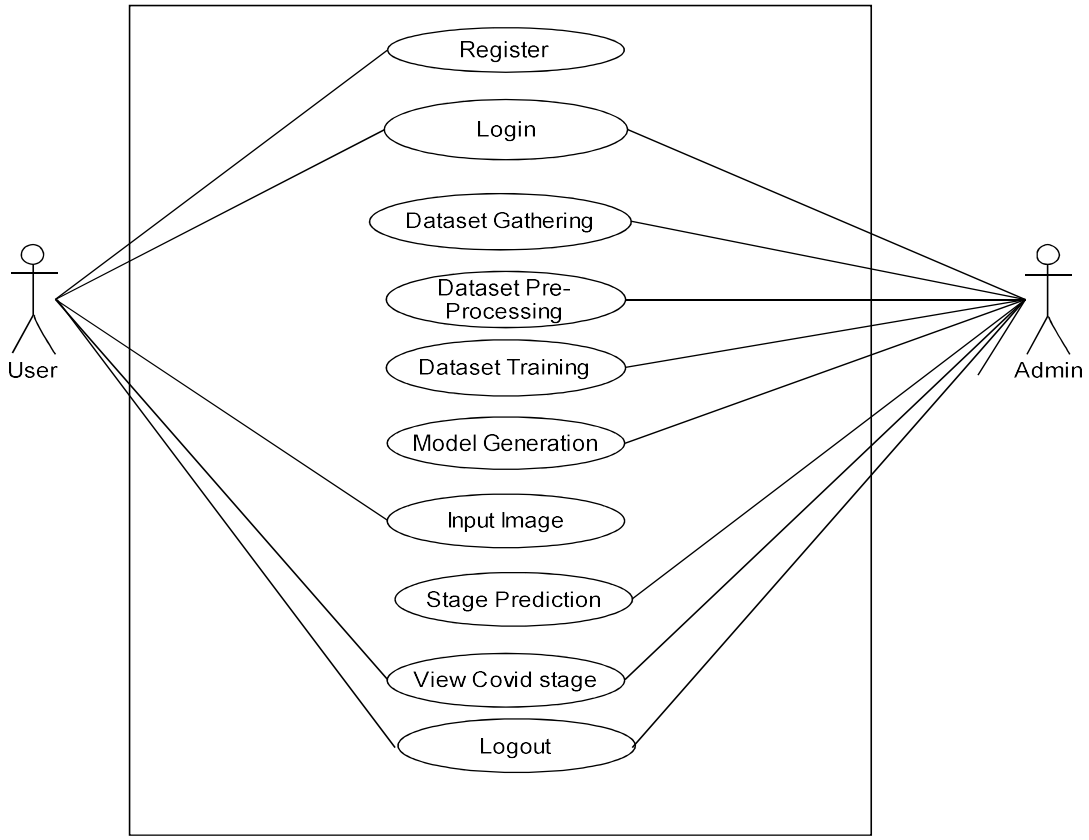


Fig 4.6 Use case Diagram.

4.4.2 Activity diagrams

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control. Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shape types:

- Rounded rectangles represent actions;
- Diamonds represent decisions;

- Bars represent the start (split) or end (join) of concurrent activities;
- A black circle represents the start (initial state) of the workflow;
- An encircled black circle represents the end (final state).

Arrows run from the start towards the end and represent the order in which activities happen. Hence they can be regarded as a form of flowchart. Typical flowchart techniques lack constructs for ex-pressing concurrency. However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.

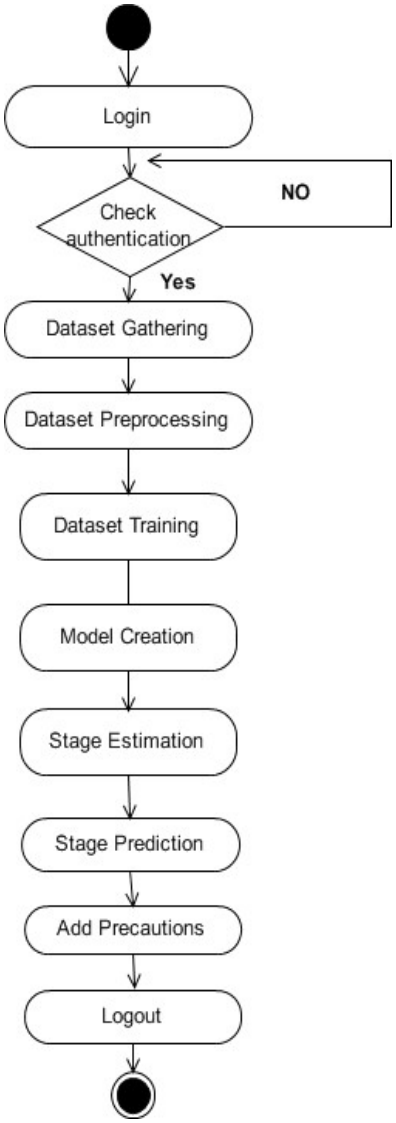


Fig 4.7 Activity Diagram.

4.4.3 Class Diagram

The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a structural diagram. The purpose of the class diagram is to model the static view of an application.

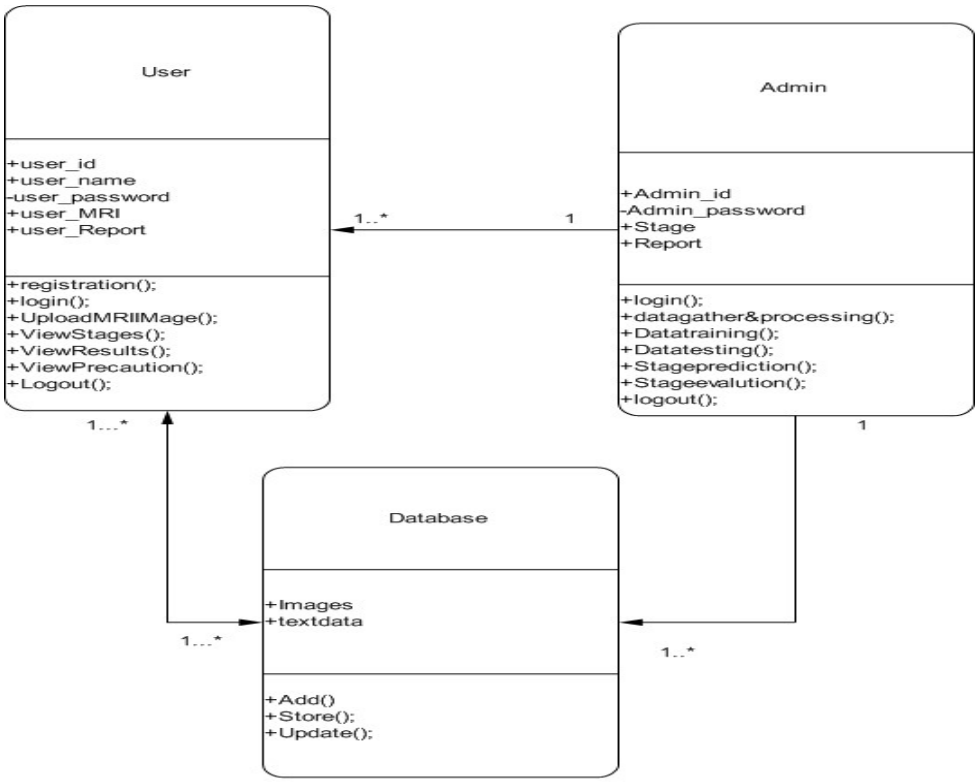


Fig 4.8 Class Diagram

4.4.4 Sequence Diagram

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

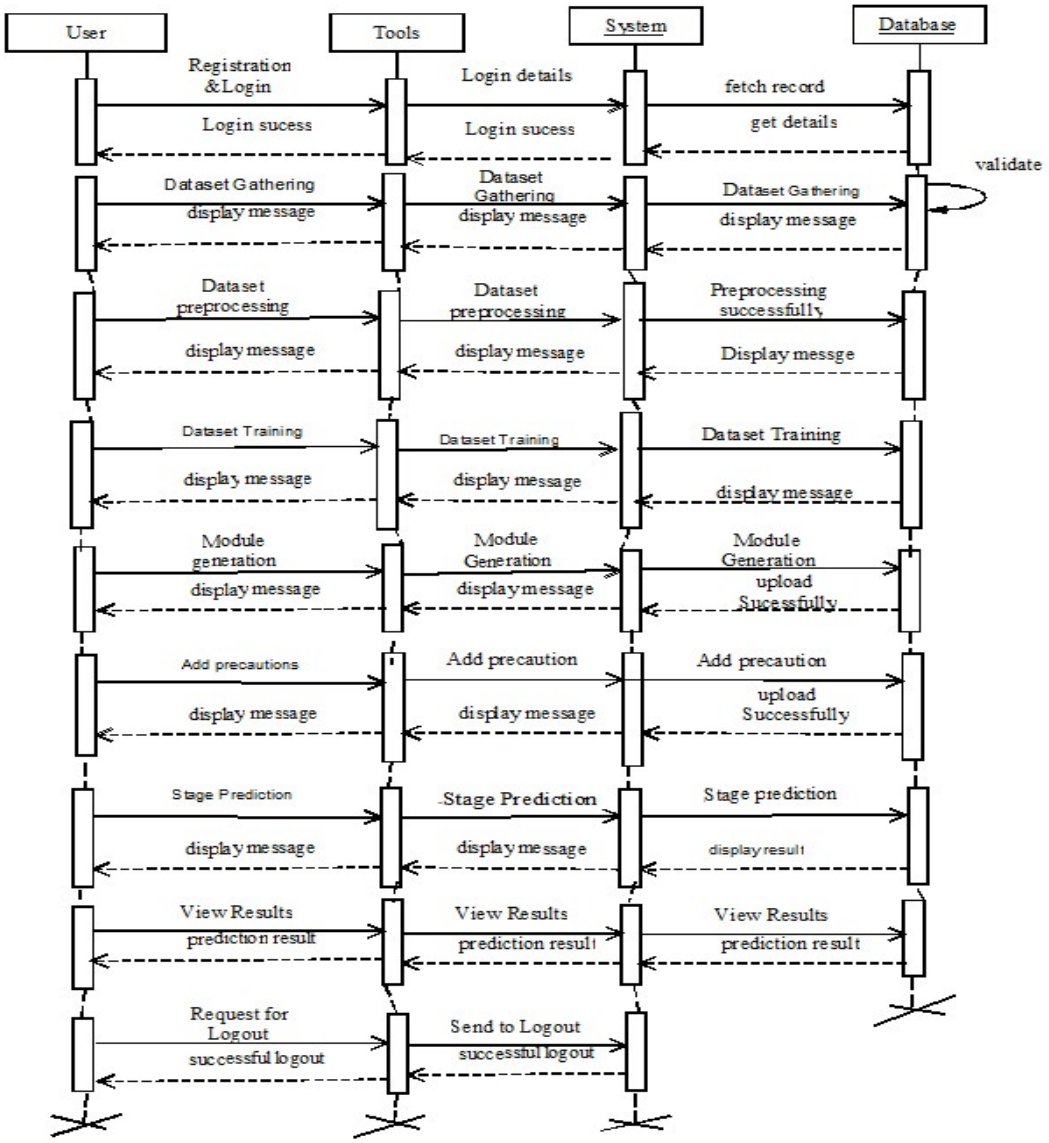


Fig 4.9 Sequence diagram.

4.4.5 Component Diagram

A Component Diagram displays the structural relationship of components of a software system. These are mostly used when working with complex systems that have many components. Components communicate with each other using interfaces. The interfaces are linked using connectors.

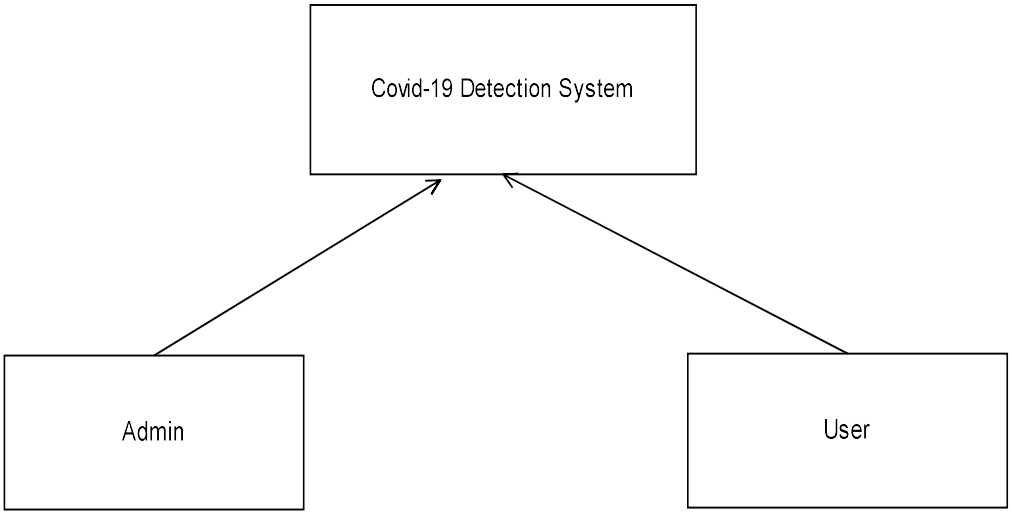


Fig 4.10 Component Diagram

4.4.6 Deployment Diagram:

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed. So deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

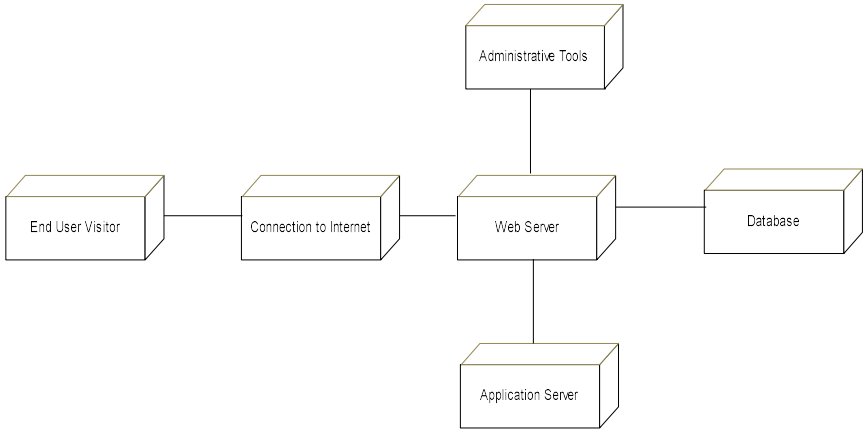


Fig 4.11 Deployment Diagram

4.4.7 State Diagram

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction.

User State Diagram

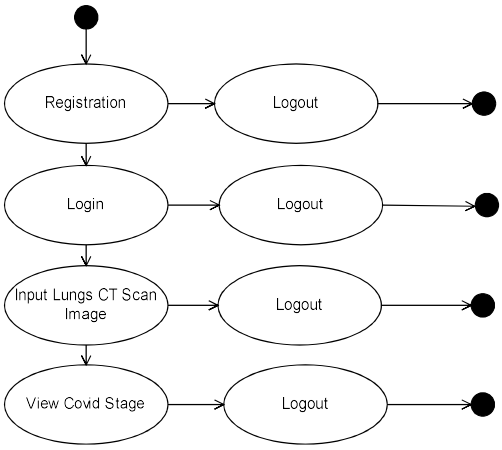


Fig 4.12 User State Diagram

Admin Stage Diagram

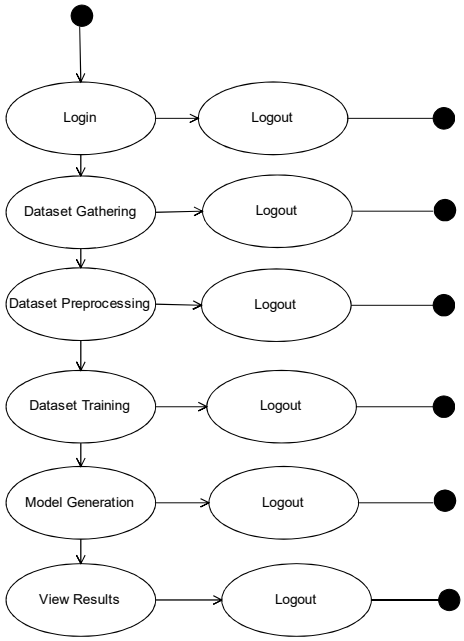


Fig 4.13 Admin State Diagram

4.5 USER INTERFACE DESIGN

4.5.1 Registration Form Page

In this page is to allow a user to register the details such as user name, user email, user number, and password after the user has registered by pressing “CLICK TO REGISTER” then the user can go to login page by pressing “CLICK TO LOGIN PAGE”.

Fig 4.14 Registration Form Page

4.5.2 Login Form Page

In this page is to allow a user to enter the user name and password if the details are matched with the database then the user can only go to the system by pressing “LOGIN TO SYSTEM”.

Fig 4.15 Login Form Page

4.5.3 Covid-19 Test Page

In this page is to allow a user to input the a Lung CT scan image by pressing "INPUT CT-SCAN IMAGE" then the system will display the result asfollowed in the Fig below.

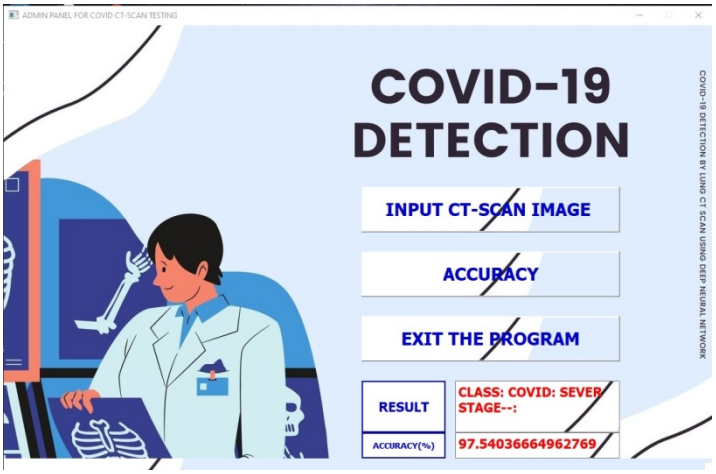


Fig 4.16 Covid-19 Test Page

4.5.4 User Report in PDF

After a user has finished the test then the result will save in the form of report in PDF format

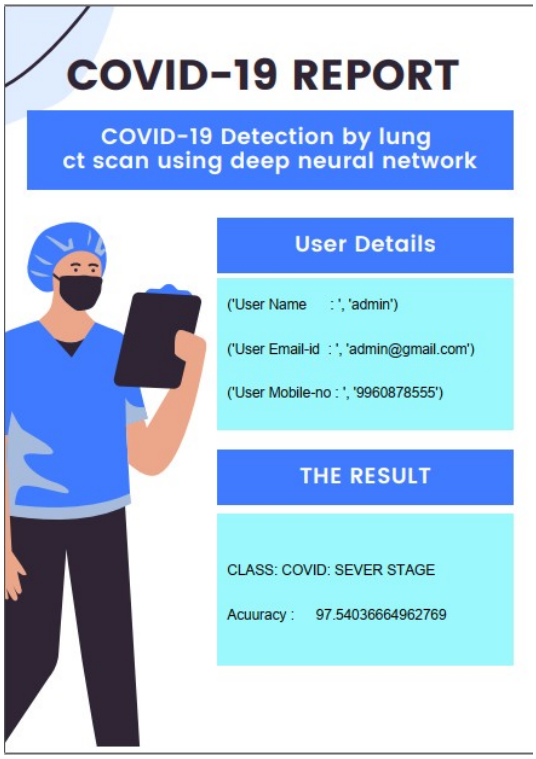


Fig 4.17 Covid-19 Test Page

CHAPTER 5

PROJECT PLAN

5.1 SYSTEM IMPLEMENTATION PLAN

• Schedule		Date	Project Activity
August	3rd Week	12/08/2020	Formation Of Project Group
	4th Week	18/08/2020	Project Topic Selection
September	4th Week	24/09/2020	Presentation On Project Ideas
	5th Week	28/09/2020	Submission Of Literature Survey
	5th Week	30/09/2020	Feasibility Assessment
October	3rd Week	16/10/2020	Design Of Mathematical Model
November	1st Week	07/11/2020	Report Preparation And Submission
January	4th Week	19/01/2020	1st module presentation
	5th Week	26/01/2020	Discussion and implementation of 2nd module
February	2nd Week	09/02/2021	Study of implementation of user module
	3rd Week	16/02/2021	Discussion about modification to Improve in of user result module.
	4th Week	26/02/2021	1st and 2nd module presentation
March	1st Week	04/03/2021	Discussion on flow of project and designing new module
	2nd Week	08/03/2021	Modification of modules.
	3rd Week	20/03/2021	Designed test cases for our module.
	5th Week	28/03/2021	Worked on user interface.
April	2nd Week	06/04/2021	Integration of all modules phase I.
	2nd Week	10/04/2021	Modification of modules.
		20/04/2021	Integration of all modules phase II
May	2nd Week	06/05/2021	Report and presentation for Sem-I
	3rd Week	13/05/2021	Modification of modules.
	5th Week	29/05/2021	Integration of all modules final
June	2nd Week	10/06/2021	Test all modules and prepare Final report
	3rd Week	17/06/2021	Final Report and presentation for Sem-II

Table 2. Project plan

CHAPTER 6
PROJECT IMPLEMENTATION

6.1 OVERVIEW OF PROJECT MODULES

Proposed method takes Lung CT scan as input. It process on input image using median filter. After that it extracts the region of interest. Then our deep dense network will look for any symptoms for corona such as glass opacity. If it found any of the trained symptom then it will gives result for COVID costiveness. The accuracy of any Deep Network depends on the training dataset. For our model we used normal Lung CT scan from LIDC Dataset and Corona image are taken from web. As there is privacy issues of corona images. Also in this situation no one is ready to make those dataset public. In Second generation, number of architectures or algorithms is present for classification problem. In other languages we have to start from scratch, but for MATLAB and Python this is another case., simply calling those function and changing the input argument, you test.

System Modules:

- Input train data
- Preprocessing
- Segmentation
- Update weight and bias
- Score Calculation
- Prediction
- Prediction result

User Modules:

- Input test data
- Prediction
- Prediction result

6.2 TOOLS AND TECHNOLOGIES USED

Python

Python is a high level interpreted language used for general purpose programming. It is widely used for scientific computing and can be used for a wide variety of general tasks from data mining to software development. Python is the main language used for this project.

Tensorflow

Tensorflow is an open source deep learning library by Google. It was originally developed by Google's engineers who were working on Google Brain and has been used for research on machine learning and deep learning. Tensorflow at its core is about computations of multidimensional arrays called tensors but what makes Tensorflow great is its ability to be flexible to deploy computations on different devices such as CPU's and GPU's

OpenCV

OpenCV (Open Source Computer Vision) is a well-established computer vision library which is written in C/C++ and has been abstracted to interface with C++, Python and Java. This is a powerful tool when working with images and has a myriad of tools regarding image data manipulation, feature extraction and etc.

Matplotlib

Matplotlib is a Python plotting library that allows programmers to create a wide variety of graphs and visualizations with ease of use. The great feature about Matplotlib is that it integrates very well with Jupyter Notebook and creating visualizations is simplified. Matplotlib also works very well with pandas and numpy.

Numpy

Numpy is a library in Python that allows for efficient numerical computing in Python. This library is highlyoptimized to do mathematical tasks. In the project workflow Numpy is heavily used in data pre-processingand preparation One of

the main features about Numpy is it's highly efficient n-dimensional array (ndarray). Compared to a list in Python a Numpy array can be n-dimensions and has more features associated with thendarray. Numpy can also perform more efficient mathematical operations compared to the math library inPython.

Keras

Keras is also a Deep Learning Framework that abstracts much of the code in the other Frameworks like Tensorflow and Theano. Compared to the other frameworks Keras is more minimalist

6.3 ALGORITHM DETAILS

6.3.1 Convolutional Neural Networks

A breakthrough in building models for Covid-19 Lung CT Images classification came with the discovery that a convolutional neural network (CNN) could be used to progressively extract higher- and higher-level representations of the Lung image content. Instead of preprocessing the CT image to derive features like textures and shapes, a CNN takes just the CT image's raw pixel data as input and "learns" how to extract these features, and ultimately infer what object they constitute.

To start, the CNN receives an input feature map: a three-dimensional matrix where the size of the first two dimensions corresponds to the length and width of the images in pixels. The size of the third dimension is 3 (corresponding to the 3 channels of a CT image: red, green, and blue) though it is gray(By replicating same channel three times). The CNN comprises a stack of modules, each of which performs three operations.

- **Convolution**

  A convolution extracts tiles of the input feature map, and applies filters to them to compute new features, producing an output feature map,

or convolved feature (which may have a different size and depth than the input feature map). Convolutions are defined by two parameters:

- Size of the tiles that are extracted (typically 3x3 or 5x5 pixels).
- The depth of the output feature map, which corresponds to the number of filters that are applied

During a convolution, the filters (matrices the same size as the tile size) effectively slide over the input feature map's grid horizontally and vertically, one pixel at a time, extracting each corresponding tile

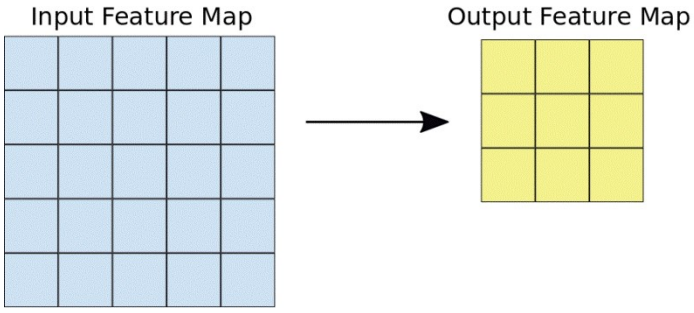


Fig 6.1 A3x3 convolution of depth
1 performed over a 5x5 input feature map, also of depth

There are nine possible 3x3 locations to extract tiles from the 5x5 feature map, so this convolution produces a 3x3 output feature map.

In Figure 3, the output feature map (3x3) is smaller than the input feature map (5x5). If you instead want the output feature map to have the same dimensions as the input feature map, you can add padding (blank rows/columns with all-zero values) to each side of the input feature map, producing a 7x7 matrix with 5x5 possible locations to extract a 3x3 tile.

For each filter-tile pair, the CNN performs element-wise multiplication of the filter matrix and the tile matrix, and then sums all the elements of the resulting matrix to get a single value. Each of these resulting values for every filter-tile pair is then output in the convolved feature matrix (see Figures 4a and 4b).

Input Feature Map

3	5	2	8	1
9	7	5	4	3
2	0	6	1	6
6	3	7	9	2
1	4	9	5	1

Convolutional Filter

1	0	0
1	1	0
0	0	1

Fig 6.2 Left: A 5x5 input feature map (depth 1).
Right: a 3x3 convolution (depth 1)

During training, the CNN "learns" the optimal values for the filter matrices that enable it to extract meaningful features (textures, edges, shapes) from the input feature map. As the number of filters (output feature map depth) applied to the input increases, so does the number of features the CNN can extract. However, the tradeoff is that filters compose the majority of resources expended by the CNN, so training time also increases as more filters are added. Additionally, each filter added to the network provides less incremental value than the previous one, so engineers aim to construct networks that use the minimum number of filters needed to extract the features necessary for accurate CT images classification.

- **Fully Connected Layers**

At the end of a convolutional neural network are one or more fully connected layers (when two layers are "fully connected," every node in the first layer is connected to every node in the second layer). Their job is to perform classification based on the features extracted by the convolutions. Typically, the final fully connected layer contains a softmax activation function, which outputs a probability value from 0 to 1 for each of the classification labels the model is trying to predict.

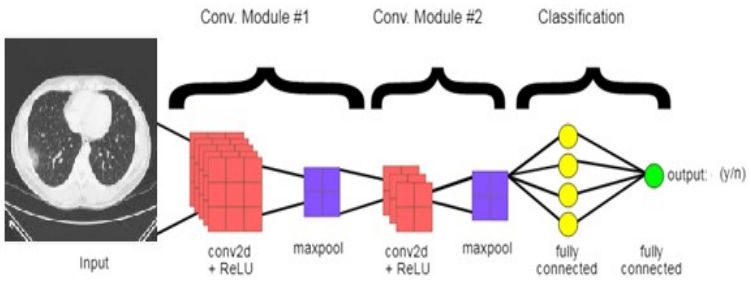


Fig 6.3 illustrates the end-to-end
Structure of a convolutional neural network.

6.3.2 ReLU

Following each convolution operation, the CNN applies a Rectified Linear Unit (ReLU) transformation to the convolved feature, in order to introduce nonlinearity into the model. The ReLU function, F(x)=max(0,x), returns x for all values of x > 0, and returns 0 for all values of x ≤ 0.ReLU is used as an activation function in a variety of neural networks;

6.3.3 Pooling

After ReLU comes a pooling step, in which the CNN downsamples the convolved feature (to save on processing time), reducing the number of dimensions of the feature map, while still preserving the most critical feature information. A common algorithm used for this process is called max pooling.

Max pooling operates in a similar fashion to convolution. We slide over the feature map and extract tiles of a specified size. For each tile, the maximum value is output to a new feature map, and all other values are discarded. Max pooling operations take two parameters:

Size of the max-pooling filter (typically 2x2 pixels)

Stride: the distance, in pixels, separating each extracted tile. Unlike with convolution, where filters slide over the feature map pixel by pixel, in max pooling, the stride determines the locations where each tile is extracted. For a 2x2 filter, a stride of 2 specifies that the max pooling operation will extract all nooverlapping 2x2 tiles from the feature map (see Figure below).

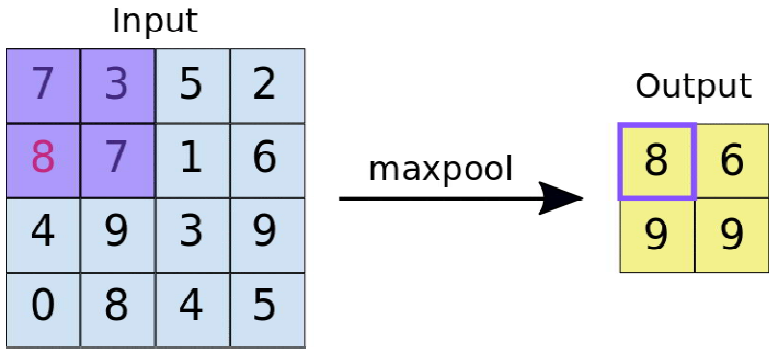
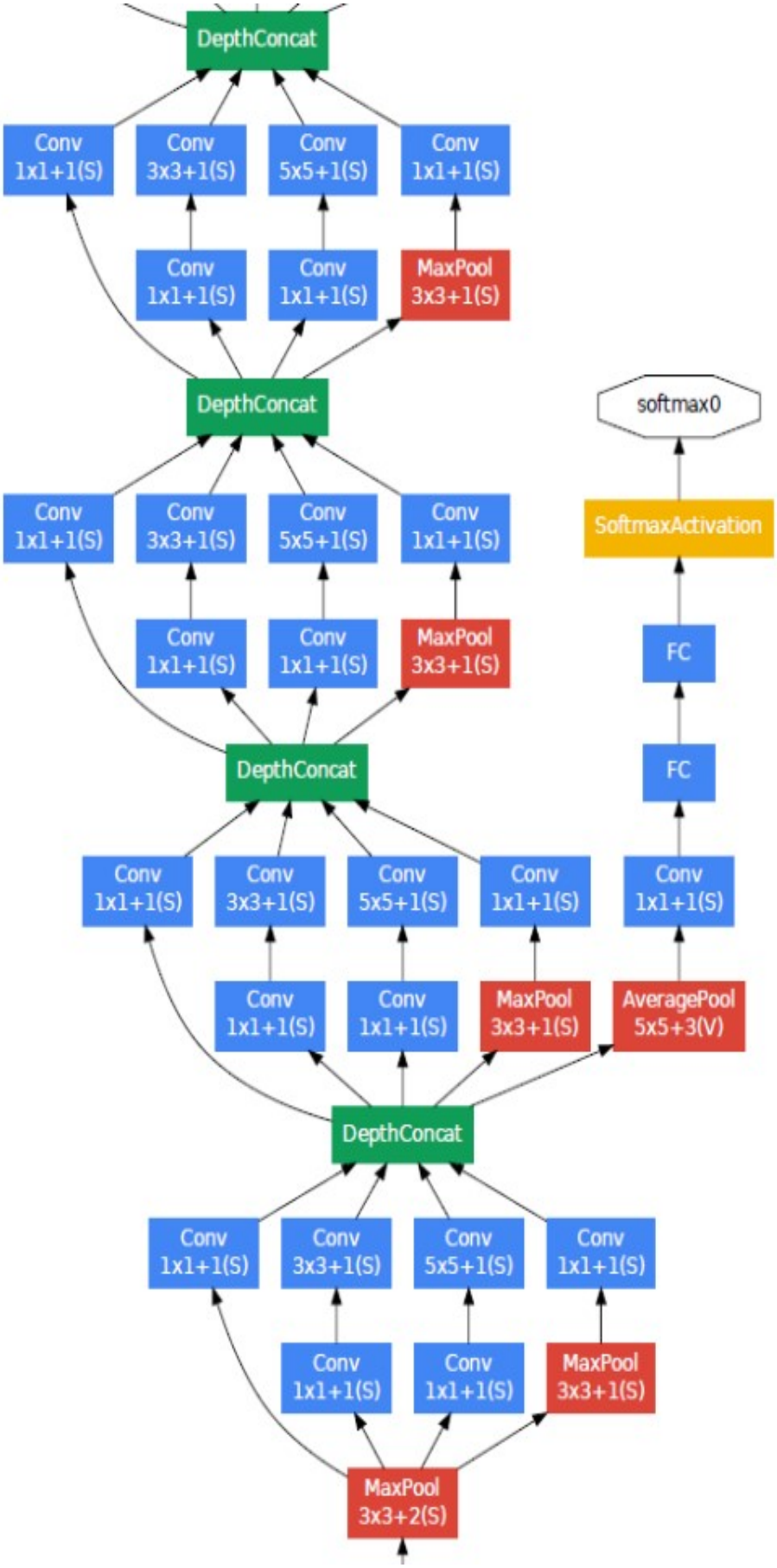


Fig 6.4 Left: Max pooling performed over a 4x4 feature map with a 2x2 filter and stride of 2. Right: the output of the max pooling operation. Note the resulting feature map is now 2x2, preserving only the maximum values from each tile.

6.3.4 GoogLeNetArchitecture Layers and Parameters


DepthConcat
Conv 1x1+1(S)
Conv 3x3+1(S)
Conv 5x5+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
MaxPool 3x3+1(S)
DepthConcat
softmax0
Conv 1x1+1(S)
Conv 3x3+1(S)
Conv 5x5+1(S)
Conv 1x1+1(S)
SoftmaxActivation
Conv 1x1+1(S)
Conv 1x1+1(S)
MaxPool 3x3+1(S)
FC
DepthConcat
FC
Conv 1x1+1(S)
Conv 3x3+1(S)
Conv 5x5+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
MaxPool 3x3+1(S)
AveragePool 5x5+3(V)
DepthConcat
Conv 1x1+1(S)
Conv 3x3+1(S)
Conv 5x5+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
Conv 1x1+1(S)
MaxPool 3x3+1(S)
MaxPool 3x3+2(S)

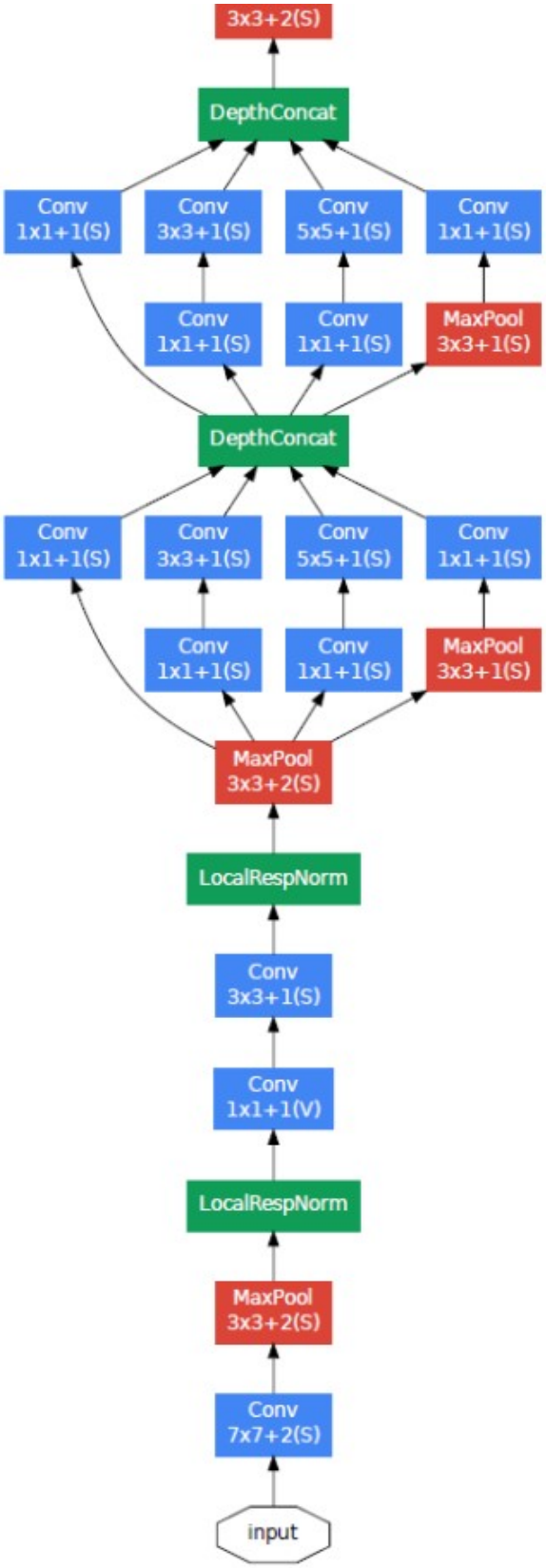


Fig 6.5 GoogLeNet Architecture

```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 224, 224, 3) 0
__________________________________________________________________________________________________
conv2d (Conv2D)                 (None, 112, 112, 64) 9472        input_1[0][0]
__________________________________________________________________________________________________
max_pooling2d (MaxPooling2D)    (None, 56, 56, 64)   0           conv2d[0][0]
__________________________________________________________________________________________________
batch_normalization (BatchNorma (None, 56, 56, 64)   256         max_pooling2d[0][0]
__________________________________________________________________________________________________
conv2d_1 (Conv2D)               (None, 56, 56, 64)   4160        batch_normalization[0][0]
__________________________________________________________________________________________________
conv2d_2 (Conv2D)               (None, 56, 56, 192)  110784      conv2d_1[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 56, 56, 192)  768         conv2d_2[0][0]
__________________________________________________________________________________________________
max_pooling2d_1 (MaxPooling2D)  (None, 28, 28, 192)  0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
conv2d_4 (Conv2D)               (None, 28, 28, 96)   18528       max_pooling2d_1[0][0]
__________________________________________________________________________________________________
conv2d_6 (Conv2D)               (None, 28, 28, 16)   3088        max_pooling2d_1[0][0]
__________________________________________________________________________________________________
max_pooling2d_2 (MaxPooling2D)  (None, 28, 28, 192)  0           max_pooling2d_1[0][0]
__________________________________________________________________________________________________
conv2d_3 (Conv2D)               (None, 28, 28, 64)   12352       max_pooling2d_1[0][0]
__________________________________________________________________________________________________
conv2d_5 (Conv2D)               (None, 28, 28, 128)  110720      conv2d_4[0][0]
__________________________________________________________________________________________________
conv2d_7 (Conv2D)               (None, 28, 28, 32)   12832       conv2d_6[0][0]
__________________________________________________________________________________________________
conv2d_8 (Conv2D)               (None, 28, 28, 32)   6176        max_pooling2d_2[0][0]
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 28, 28, 256)  0           conv2d_3[0][0]
                                                                 conv2d_5[0][0]
                                                                 conv2d_7[0][0]
                                                                 conv2d_8[0][0]
__________________________________________________________________________________________________
conv2d_10 (Conv2D)              (None, 28, 28, 128)  32896       concatenate[0][0]
__________________________________________________________________________________________________
conv2d_12 (Conv2D)              (None, 28, 28, 32)   8224        concatenate[0][0]
__________________________________________________________________________________________________
max_pooling2d_3 (MaxPooling2D)  (None, 28, 28, 256)  0           concatenate[0][0]
__________________________________________________________________________________________________
conv2d_9 (Conv2D)               (None, 28, 28, 128)  32896       concatenate[0][0]
__________________________________________________________________________________________________
conv2d_11 (Conv2D)              (None, 28, 28, 192)  221376      conv2d_10[0][0]
__________________________________________________________________________________________________
conv2d_13 (Conv2D)              (None, 28, 28, 96)   76896       conv2d_12[0][0]
__________________________________________________________________________________________________
conv2d_14 (Conv2D)              (None, 28, 28, 64)   16448       max_pooling2d_3[0][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 28, 28, 480)  0           conv2d_9[0][0]
                                                                 conv2d_11[0][0]
                                                                 conv2d_13[0][0]
                                                                 conv2d_14[0][0]
__________________________________________________________________________________________________
max_pooling2d_4 (MaxPooling2D)  (None, 14, 14, 480)  0           concatenate_1[0][0]
__________________________________________________________________________________________________
conv2d_16 (Conv2D)              (None, 14, 14, 96)   46176       max_pooling2d_4[0][0]
__________________________________________________________________________________________________
conv2d_18 (Conv2D)              (None, 14, 14, 16)   7696        max_pooling2d_4[0][0]
__________________________________________________________________________________________________
max_pooling2d_5 (MaxPooling2D)  (None, 14, 14, 480)  0           max_pooling2d_4[0][0]
__________________________________________________________________________________________________
conv2d_15 (Conv2D)              (None, 14, 14, 192)  92352       max_pooling2d_4[0][0]
__________________________________________________________________________________________________
conv2d_17 (Conv2D)              (None, 14, 14, 208)  179920      conv2d_16[0][0]
__________________________________________________________________________________________________
conv2d_19 (Conv2D)              (None, 14, 14, 48)   19248       conv2d_18[0][0]
__________________________________________________________________________________________________
conv2d_20 (Conv2D)              (None, 14, 14, 64)   30784       max_pooling2d_5[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 14, 14, 512)  0           conv2d_15[0][0]
                                                                 conv2d_17[0][0]
                                                                 conv2d_19[0][0]
                                                                 conv2d_20[0][0]
__________________________________________________________________________________________________
conv2d_23 (Conv2D)              (None, 14, 14, 112)  57456       concatenate_2[0][0]
__________________________________________________________________________________________________
conv2d_25 (Conv2D)              (None, 14, 14, 24)   12312       concatenate_2[0][0]
```

max_pooling2d_6 (MaxPooling2D)	(None, 14, 14, 512)	0	concatenate_2[0][0]
conv2d_22 (Conv2D)	(None, 14, 14, 160)	82080	concatenate_2[0][0]
conv2d_24 (Conv2D)	(None, 14, 14, 224)	226016	conv2d_23[0][0]
conv2d_26 (Conv2D)	(None, 14, 14, 64)	38464	conv2d_25[0][0]
conv2d_27 (Conv2D)	(None, 14, 14, 64)	32832	max_pooling2d_6[0][0]
concatenate_3 (Concatenate)	(None, 14, 14, 512)	0	conv2d_22[0][0] conv2d_24[0][0] conv2d_26[0][0] conv2d_27[0][0]
conv2d_29 (Conv2D)	(None, 14, 14, 128)	65664	concatenate_3[0][0]
conv2d_31 (Conv2D)	(None, 14, 14, 24)	12312	concatenate_3[0][0]
max_pooling2d_7 (MaxPooling2D)	(None, 14, 14, 512)	0	concatenate_3[0][0]
conv2d_28 (Conv2D)	(None, 14, 14, 128)	65664	concatenate_3[0][0]
conv2d_30 (Conv2D)	(None, 14, 14, 256)	295168	conv2d_29[0][0]
conv2d_32 (Conv2D)	(None, 14, 14, 64)	38464	conv2d_31[0][0]
conv2d_33 (Conv2D)	(None, 14, 14, 64)	32832	max_pooling2d_7[0][0]
concatenate_4 (Concatenate)	(None, 14, 14, 512)	0	conv2d_28[0][0] conv2d_30[0][0] conv2d_32[0][0] conv2d_33[0][0]
conv2d_35 (Conv2D)	(None, 14, 14, 144)	73872	concatenate_4[0][0]
conv2d_37 (Conv2D)	(None, 14, 14, 32)	16416	concatenate_4[0][0]
max_pooling2d_8 (MaxPooling2D)	(None, 14, 14, 512)	0	concatenate_4[0][0]
conv2d_34 (Conv2D)	(None, 14, 14, 112)	57456	concatenate_4[0][0]
conv2d_36 (Conv2D)	(None, 14, 14, 288)	373536	conv2d_35[0][0]
conv2d_38 (Conv2D)	(None, 14, 14, 64)	51264	conv2d_37[0][0]
conv2d_39 (Conv2D)	(None, 14, 14, 64)	32832	max_pooling2d_8[0][0]
concatenate_5 (Concatenate)	(None, 14, 14, 528)	0	conv2d_34[0][0] conv2d_36[0][0] conv2d_38[0][0] conv2d_39[0][0]
conv2d_42 (Conv2D)	(None, 14, 14, 160)	84640	concatenate_5[0][0]
conv2d_44 (Conv2D)	(None, 14, 14, 32)	16928	concatenate_5[0][0]
max_pooling2d_9 (MaxPooling2D)	(None, 14, 14, 528)	0	concatenate_5[0][0]
conv2d_41 (Conv2D)	(None, 14, 14, 256)	135424	concatenate_5[0][0]
conv2d_43 (Conv2D)	(None, 14, 14, 320)	461120	conv2d_42[0][0]
conv2d_45 (Conv2D)	(None, 14, 14, 128)	102528	conv2d_44[0][0]
conv2d_46 (Conv2D)	(None, 14, 14, 128)	67712	max_pooling2d_9[0][0]
concatenate_6 (Concatenate)	(None, 14, 14, 832)	0	conv2d_41[0][0] conv2d_43[0][0] conv2d_45[0][0] conv2d_46[0][0]
max_pooling2d_10 (MaxPooling2D)	(None, 7, 7, 832)	0	concatenate_6[0][0]
conv2d_48 (Conv2D)	(None, 7, 7, 160)	133280	max_pooling2d_10[0][0]
conv2d_50 (Conv2D)	(None, 7, 7, 32)	26656	max_pooling2d_10[0][0]
max_pooling2d_11 (MaxPooling2D)	(None, 7, 7, 832)	0	max_pooling2d_10[0][0]
conv2d_47 (Conv2D)	(None, 7, 7, 256)	213248	max_pooling2d_10[0][0]
conv2d_49 (Conv2D)	(None, 7, 7, 320)	461120	conv2d_48[0][0]
conv2d_51 (Conv2D)	(None, 7, 7, 128)	102528	conv2d_50[0][0]
conv2d_52 (Conv2D)	(None, 7, 7, 128)	106624	max_pooling2d_11[0][0]

concatenate_7 (Concatenate)	(None, 7, 7, 832)	0	conv2d_47[0][0] conv2d_49[0][0] conv2d_51[0][0] conv2d_52[0][0]
conv2d_54 (Conv2D)	(None, 7, 7, 192)	159936	concatenate_7[0][0]
conv2d_56 (Conv2D)	(None, 7, 7, 48)	39984	concatenate_7[0][0]
max_pooling2d_12 (MaxPooling2D)	(None, 7, 7, 832)	0	concatenate_7[0][0]
conv2d_53 (Conv2D)	(None, 7, 7, 384)	319872	concatenate_7[0][0]
conv2d_55 (Conv2D)	(None, 7, 7, 384)	663936	conv2d_54[0][0]
conv2d_57 (Conv2D)	(None, 7, 7, 128)	153728	conv2d_56[0][0]
conv2d_58 (Conv2D)	(None, 7, 7, 128)	106624	max_pooling2d_12[0][0]
concatenate_8 (Concatenate)	(None, 7, 7, 1024)	0	conv2d_53[0][0] conv2d_55[0][0] conv2d_57[0][0] conv2d_58[0][0]
average_pooling2d (AveragePooli	(None, 4, 4, 512)	0	concatenate_2[0][0]
average_pooling2d_1 (AveragePoo	(None, 4, 4, 528)	0	concatenate_5[0][0]
average_pooling2d_2 (AveragePoo	(None, 1, 1, 1024)	0	concatenate_8[0][0]
conv2d_21 (Conv2D)	(None, 4, 4, 128)	65664	average_pooling2d[0][0]
conv2d_40 (Conv2D)	(None, 4, 4, 128)	67712	average_pooling2d_1[0][0]
flatten_2 (Flatten)	(None, 1024)	0	average_pooling2d_2[0][0]
flatten (Flatten)	(None, 2048)	0	conv2d_21[0][0]
flatten_1 (Flatten)	(None, 2048)	0	conv2d_40[0][0]
dropout_2 (Dropout)	(None, 1024)	0	flatten_2[0][0]
dense (Dense)	(None, 256)	524544	flatten[0][0]
dense_1 (Dense)	(None, 256)	524544	flatten_1[0][0]
dense_2 (Dense)	(None, 256)	262400	dropout_2[0][0]
dropout (Dropout)	(None, 256)	0	dense[0][0]
dropout_1 (Dropout)	(None, 256)	0	dense_1[0][0]
main (Dense)	(None, 5)	1285	dense_2[0][0]
aux1 (Dense)	(None, 5)	1285	dropout[0][0]
aux2 (Dense)	(None, 5)	1285	dropout_1[0][0]

Total params: 7,423,295
Trainable params: 7,422,783
Non-trainable params: 512

Fig 6.6 GoogLeNet Parameters

6.3.5Training Accuracy

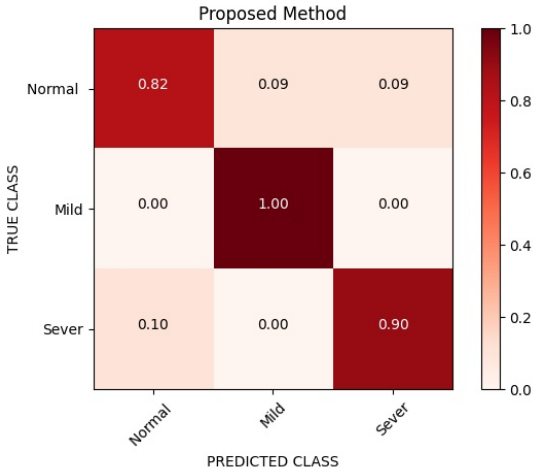


Fig 6.7 Training Accuracy Graph

6.3.6 Image Insert and Enhancement

- **Original Image Input**

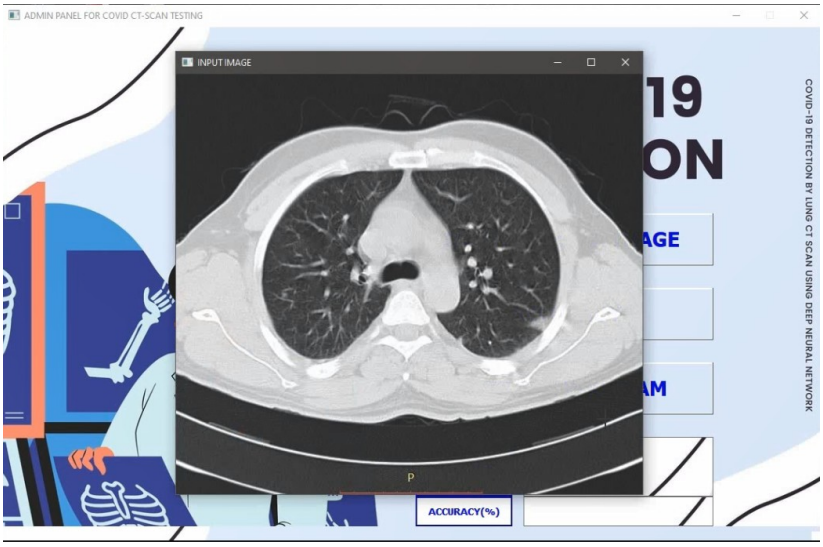


Fig 6.8 Original Image Input

Image Insert

```
1  image_path= filedialog.askopenfilename(filetypes = (("BROWSE CT IMAGE", "*.jpg"), ("All files", "*")))
2  I=cv2.imread(image_path)
3  cv2.imshow('INPUT IMAGE',I);
4  cv2.waitKey(1000)
```

- **Resizing Image**

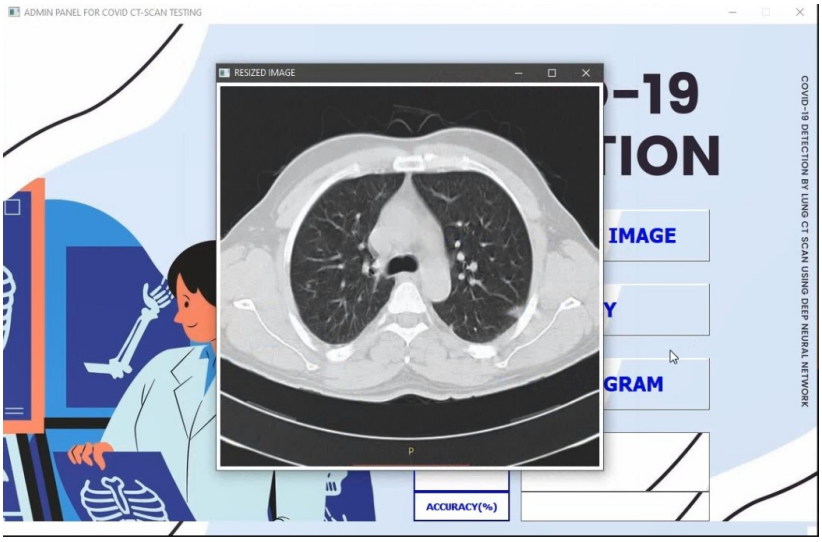


Fig 6.9 Resizing Image

Resizing Image

```
1  img = cv2.resize(I,(512,512),3)
2  cv2.imshow('RESIZED IMAGE',img)
3  cv2.waitKey(1000)
```

- **Median Filtered**

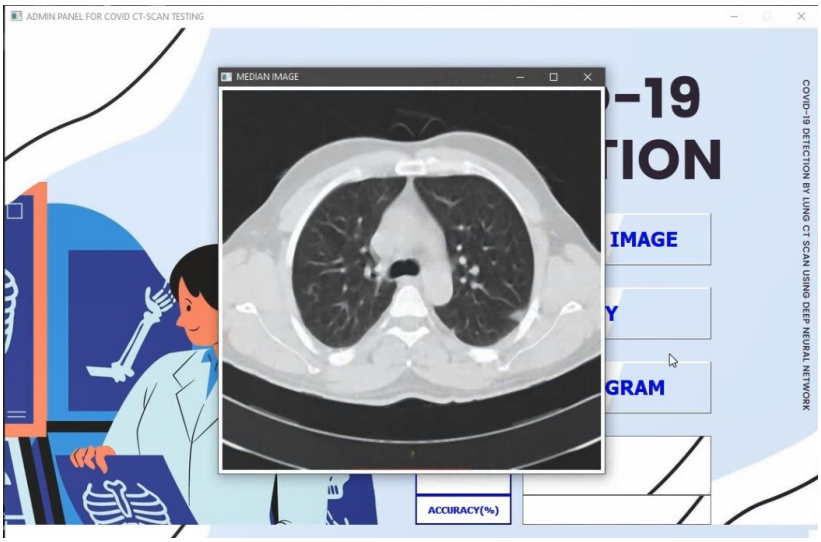


Fig 6.10 Median Image

Median Filtered

```
1  img1 = cv2.medianBlur(img,5)
2  cv2.imshow('MEDIAN IMAGE',img1)
3  cv2.waitKey(1000)
```

- **Gray Conversion**

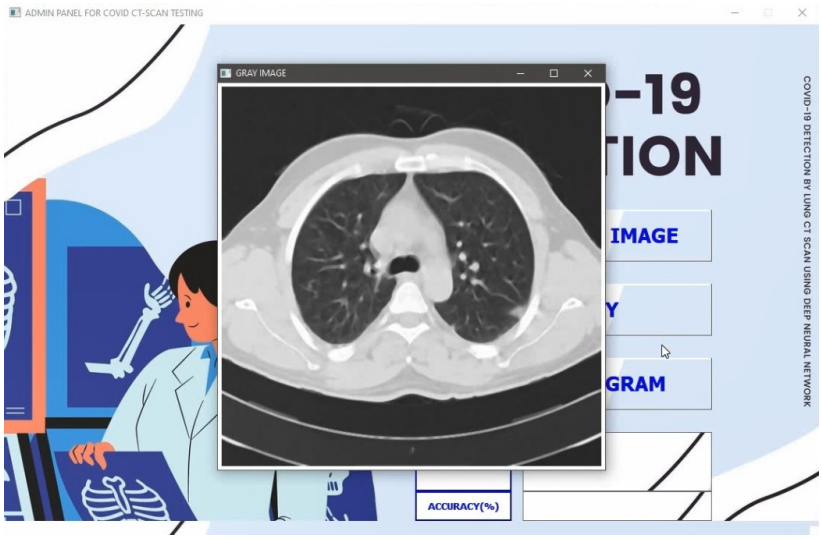


Fig 6.11 Gray Conversions

Gray Conversion

```
1  gray = cv2.cvtColor(img1,cv2.COLOR_BGR2GRAY)
2  cv2.imshow('GRAY IMAGE',gray)
3  cv2.waitKey(1000)
```

CHAPTER 7
SOFTWARE TESTING

7.1 TYPE OF TESTING

7.1.1 Unit Testing

Unit testing concentrates verification on the smallest element of the program – the module. Using the detailed design description important control paths are tested to establish errors within the bounds of the module.

In this system each sub module is tested individually as per the unit testing such as Register, Login, Covid-Test etc are tested individually. Their input field validations are tested.

7.1.2 Integration Testing

Once all the individual units have been tested there is a need to test how they were put together to ensure no data is lost across interface, one module does not have an adverse impact on another and a function is not performed correctly.

After unit testing each and every sub module is tested with integrating each other.

- **Functional Testing**

  Functional testing is the testing to ensure that the specified functionality required in the system requirements works. It falls under the class of black box testing.

- **System Testing**

  System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

- **Stress Testing**

  Stress testing is the testing to evaluate how system behaves under unfavorable conditions. Testing is conducted at beyond limits of the specifications. It falls under the class of black box testing.

- **Performance Testing**

  Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing.

- **Usability Testing**

  Usability testing is performed to the perspective of the client, to evaluate how the GUI is user-friendly? How easily can the client learn? After learning how to use, how proficiently can the client perform? How pleasing is it to use its design? This falls under the class of black box testing.

- **Acceptance Testing**

  Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing

System testing for the current system

In this level of testing we are testing the system as a whole after integrating all the main modules of the project. We are testing whether system is giving correct output or not. All the modules were integrated and the flow of information among different modules was checked. It was also checked that whether the flow of data is as per the requirements or not. It was also checked that whether any particular module is non-functioning or not i.e. once the integration is over each and every module is functioning in its entirety or not.

In this level of testing we tested the following: -

- Whether all the forms are properly working or not.
- Whether all the forms are properly linked or not.

- Whether all the databases are properly linked or not.
- Whether all the images are properly displayed or not.
- Whether data retrieval is proper or not.

7.2 TEST CASES AND TEST RESULTS

Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. The tester is aware of *what* the software is supposed to do but is not aware of how it does it. For instance, the tester is aware that a particular input returns a certain, invariable output but is not aware of *how* the software produces the output in the first place.

- **Test Cases**

  Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output without any knowledge of the test object's internal structure.

- **Test Design Techniques**

  Typical black-box test design techniques include:

- Decision table testing
- All-pairs testing
- State transition Analysis
- Equivalence partitioning
- Boundary value analysis
- Cause–effect graph
- Error guessing

Advantages

- Efficient when used on large systems.
- Since the tester and developer are independent of each other, testing is balanced and unprejudiced.
- Tester can be non-technical.
- There is no need for the tester to have detailed functional knowledge of system.
- Tests will be done from an end user's point of view, because the end user should accept the system. (This testing technique is sometimes also called Acceptance testing.)
- Testing helps to identify vagueness and contradictions in functional specifications.
- Test cases can be designed as soon as the functional specifications are complete.

Disadvantages

- Test cases are challenging to design without having clear functional specifications.
- It is difficult to identify tricky inputs if the test cases are not developed based on specifications.
- It is difficult to identify all possible inputs in limited testing time. As a result, writing test cases may be slow and difficult.
- There are chances of having unidentified paths during the testing process.
- There is a high probability of repeating tests already performed by the programmer.

TEST CASE AND RESULTS

Test Case ID	1
Test Case Description	Checking the functionality of “CLICK TO REGISTER” button
Steps for testing	1. Enter user name 2. Enter user email id (must following @__mail.com) 3. Enter user mobile number (must be10 digits) 4. Enter user password 5. Click “ CLICK TO REGISTER” button

Test Case Result	Message box “Registration successfully” should be display
Action Result	Message box displayed
Status	Pass
Test Case ID	2
Test Case Description	Checking the functionality of “CLICK TO REGISTER” button
Steps for testing	1. Enter user name 2. Enter user email id (not following @__mail.com) 3. Enter user mobile number (not be10 digits) 4. Enter user password 5. Click “ CLICK TO REGISTER” button
Test Case Result	Message box “Invalid email or Number ” should be display
Action Result	Message box displayed
Status	Pass
Test Case ID	3
Test Case Description	Checking the functionality of “CLICK TO LOGIN PAGE” button
Steps for testing	Click ““CLICK TO LOGIN PAGE”
Test Case Result	LOGIN FORM should be displayed
Action Result	LOGIN FORM displayed
Status	Pass
Test Case ID	4
Test Case Description	Checking the functionality of “CLICK TO EXIT” button
Steps for testing	Click “CLICK TO EXIT”
Test Case Result	The system should be terminated
Action Result	The system terminated
Status	Pass
Test Case ID	5
Test Case Description	Checking the functionality of “LOGIN TO SYSTEM” button
Steps for testing	1. Enter valid user name 2. Enter valid user password 3. Click “LOGIN TO SYSTEM” button
Test Case Result	“COVID-19 DETECTION” panel should be displayed
Action Result	The panel displayed
Status	Pass
Test Case ID	6
Test Case Description	Checking the functionality of “LOGIN TO SYSTEM” button
Steps for testing	1. Enter invalid user name 2. Enter invalid user password 3. Click “LOGIN TO SYSTEM” button
Test Case Result	“COVID-19 DETECTION” panel should not be displayed and showed “Login failed” in message box
Action Result	The panel not displayed and showed “Login failed”
Status	Pass
Test Case ID	7
Test Case Description	Checking the functionality of “INPUT CT-SCAN IMAGE” button
Steps for testing	1. Click “INPUT CT-SCAN IMAGE” button

	2. Insert Lung CT scan image
Test Case Result	The system should do all function (Input Image, Resized image, Median Image, Gray Conversion), and Result with Accuracy should be displayed
Action Result	All function worked and displayed
Status	Pass
Test Case ID	8
Test Case Description	Checking the functionality of "ACCURACY" button
Steps for testing	1. Click "ACCURACY" button
Test Case Result	The system should be displayed Accuracy Graph
Action Result	Accuracy graph displayed
Status	Pass

Modules to be tested: Registration Form

- Enter email id which not following "@__mail.com" click on "CLICK TO REGISTER" button

  Expected: It should display "Invalid email"

  Result: Pass
- Enter email id which following "@__mail.com" click on "CLICK TO REGISTER" button

  Expected: It should display "Registration successfully"

  Result: Pass
- Enter mobile number not 10 digits "CLICK TO REGISTER" button

  Expected: It should display "Invalid number"

  Result: Pass
- Enter mobile number in 10 digits "CLICK TO REGISTER" button

  Expected: It should display "Registration successfully"

  Result: Pass
- "CLICK TO EXIT" button

  Expected: It should terminate the system

  Result: Pass

Modules to be tested: Login Form

- Enter invalid user name and user password click “LOGIN TO SYSTEM”

  Expected: It should not display “COVID-19 DETECTION” panel and shows “Login failed”

  Result: Pass
- Enter valid user name and user password click “LOGIN TO SYSTEM”

  Expected: It should display “COVID-19 DETECTION” panel

  Result: Pass
- “CLICK TO EXIT” button

  Expected: It should terminate the system

  Result: Pass

Modules to be tested: COVID-19 DETECTION panel

- Click “INPUT CT-SCAN IMAGE” and insert image

  Expected: It should do all functions and display the results

  Result: Pass
- Click “INPUT CT-SCAN IMAGE” and insert image

  Expected: It should do all functions and display the results

  Result: Pass
- “ACCURACY” button

  Expected: It should display Accuracy Graph

  Result: Pass
- “EXIT THE PROGRAM” button

  Expected: It should terminate the system

  Result: Pass

CHAPTER 8

RESULTS

8.1 OUTCOMES

	ACCURACY_ALL_STAGES	RESULT_ALL_STAGES
0	87.569433	CLASS: COVID: SEVER STAGE--:
1	97.540367	CLASS: COVID: SEVER STAGE--:
2	85.514830	CLASS: NORMAL--:
3	93.674446	CLASS: NORMAL--:
4	89.956076	CLASS: NORMAL--:
5	85.757529	CLASS: COVID: I STAGE--:
6	89.452484	CLASS: COVID: I STAGE--:
7	88.377305	CLASS: COVID: I STAGE--:
8	75.410497	CLASS: COVID: SEVER STAGE--:
9	82.175028	CLASS: COVID: II STAGE--:
10	85.317376	CLASS: COVID: II STAGE--:
11	88.191350	CLASS: COVID: II STAGE--:
12	81.622881	CLASS: COVID: I STAGE--:
13	76.780719	CLASS: COVID: SEVER STAGE--:
14	74.295918	CLASS: COVID: II STAGE--:
15	82.072531	CLASS: COVID: II STAGE--:
16	89.259363	CLASS: COVID: I STAGE--:
17	88.194447	CLASS: COVID: I STAGE--:
18	74.986058	CLASS: COVID: II STAGE--:
19	89.530136	CLASS: COVID: SEVER STAGE--:
20	80.277365	CLASS: COVID: I STAGE--:
21	84.517808	CLASS: COVID: II STAGE--:
22	80.768191	CLASS: COVID: I STAGE--:
23	88.496257	CLASS: COVID: I STAGE--:
24	80.647004	CLASS: COVID: I STAGE--:
25	73.674446	CLASS: NORMAL--:
26	85.514830	CLASS: NORMAL--:
27	85.317376	CLASS: COVID: SEVER STAGE--:
28	82.072531	CLASS: COVID: SEVER STAGE--:
29	81.622881	CLASS: COVID: I STAGE--:
30	95.956076	CLASS: NORMAL--:
31	82.175028	CLASS: NORMAL--:
32	84.655984	CLASS: COVID: II STAGE--:
33	82.072531	CLASS: COVID: II STAGE--:
34	84.185462	CLASS: COVID: I STAGE--:
35	81.194447	CLASS: COVID: II STAGE--:
36	97.540367	CLASS: COVID: II STAGE--:
37	85.317376	CLASS: COVID: I STAGE--:
38	75.956076	CLASS: COVID: SEVER STAGE--:
39	82.175028	CLASS: COVID: II STAGE--:
40	82.072531	CLASS: COVID: I STAGE--:
41	97.540367	CLASS: COVID: II STAGE--:
42	82.175028	CLASS: COVID: I STAGE--:
43	85.252565	CLASS: COVID: II STAGE--:
44	75.956076	CLASS: COVID: II STAGE--:
45	81.622881	CLASS: COVID: SEVER STAGE--:
46	82.072531	CLASS: NORMAL--:
47	78.365217	CLASS: COVID: I STAGE--:
48	84.655984	CLASS: COVID: I STAGE--:
49	83.622881	CLASS: COVID: SEVER STAGE--:
50	97.635489	CLASS: COVID: II STAGE--:
51	84.655984	CLASS: COVID: II STAGE--:
52	95.956076	CLASS: COVID: SEVER STAGE--:
53	85.317376	CLASS: COVID: II STAGE--:
54	91.194447	CLASS: COVID: II STAGE--:
55	95.956076	CLASS: COVID: SEVER STAGE--:
56	81.622881	CLASS: COVID: II STAGE--:
57	85.317376	CLASS: COVID: I STAGE--:
58	97.540367	CLASS: COVID: SEVER STAGE--:
59	82.072531	CLASS: NORMAL--:

	ACCURACY_ALL_STAGES
count	60.000000
mean	85.173675
std	6.259845
min	73.674446
25%	81.622881
50%	84.655984
75%	88.407043
max	97.635489

RESULT_ALL_STAGES

CLASS: COVID: I STAGE--:	18
CLASS: COVID: II STAGE--:	20
CLASS: COVID: SEVER STAGE--:	13
CLASS: NORMAL--:	9

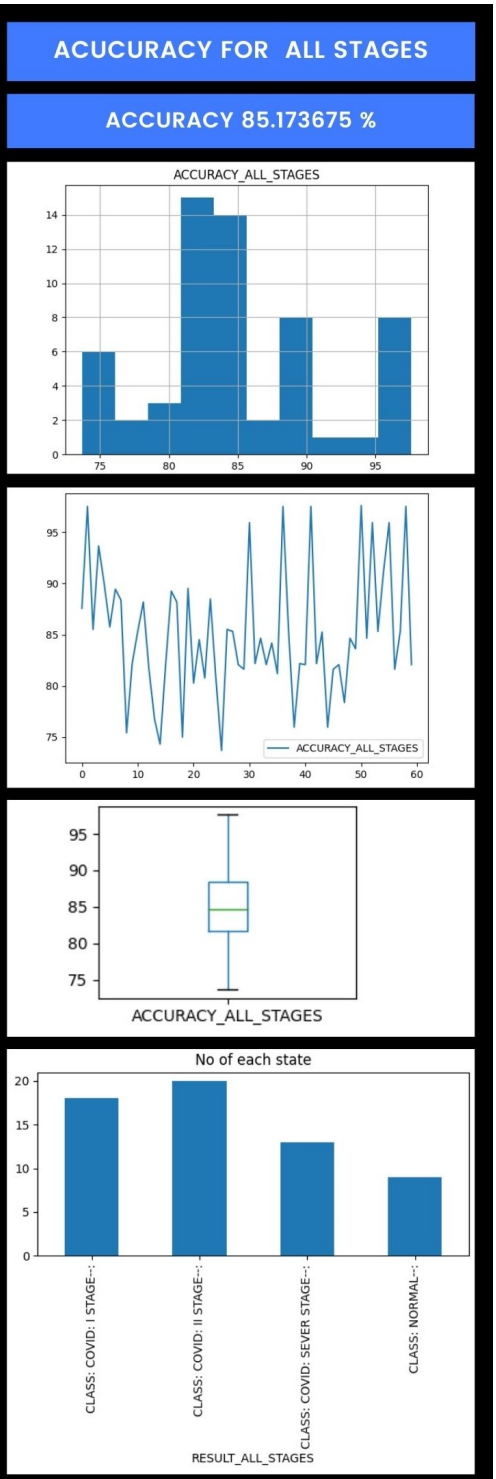


Fig 8.1 Accuracy for all stages

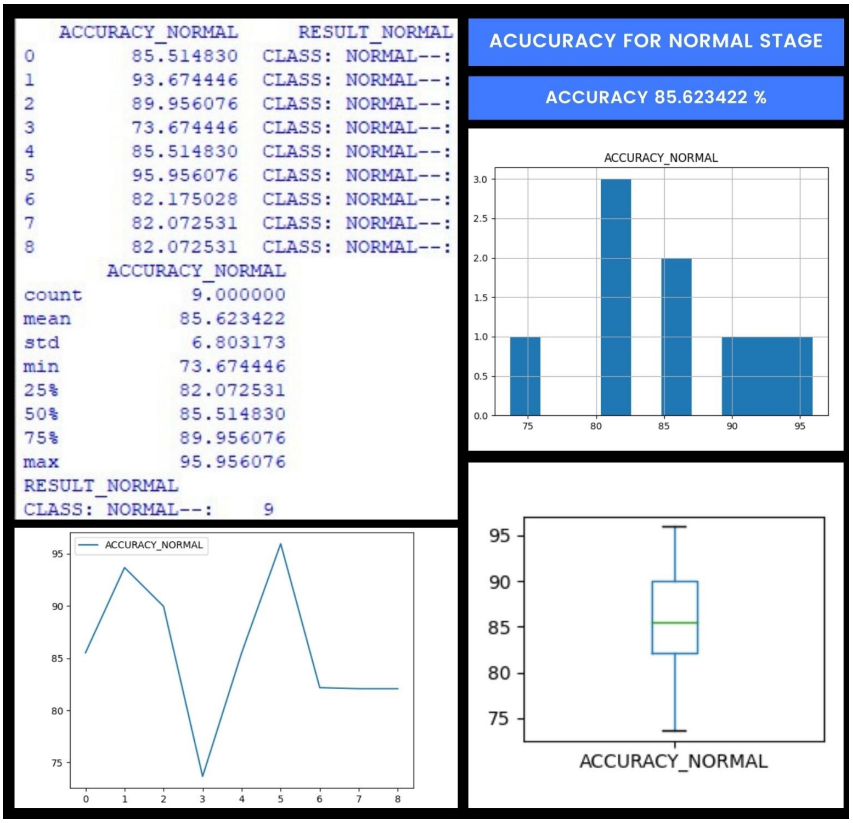


Fig 8.2 Accuracy for normal stage

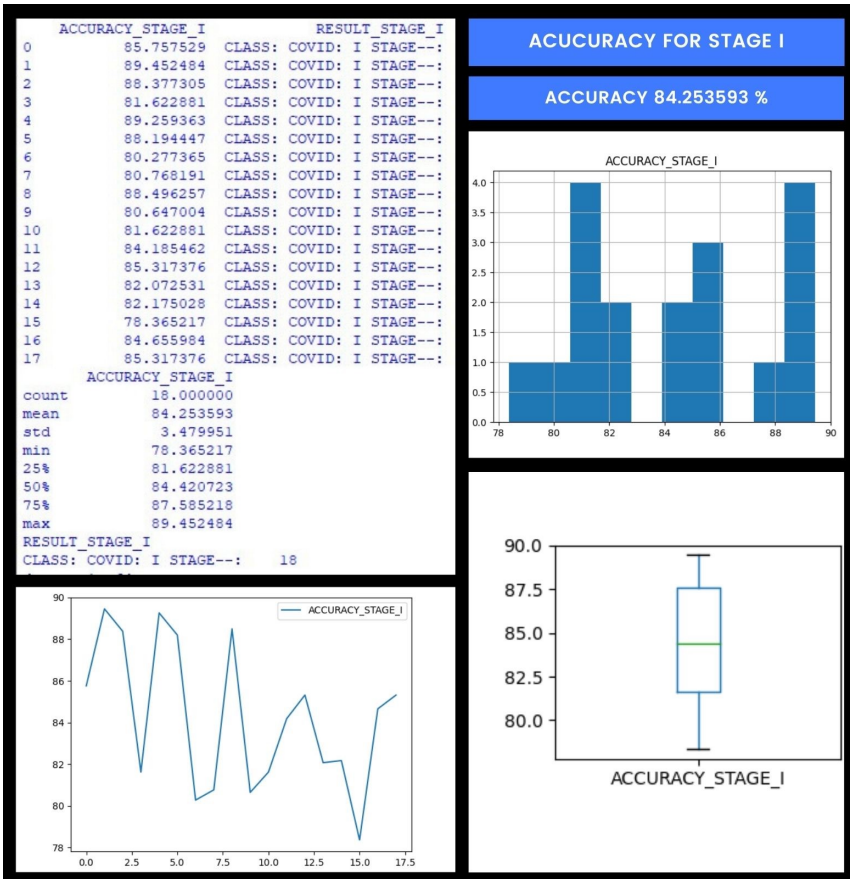


Fig 8.3 Accuracy for stage I

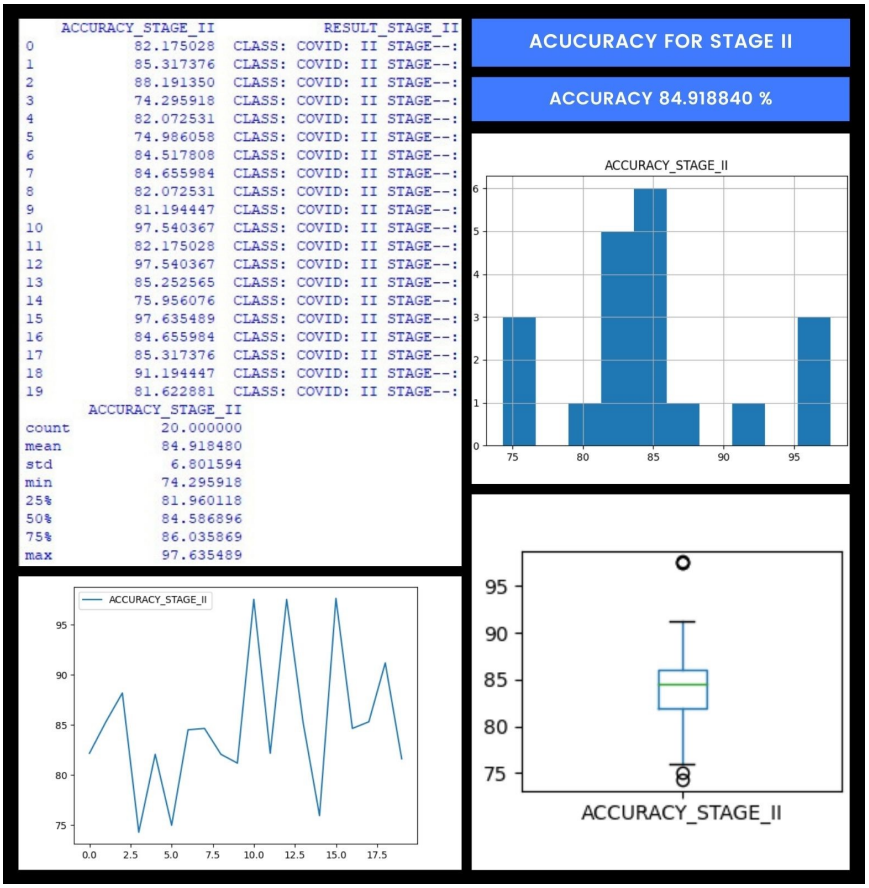


Fig 8.4 Accuracy for stage II

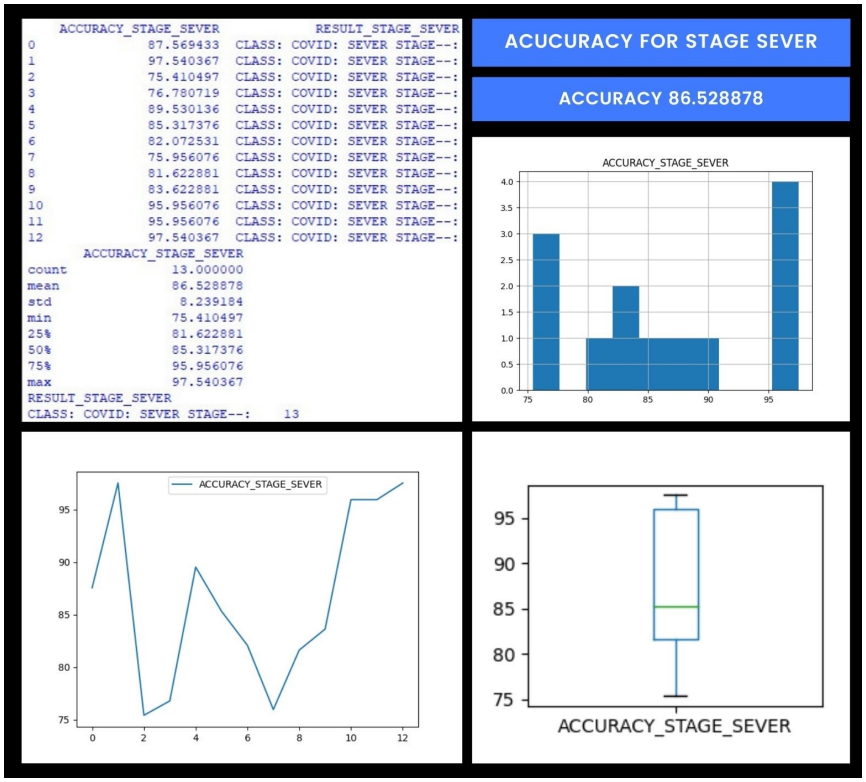


Fig 8.5 Accuracy for stage sever

8.2 SCREEN SHOTS

Fig 8.6 User Registration UI

Fig 8.7 User Login UI

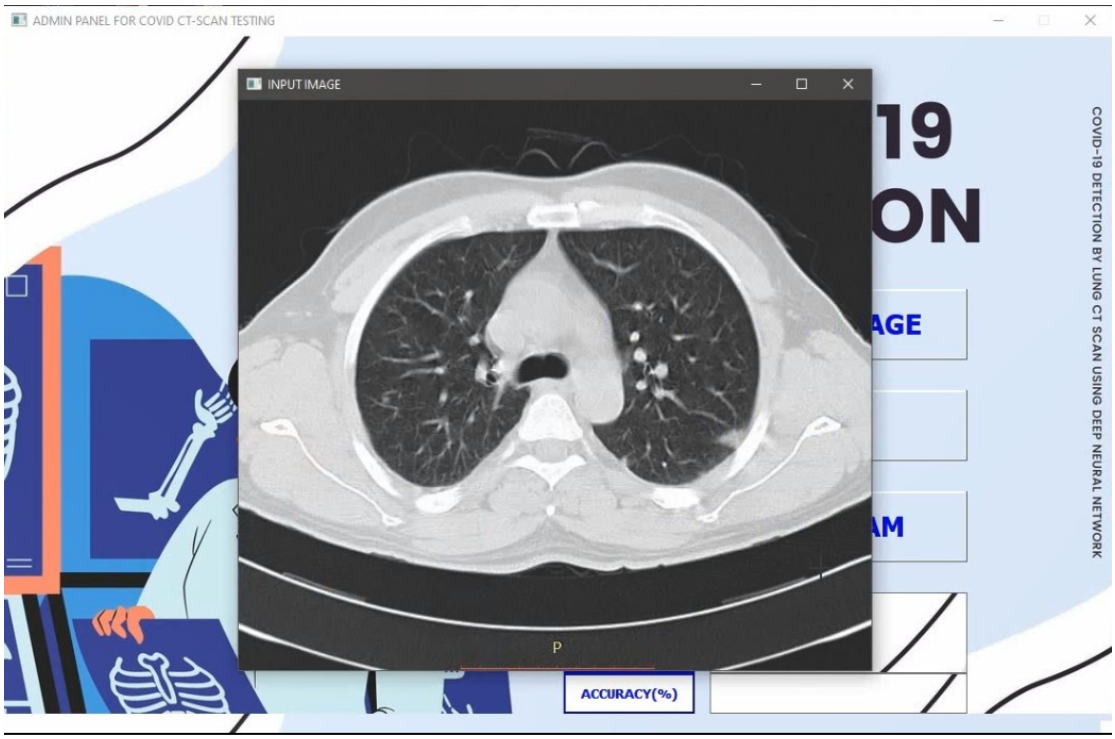


Fig 8.8 Input Images UI

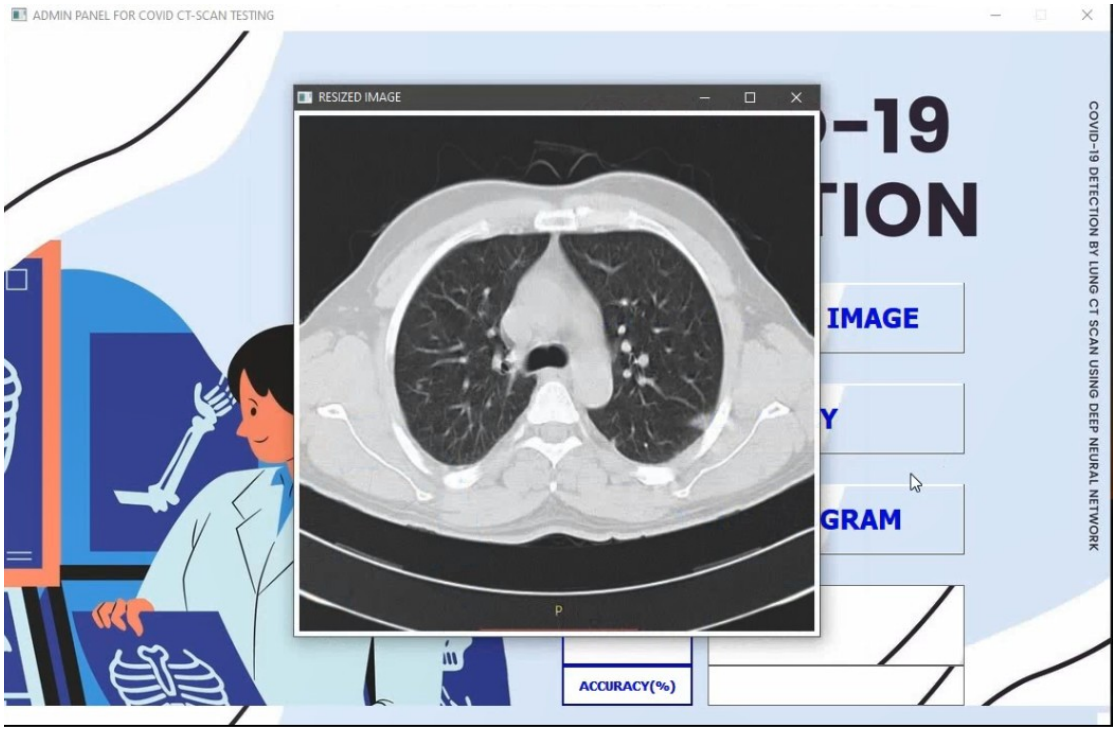


Fig 8.9 Resized Images UI

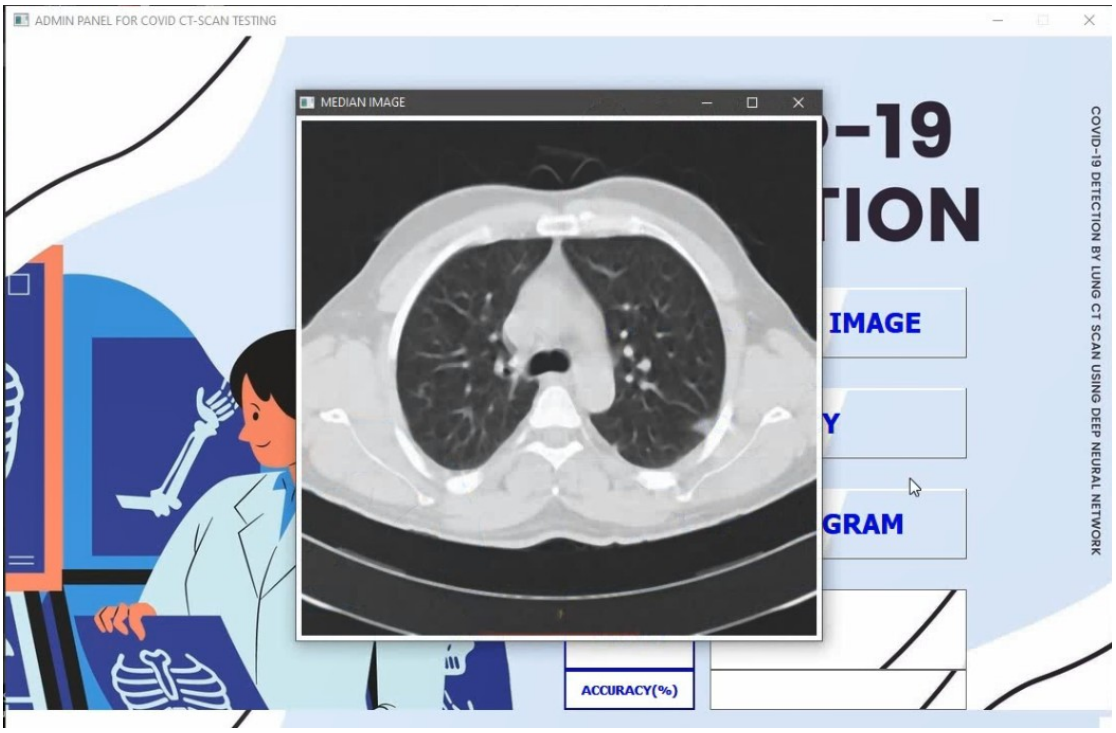


Fig 8.10 Median Images UI

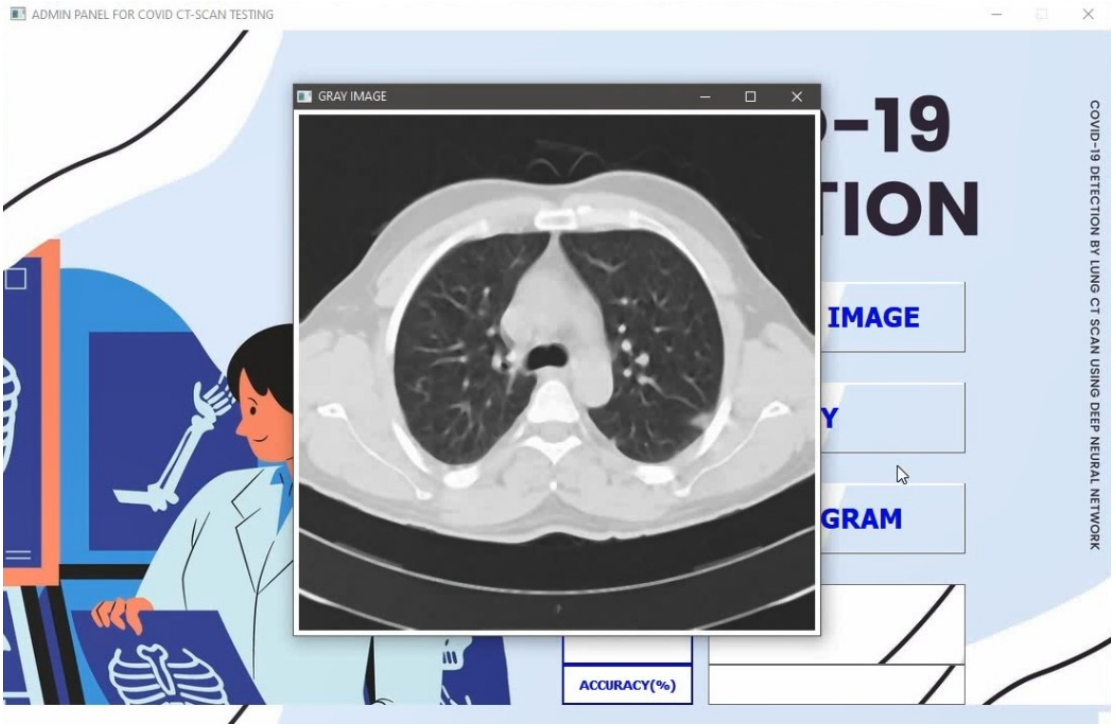


Fig 8.11 Gray Images UI

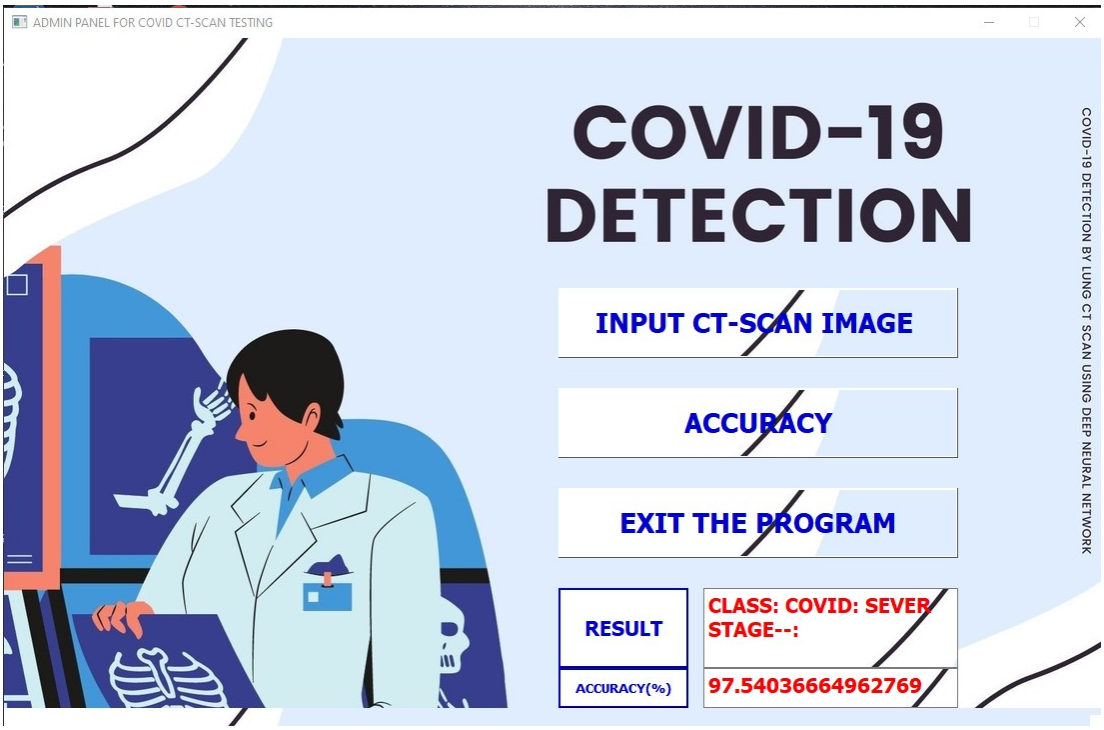


Fig 8.12 Covid-19 Detection UI

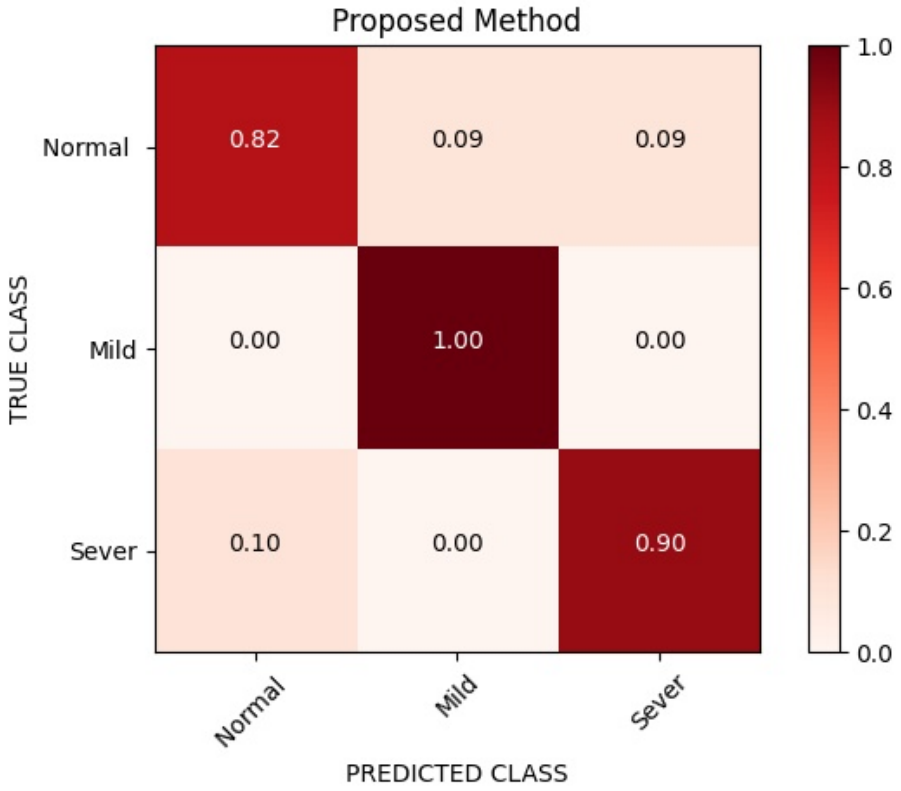


Fig 8.13 Accuracy UI

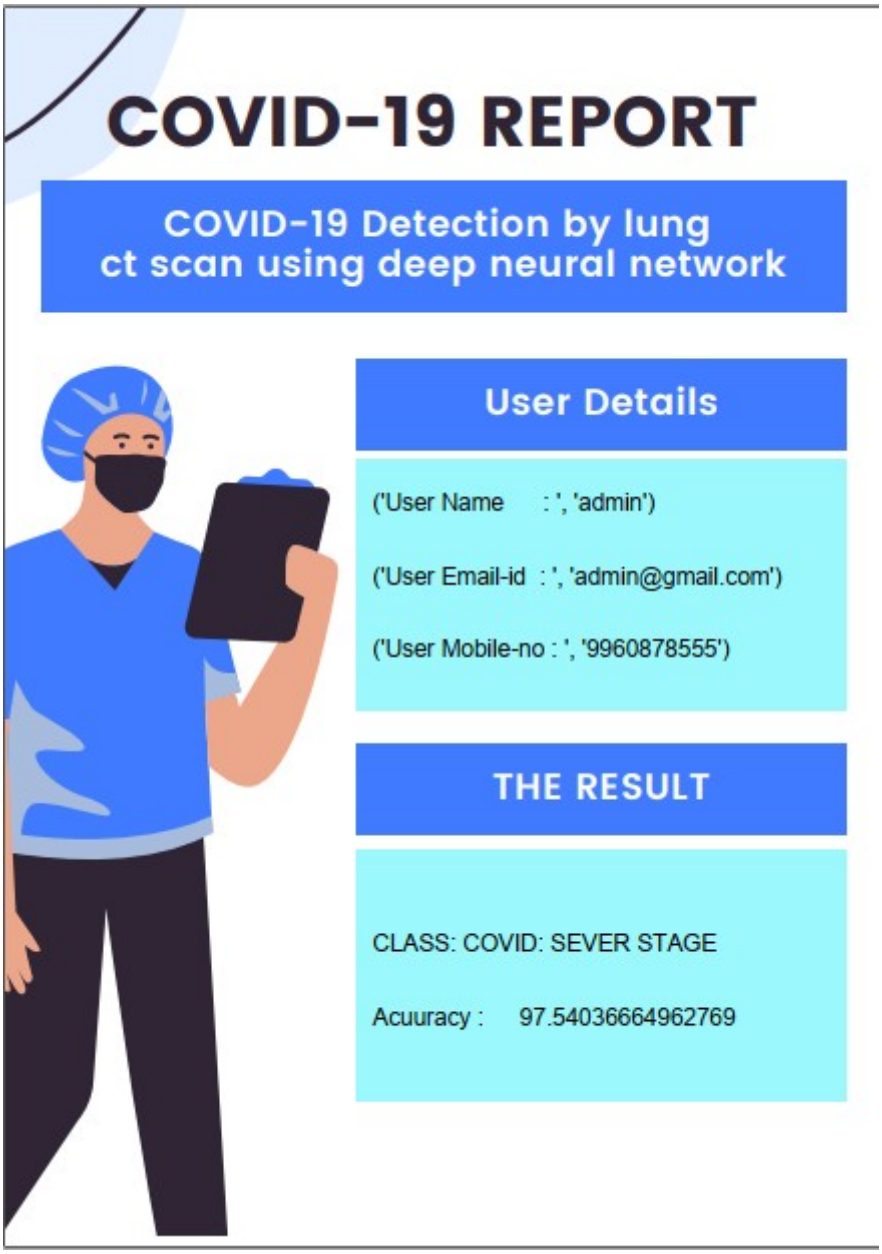


Fig 8.14 Report Covid Template PDF

CHAPTER 9
CONCLUSIONS

9.1 CONCLUSIONS

We are going to invent covid-19 detection on CT scan images of lungs that system over machine learning and CNN techniques which solves existing accuracy problem as well as reduce death rates or spreading by pandemic disease like covid-19. For future work, we can implement this technique on some more diseases with rich dataset. Increasing the number of diseases and dataset used for the process can improve the accuracy.

9.2 LIMITATIONS

- Large number of training dataset is required for more accuracy.

9.3 FUTURE WORK

- In future work we are going to focus on more detailed study and developed Android apps for medical related disease.

9.4 ADVANTAGES

- To reduce death rate by early detection of Covid in the world.
- To give unique solution for covid-19 diseases.
- To give most promising tool that can acceptable by all the doctors.

9.5 APPLICATION

- Our application is used in hospital.
- Our application is also used in clinic.

Appendix A: Problem statement feasibility assessment using, satisfiability analysis and NP Hard, NP-Complete or P type using modern algebra and relevant mathematical models.

Title:

Project problem statement is feasibility assessment using NP-Hard, NP-Complete or satiability issues using modern algebra and relevant mathematical models.

1. Theory:

2.1 What is P?

- P is set of all decision problems which can be solved in polynomial time by a deterministic.
- Since it can be solved in polynomial time, it can be verified in polynomial time.
- Therefore P is a subset of NP.

2.2 What is N?

- "N" in "NP" refers to the fact that you are not bound by the normal way a computer works, which is step-by-step. The "N" actually stands for "Non-deterministic". This means that you are dealing with an amazing kind of computer that can run things simultaneously or could somehow guess the right way to do things, or something like that.
- So this "N" computer can solve lots more problems in "P" time - for example it can just clone copies of itself when needed.
- So, programs that takes dramatically longer as the problem gets harder (i.e. not in "P") could be solved quickly on this amazing "N" computer and so are in "NP".
- Thus "NP" means "we can solve it in polynomial time if we can break the normal rules of step-by-step computing".

2.3 What is NP?

- "NP" means "we can solve it in polynomial time if we can break the normal rules of step-by-step computing".

2.4 Project status:

Project Status:

Problem: The main problem how to disease prediction using Images.

Solution: In the Feasibility Study stage, the assigned project is analyzed, then information about the project participants is collected, and the requirements for the system are gathered and analyzed. During the Feasibility Study stage, the project's goals, parameters and restraints are agreed and a conceptual problem solution is prepared. Admin add all disease related images and symptoms . All this procedure forcovid-19 disease prediction due to help of Images is possible in real time, so this project is NP-Complete.

The project is NP-Complete.

Mathematical Model

- **System Description:**
- **Mathematical Model:** Let us consider S as a system for automatically recommends vehicle to customer. S={F,I,O,e, Φ}}
- **INPUT:**

Identify the inputs F= f1, f2, f3 ....., fn— F as set of functions to execute commands.

- **I=** i1, i2, i3 Sets of inputs to the function set
- **O**= o1, o2, o3 Set of outputs from the function sets,
- **e** = End of the program.

S1= I, F, O

- **I** = Query submitted by the User, i.e. query images
- **O** = Output of desired query, i.e. covid stage prediction
- **F** = Functions implemented to get the output, i.e. CNN for extraction of image data.

A] Mapping Diagram

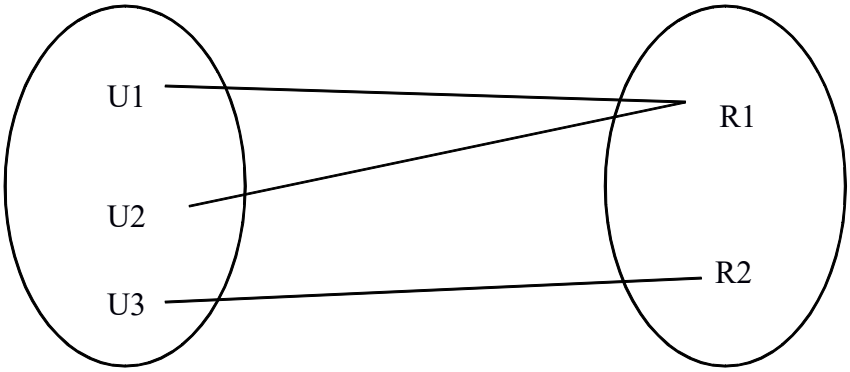


Fig 6.1 Mapping Diagram

Where,

U=users

R=Cancer stage.

U1=Search image1 for get right covid type

U2= Search image2 for get right covid type
U3=Search image1 for get right covid type
R1= Result of right covid disease stage.
R2=Result of wrong covid disease stage.

4. Set Theory

S={s, e, X, Y,Φ}

Where,

s = Start of the program.

1. User Login with Credential.

2. User inserts the image for checking covid-19 disease.

3. System predict the stage of covid-19 disease.

e = End of the program.

- **Failures and Success conditions.**

Success:

1. Search the required information from available image data in the dataset.
2. Customer gets result very fast according to their needs.

Failures:

1. Huge database can lead to more time consumption to get the information.
2. Hardware failure.
3. Software failure.

Appendix B: Details of the papers referred in IEEE format (given earlier) Summary of the above paper in not more than 3-4 lines. Here you should write the seed idea of the papers you had referred for preparation of this project report in the following format.

[1] Wei Zhao, ZhengZhong, XingzhiXie, Qizhi Yu and Jun Liu **"Relation Between Chest CT Findings and Clinical Conditions of Coronavirus Disease (COVID-19) Pneumonia: A Multicenter Study,"** American Journal of Roentgenology: 1-6. 10.2214/AJR.20.22976

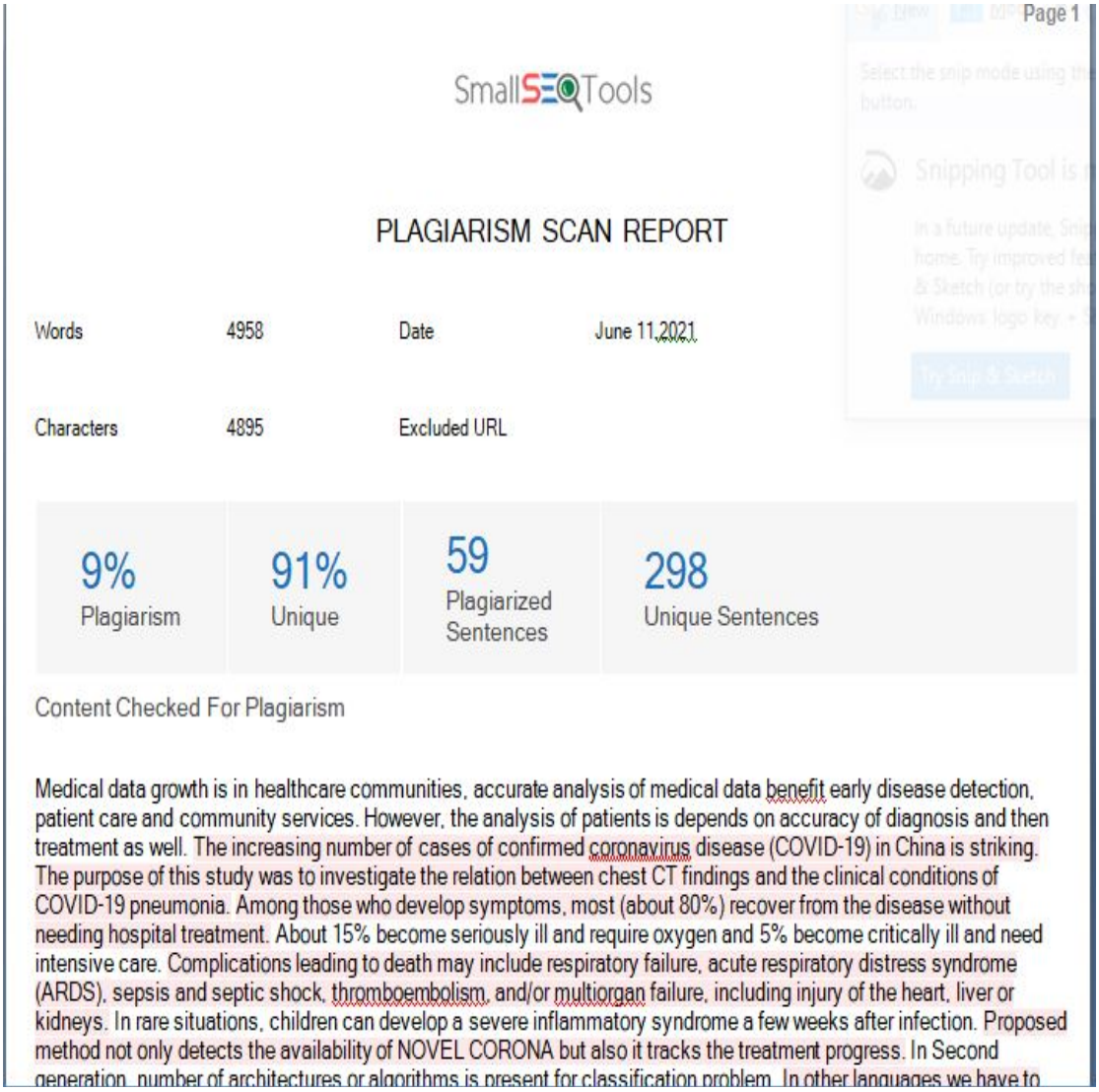
[2]Wang W, Xu Y, Gao R, **"Detection of SARS-CoV-2 in Different Types of Clinical Specimens. JAMA"** Published online March 11, 2020. doi:10.1001/jama.2020.3786

[3] Tao Ai, Zhenlu Yang, HongyanHou, Chenao Zhan, Chong Chen, WenzhiLv, Qian Tao, Ziyong Sun, Liming Xia ," **Correlation of Chest CT and RT-PCR Testing in Coronavirus Disease 2019(COVID-19) in China: A Paper of 1014 Cases**" Feb 26 2020https://doi.org/10.1148/radiol.2020200642.

[4] Laura Matrajt, Tiffany Leung **"Evaluating the Effectiveness of Social Distancing Interventions to Delay or Flatten the Epidemic Curve of Coronavirus Disease"** pmc article doi: 10.3201/eid2608.201093. Epub 2020 Apr 28.

[5] ShashiYadav "**Deep Learning based Safe Social Distancing and Face Mask Detection in Public Areas for COVID- 19 Safety Guidelines Adherence"**International Journal for Research in Applied Science & Engineering Technology (IJRASET) 2020

Appendix C: Plagiarism Report



Page 1

SmallSEOTools

PLAGIARISM SCAN REPORT

Words	4958	Date	June 11,2021
Characters	4895	Excluded URL	

9%	91%	59	298
Plagiarism	Unique	Plagiarized Sentences	Unique Sentences

Content Checked For Plagiarism

Medical data growth is in healthcare communities, accurate analysis of medical data benefit early disease detection, patient care and community services. However, the analysis of patients is depends on accuracy of diagnosis and then treatment as well. The increasing number of cases of confirmed coronavirus disease (COVID-19) in China is striking. The purpose of this study was to investigate the relation between chest CT findings and the clinical conditions of COVID-19 pneumonia. Among those who develop symptoms, most (about 80%) recover from the disease without needing hospital treatment. About 15% become seriously ill and require oxygen and 5% become critically ill and need intensive care. Complications leading to death may include respiratory failure, acute respiratory distress syndrome (ARDS), sepsis and septic shock, thromboembolism, and/or multiorgan failure, including injury of the heart, liver or kidneys. In rare situations, children can develop a severe inflammatory syndrome a few weeks after infection. Proposed method not only detects the availability of NOVEL CORONA but also it tracks the treatment progress. In Second generation, number of architectures or algorithms is present for classification problem. In other languages we have to