

COVID-19 Prediction Using X-Ray Images

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in

Information Technology

By

APARNA G (Reg. No.37120010)
BHARATHI R (Reg. No. 37120015)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119

MARCH 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)
Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai – 600 119
www.sathyabama.ac.in



DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **G Aparna (37120010), R Bharathi (37120015)** who carried out the project entitled **"COVID-19 Prediction Using X-Ray Images"** under my supervision from November 2020 to March 2021.

Internal Guide

Dr. S.GOWRI. M.E., Ph.D.,

Head of the Department

Dr. R.SUBHASHINI. M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, **APARNA G (37120010), BHARATHI R (37120015)** hereby declare that the Project Report entitled **“COVID-19 Prediction Using X-Ray Images”** done by us under the guidance of **Dr. S. GOWRI. M.E., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr.R.Subhashini M.E., Ph.D., Head** of the Department of Information Technology for providing me necessary support and details at the right time during the progressive reviews.

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

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTRACT

Corona virus disease (COVID-19) is a pandemic caused by the corona virus SARS-CoV-2 that was not previously seen in humans. COVID-19 is spreading rapidly throughout the world. COVID-19 can be detected by a lung infection of the patients. The standard method for detecting COVID-19 is the Reverse transcription-polymerase chain reaction (RT-PCR) test. But, the availability of RT-PCR tests is in short supply. As a result of this, the early detection of the disease is difficult. The easily obtainable modes like X-rays are often used for detecting infections in the lungs. It is confirmed that X-ray scans can be widely used for efficient COVID-19 diagnosis. But a physical diagnosis of X-rays of an outsized number of patients is a long term process. A deep learning-based diagnosis process can help radiologists in detecting COVID-19 from X-ray scans. Pretrained CNNs are commonly used in detecting diseases from datasets. In this paper, we propose a CNN model with a parallelization strategy that extracts the features in the X-ray images by applying filters parallelly through the images. Our proposed method aims to attain higher accuracy and a less loss rate with precision. In this paper, we compare the accuracy and loss rates of three types of CNN - VGG-16, MobileNet and CNN with the parallelization technique. Here we use two datasets consisting of COVID X-ray images and Non-COVID X-ray images for the prediction of COVID-19 using Convolution Neural Network.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO
1.	INTRODUCTION	
	1.1 CORONA VIRUS	1
	1.2 TYPES OF COVID	2
	1.3 STRUCTURE OF COVID	3
	1.4 DIAGNOSIS OF COVID	4
	1.5 ROLE OF X-RAY IMAGES IN FINDING COVID 19	5
2.	LITERATURE REVIEW	
	2.1 LITERATURE REVIEW	9
3.	METHODOLOGY	
	3.1 DATA COLLECTION	11
	3.1.1 PA PROJECTION OF X-RAY	11
	3.1.2 ADVANTAGES OF PA VIEW	11
	3.1.3 SEGREGATION OF IMAGES	12
	3.2 DATA PREPROCESSING	12
	3.2.1 NUMPY ARRAY	13
	3.2.2 CATEGORICAL FORM	13
	3.3 TRAINING THE CNN	13
	3.3.1 VGG 16	14
	3.3.2 MobileNet	16
	3.3.3 PARALLEL CNN	18
	3.4 WEB APP IMPLEMENTATION	23
4.	RESULTS AND DISCUSSION	
	4.1 RESULT	24
	4.2 VALIDATION ACCURACY	24
	4.3 VALIDATION LOSS	25
5.	CONCLUSION	
	5.1 CONCLUSION	29
	REFERENCES	30
	APPENDICES	33

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
1.1	Structure of Covid	3
1.2	RT-PCR Test	5
1.3	Types of View in X-ray Images	6
1.4	Ground Glass Opacity	6
1.5	Consolidation	7
1.6	Serial Radiological progression seen with Covid-19 pneumonia.	8
3.1	Covid Positive and Negative Folders	12
3.2	Layers of VGG-16	14
3.3	Architecture of VGG-16	15
3.4	Layers in MobileNet	17
3.5	Architecture of MobileNet	17
3.6	Architecture of Proposed CNN model	20
3.7	Dataflow of Proposed CNN	21
3.8	Model Summary	22
3.9	Output Screen	23
3.10	Selecting Image using ImageSelector	23
4.1	Sample of Covid Positive and Negative Image	24
4.2	Validation Accuracy Graph	25
4.3	Validation Loss Graph	26

LIST OF TABLES

TABLE NO	TABLE NAME	PAGE NO
4.1	Confusion Matrix of VGG-16	27
4.2	Confusion Matrix of MobileNet	27
4.3	Confusion Matrix of our Proposed CNN	27

LIST OF ABBREVIATIONS

CNN	–	C onvolutional N eural N etwork
COVID-19	–	C orona V irus D isease 2019
SARS	–	S evere A cute R espiratory S yndrome
MERS	–	M iddle E ast R espiratory S yndrome
RNA	–	R ibonucleic A cid
DNA	–	D eoxyribonucleic A cid
ACE2	–	A ngiotensin C onverting E nzyme 2
RT-PCR	–	R everse T ranscription P olymerase C hain R eaction
VGG	–	V isual G eometry G roup

CHAPTER 1

INTRODUCTION

1.1 CORONA VIRUS

On New Year's Eve 2019, when China informed the world about a bunch of pneumonia cases of unknown origin in Wuhan City, Hubei Province, the corona virus epidemic came to light. The disease rapidly spread to more Chinese provinces and eventually all over the world. It has now been declared as a pandemic by the World Health Organization. The virus was named SARS-CoV-2, hence the disease is known as COVID19. The outbreak spread quickly all around the world. Aged individuals and others with existing medical conditions, such as diabetes, cancer, and respiratory illness are most likely to experience severe illnesses. They may require immediate medical attention and a ventilator system to make breathing easier if the patient's condition is critical.

COVID-19 is an infectious virus that can infect humans. Bats tend to be the COVID-19 virus reservoir, according to phylogenetic analysis using available full genome sequences, but the intermediate host(s) has yet to be identified. Though three major areas of work are already underway in China to raise our understanding of the outbreak's pathogenic origin. These include early investigations into cases with symptoms that occurred near Wuhan in December 2019, as well as ecological sampling from the Huanan Wholesale Seafood Market and other nearby markets, and the gathering of comprehensive information on the point of origin and nature of wildlife species sold on the Huanan market, as well as the animals' final destinations once the market has closed.

Corona viruses (CoV) are a diverse group of viruses that can cause anything from a common cold to more chronic conditions. A novel corona virus (nCoV) is a new strain that has not been previously identified in humans. The new virus was given the name "COVID-19 virus". For about 18,73,858 laboratory-confirmed cases recorded as of 2nd week of April 2020, the maximum number of cases (77.8%) was between 30 and 69 years of age. Among the recorded cases, 21.6% are farmers or employees by profession, 51.1% are male and 77.0% are Hubei.

However, there are still a lot of worries about the new coronavirus. Despite the fact that it appears to be transmitted to humans by animals, it is critical to distinguish between individual animals and other sources, the direction of transmission, the incubation period, and the characteristics of the susceptible population as well as the survival rate. Despite this, there is currently very little clinical information about COVID-19 disease, and data on age ranges, the virus's animal origin, incubation time, epidemic curve, viral spectroscopy, transmission pathogenesis, autopsy findings, and any clinical responses to antivirals are missing among the serious cases.

1.2 TYPES OF COVID

Corona viruses are naturally divided into four groups that cause gastrointestinal and respiratory infections: Gammacoronavirus, Deltacoronavirus, Betacoronavirus, and Alphacoronavirus. The first two types primarily affect birds, while the third and fourth types primarily affect mammals. Human CoVs have been classified into six groups. These include the Betacoronaviruses HCoVHKU1, HCoV-OC43, Middle East Respiratory Syndrome corona virus (MERS-CoV), Severe Acute Respiratory Syndrome corona virus (SARS-CoV), HCoV229E, and HCoV-NL63, as well as the Alphacoronaviruses HCoV229E and HCoV-NL63. Corona viruses did not cause widespread concern until the 2003 SARS pandemic, which was followed by the MERS outbreaks in 2012 and, most recently, the COVID-19 outbreaks. SARS-CoV and MERS-CoV are highly pathogenic viruses that spread from bats to palm civets and dromedary camels, and then to humans.

COVID-19 is transmitted by dust particles and fomites when the infector and the infected person are in near proximity. COVID-19 has not been identified in the air, and it is not considered to be a significant transmission engine based on scientific evidence; however, it is possible if such aerosol-generating activities are used in medical facilities. The active virus has been identified in a limited number of clinical studies, and faeces spreading has been observed in some patients. Furthermore, it does not appear that the faecal-oral route is a COVID-19 transmission engine; its role and importance for COVID-19 must be determined.

1.3 STRUCTURE OF COVID

Viruses are parasitic organisms that can only replicate within cells. They are therefore distinct from bacteria and fungi, which reproduce themselves and are found in soil, water, organic wastes, sewage, and within species.

Animal and plant viruses are divided into two groups: those with long DNA molecules in their genetic material and those with RNA molecules in their genetic material. Herpes, Adenoviruses, and wart viruses are all DNA viruses. Corona viruses, including influenza viruses, HIV, and rhinoviruses, use RNA molecules to encode their genomes, as do influenza viruses, HIV, and rhinoviruses (common cold). The virus that causes COVID-19, SARS-CoV-2, infects mammals and birds. It's quite similar to the viruses that caused the SARS (Severe Acute Respiratory Syndrome) and MERS (Middle East Respiratory Syndrome) outbreaks in the past.

Long RNA polymers are tightly packed into the middle of the corona virus particle, which is surrounded by a protective capsid, which is a lattice of repeated protein molecules known as coat or capsid proteins. These proteins are known as nucleocapsids in the corona virus (N). An outer membrane envelope consisting of lipids (fats) with proteins incorporated surrounds the corona virus core particle. These membranes come from the cells where the virus was last assembled, but they've been changed to include unique viral proteins like the spike (S), membrane (M), and envelope (E).

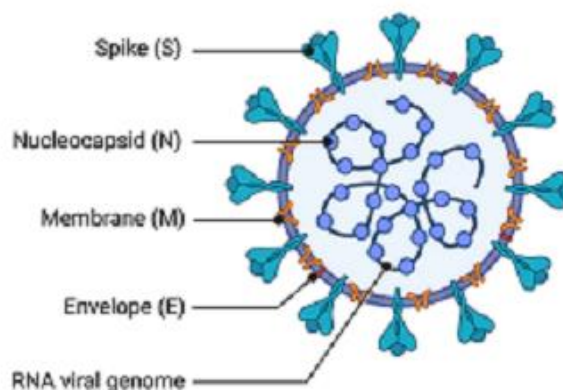


Fig: 1.1 – Structure of COVID

Spike proteins are a group of proteins in the outer membrane that protrude from the particle (S). These proteins bind to receptor proteins on the infected cells, allowing them to infect them.

Corona virus particles are easily inactivated (killed) when exposed to 70% ethanol or 90% isopropanol (rubbing alcohol), hypochlorite bleach, hydrogen peroxide solutions, soaps and detergents, UV light, and high cooking temperatures.

Corona viruses mainly infect human lung cells through an enzyme called Angiotensin Converting Enzyme 2 receptor (ACE2). ACE2 is a member of the angiotensin converting enzymes (ACE) family, which also includes ACE, for which many Americans take blood pressure medications that work by inhibiting ACE. The virus spike protein recognizes and binds to the ACE2 receptor as the first step in viral infection. After that, the virus is absorbed into lung cells, and viral RNA is released into the cytoplasm. The viral RNA molecules instruct the cells to synthesise hundreds of thousands of nucleocapsid, membrane, envelope, and spike proteins while also recruiting the cellular apparatus to produce thousands of copies of the viral RNA. These clump together to form new virus particles, which bud out of the cell membrane. The freshly developed viral particles are released by the cells, which spread the infection and ultimately die.

1.4 DIAGNOSIS OF COVID

COVID-19 signs typically occur 2 to 14 days after the virus is first transmitted to the body. In the early stages of infection, some people show little or no symptoms of disease, but they may still spread the virus to others. In mild cases, self-quarantine and home treatment may be all that is needed to completely recover and prevent the virus from spreading to others. However, in some cases, more complex medical procedures are needed.

Polymerase chain reaction (PCR) testing is still the most popular COVID-19 diagnostic process. When severe acute respiratory syndrome (SARS) first emerged in 2002, this was the type of test used to diagnose it.

To collect a sample for this test, a healthcare provider will likely perform one of the following:

- swab your nose or the back of your throat
- aspirate fluid from your lower respiratory tract
- take a saliva or stool sample

Researchers then extract nucleic acid from the virus sample and use reverse transcription PCR (RT-PCR) to amplify sections of its genome. This effectively allows them a greater sample size to compare viral strains. Within the SARS-CoV-2 genome, there are two genes.

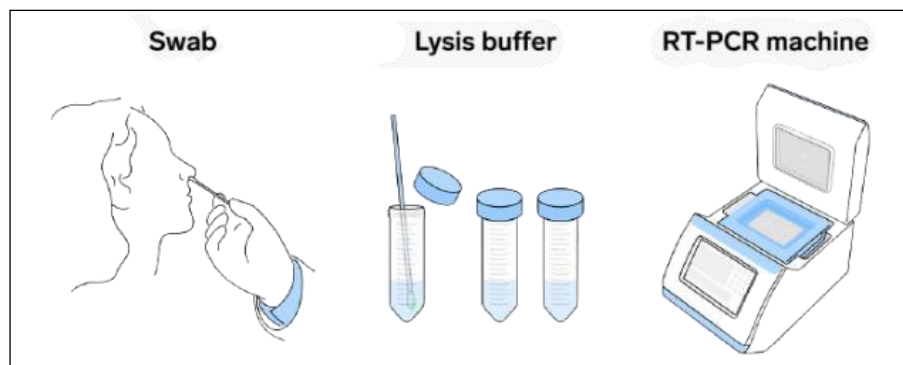


Fig: 1.2 – RT-PCR Test

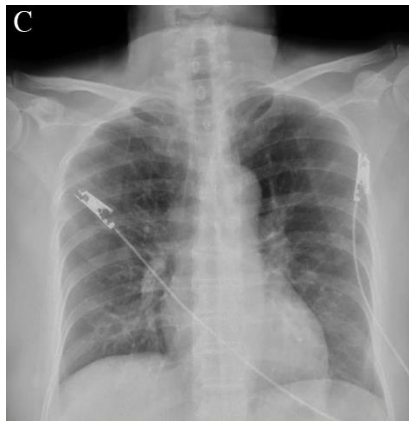
Test results are:

- positive if both genes are found
- inconclusive if only one gene is found
- negative if neither gene is found

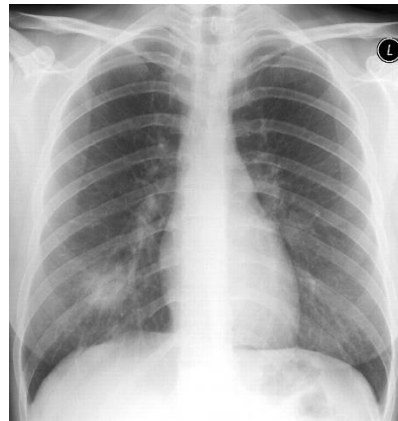
RT-PCR samples are often checked in batches at locations other than the ones from which they were obtained. This means that test results could take a day or more to arrive.

1.5 ROLE OF X-RAY IMAGES IN DIAGNOSING COVID 19

The projection of X-ray scans have PA and AP view. The standard view used for chest X-ray is PA view. If patients can't able to take scan in PA view, AP view is an alternative for PA view. So we use PA view chest X-rays in this work.



PA View



AP View

Fig: 1.3 – Types of views in X-ray image

The density of the lungs rises in covid-19 pneumonia, as it does in other pneumonias. This can be seen on radiography as whiteness in the lungs, which obscures the usual lung markings depending on the severity of the pneumonia; however, this can be delayed or absent.

A ground glass pattern appears when lung marks are partly blurred by increased whiteness. This can be subtle, and a radiologist might be required to confirm it.

Ground glass opacity can be accompanied by peripheral, coarse, horizontal white lines, bands, or reticular changes that can be identified as linear opacities.



Fig: 1.4 – Ground Glass Opacity

Consolidation occurs when the marks on the lungs are entirely lost due to the whiteness (this is usually seen in severe disease). In a small case series in Korea, it was discovered that in those with radiological anomalies and polymerase chain

reaction (PCR) reported covid-19 infection, 70% of the radiographical opacities (number of lesions, not patients) were consolidation.

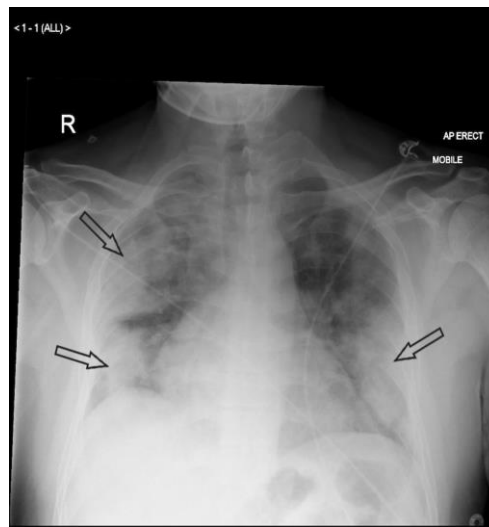
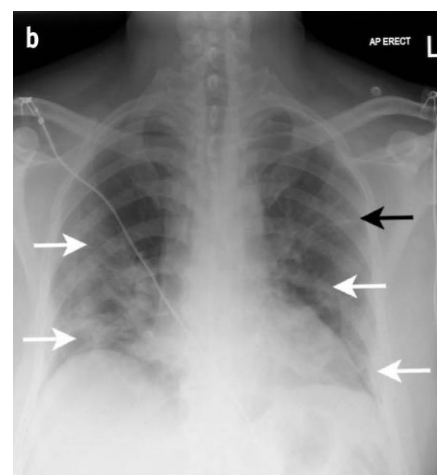
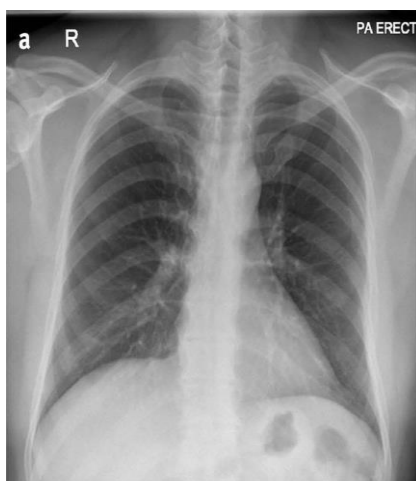


Fig: 1.5 - Consolidation

Because of the radiographic appearances of multifocal ground glass opacity, linear opacities, and consolidation, Covid-19 pneumonia is classified as an atypical pneumonia. Some atypical pneumonias, like other coronavirus infections, show similar improvements (severe acute respiratory system, SARS, and Middle East respiratory syndrome, MERS).



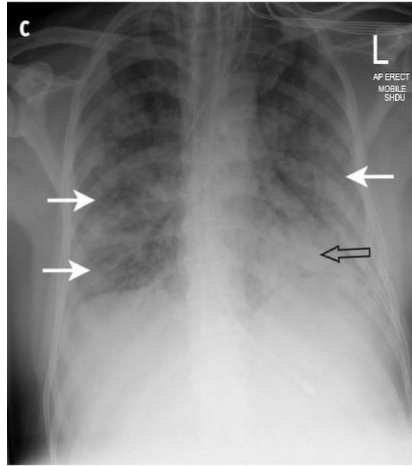


Fig: 1.6 – Serial Radiological progression seen with COVID-19 pneumonia

To detect COVID-19 patients cost-effectively and to get the results in less than 10 minutes, X-rays can be used, said Dr. Vishal Rao, who is a member of the consultative group for COVID and a surgical expert. Dr. Rao who started the project said that an artificial intelligence system can be used for analyzing the X-ray, in which a computerized method can scan for the abnormal area that will show the relevant results.

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

In [9], according to The New Indian Express' report, Dr. Rao, a member of the COVID consultant group said that for diagnosing COVID-19 patients immediately and cost-efficiently X-rays can be used for better results. With this as an idea, we've proposed a CNN model which could predict COVID-19 with a higher accuracy rate.

In [2], the authors developed a deep learning model to predict COVID-19 using patients' CT images. Using deep learning, they have trained a neural network to diagnose COVID-19 using the patients' CT images. They have attained an accuracy of 76% using their model.

The authors of [5] proposed the parallelization concept in their research paper. From this, the parallel convolution neural network concept is selected for this proposed work.

In [1] the proposed structure is divided into three stages. To begin, features are extracted from CT scans using AlexNet, a Convolutional Neural Network (CNN). Second, a proposed features selection algorithm based on Stochastic Fractal Search (SFS), Guided Whale Optimization Algorithm (Guided WOA), is used, followed by feature balancing. Finally, a proposed voting classifier, Guided WOA based on Particle Swarm Optimization (PSO), combines the predictions of various classifiers to determine the most voted class.

The existing artificial intelligence-based models for COVID-19 detection and the challenges these systems face are addressed in [13]. Several deep learning architectures, such as ResNet, Inception, Googlenet, and others, are used to detect COVID-19. A thorough analysis of these approaches can be found here.

Qiu H [12] compiled a large data set of X-ray and CT scan images from various sources and used deep learning and transfer learning algorithms to create a simple and efficient COVID-19 detection system.

The importance of medical image in controlling and treating COVID-19 disease is demonstrated in a succinct review by Dong et al. [14]. One of the most important points raised is that an automated tool can be a useful complement to the laboratory-based real-time polymerase chain reaction (RT-PCR) test.

Harsh Sharma in [11] deployed four types of models to explain the efficiency of adding dropout layers. These dropout layers play a major role in preventing overfitting.

To characterize the various manifestations of COVID-19 and community-acquired pneumonia (CAP) infections, a dual sampling attention network [16] was proposed. A lung mask was used to eliminate the image context of non-lung areas in chest CT to focus on the lungs.

Sammy V. Militante in [4] proposed a VGG-16 model in their research to predict COVID-19 and pneumonia on X-ray scan images using the Convolutional Neural Network method. With this model, they have obtained 95% accuracy.

Abdel-Basset et al. used the x-ray segmentation technique to extract similar small regions that could be COVID-19 features in [18]. They used an improved marine predator's algorithm to create a hybrid COVID-19 detection model (IMPA).

In the work done by Wu et al. [3], they proposed a model of weakly supervised CNN which detects and classifies COVID-19 with the help of CT scans and it has achieved 96.2% accuracy.

CHAPTER 3

METHODOLOGY

3.1 DATA COLLECTION

The first and foremost step in deep learning is data collection. We have used two datasets to apply the proposed model. One of which is the Covid Chest X-Ray dataset from GitHub consisting of covid positive X-Ray scan images. The second dataset is the COVID-19 Radiography Database dataset which consists of Covid and normal X-ray images.

In the first dataset, we are going to do get only the positive images of covid 19. It has the csv file called metadata which consists of all the information such as patient age, gender, finding, modality, view, etc. In that we are going to collect the PA view of the x-ray images move them into a separate folder as PA images only have important features. From the PA folder we are going to collect the COVID affected images and move them into COVID 19 positive folder.

3.1.1 PA PROJECTON OF X-RAY

A chest X-ray is a painless and healthy way to see the structures of the chest wall, including the collar bone, breastbone, ribs, shoulder blade, and associated muscles. It also assists in the visualization of the lungs, airways, heart, food tubing, and diaphragm, which are all internal organs of the chest. A small amount of radiation is used to capture the image on a special X-ray film. The X-ray picture is monochrome. Dense materials, like bones, appear white, while weaker body tissues, like skin and muscles, appear darker. The actual radiation exposure period is typically less than a second, but the entire test can take up to 15-20 minutes.

3.1.2 ADVANTAGES OF PA VIEW

- ❖ The poster o anterior (PA) view is the standard frontal chest projection
- ❖ The x-ray beam traverses the patient from posterior to anterior
- ❖ It is performed standing and in full inspiration with the patient hugging the detector to pull the scapulae laterally
- ❖ It is the best general radiographic technique to examine the lungs, bony thoracic cavity, mediastinum and great_vessels

- ❖ Technically excellent visualization of the mediastinum and lungs, with accurate assessment of heart size

3.1.3 SEGREGATION OF IMAGES

In the second dataset, the images are already categorized into covid and normal. So, we move the covid images to covid 19 positive folder and normal images to covid 19 negative folders. We have totally 1341 images in negative folder and 1396 images in positive folder.

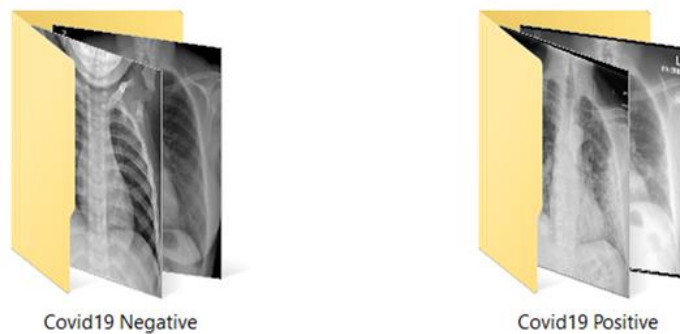


Fig: 3.1 – Covid Positive and Negative folders

We segregate the images in different folders for easier understanding of deep learning model and to speed up the process. So the data is now ready for preprocessing.

3.2 DATA PREPROCESSING

After splitting datasets into positive and negative categories, we convert them into grayscale images using convert color command in opencv. Then the converted images are resized into a common size of 100*100. We resize the images to 100*100 to speed up the process. Larger size of images makes the process to take high amount of time.

After modifying them, we store the modified data into a Numpy array as .npz. We have two numpy arrays for the images and labels. We set the label as 0 for covid negative and 1 for covid positive. The images are stored in data.npz array and labels are stored in target.npz array. We normalize all the images by dividing the entire data array by 255 to get a range from 0 to 1. Then we reshape the data array into 4 dimensional in order to train the neural network. Then the target array is converted into categorical form.

3.2.1 Numpy Array

Numpy (Numerical Python) is an open source Python library that is widely used in science and engineering. It's the de facto standard in Python for working with numerical data, and it's at the heart of the scientific Python and PyData ecosystems. Numpy users range from amateur coders to seasoned experts performing cutting-edge scientific and industrial research and development. Pandas, SciPy, Matplotlib, scikit-learn, scikit-image, and most other data science and scientific Python packages make extensive use of the Numpy API.

Multidimensional array and matrix data structures are included in the NumPy library. It provides methods for efficiently operating on ndarray, a homogeneous n-dimensional array object. NumPy can be used to perform a wide range of array-based mathematical operations. It extends Python with powerful data structures that ensure fast calculations with arrays and matrices, as well as a large library of high-level mathematical functions that work with these arrays and matrices.

3.2.2 CATEGORICAL FORM

Categorical data is information that has a small number of potential values. Furthermore, the data in the group does not have to be numerical; it may be textual. Machine learning models are all mathematical models that require numbers to work. One of the key reasons we need to pre-process categorical data before feeding it to machine learning models is because of this.

3.3 TRAINING THE CNN MODEL

The training phase is where the deep learning algorithms are used. We have tried 3 different algorithms for training the model. The best algorithm with more accuracy rate and less validation loss is chosen for predicting the result. The three algorithms are following:

- ❖ VGG 16
- ❖ MobileNet
- ❖ Parallel CNN

3.3.1 VGG 16

K. Simonyan and A. Zisserman of the University of Oxford proposed VGG16 as a Convolutional neural network model in their paper "Very Deep Convolutional Networks for Large-Scale Image Recognition." In ImageNet, a dataset with over 14 million images divided into 1000 groups, the model achieves 92.7 percent top-5 test accuracy. It was one of the famous model submitted to ILSVRC-2014. It was a well-known model that was submitted to the 2014 ILSVRC. It outperforms AlexNet by sequentially replacing large kernel-size filters (11 and 5 in the first and second Convolutional layers, respectively) with multiple 3x3 kernel-size filters. The NVIDIA Titan Black GPUs were used to train VGG16 for weeks.

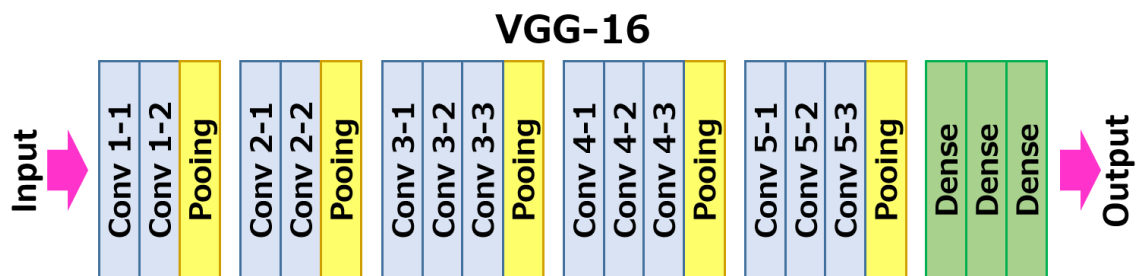


Fig: 3.2 – Layers of VGG-16

3.3.1.1 ARCHITECTURE OF VGG 16

The input to the conv1 layer is a 224 x 224 RGB image with a fixed size. The image is passed through a stack of Convolutional (conv.) layers with a very limited receptive field: 3x3, which is the smallest size to capture the concepts of left/right, up/down, and middle. It also uses 11 convolution filters in one of the settings, which can be thought of as a linear transformation of the input channels. The convolution stride is set to 1 pixel; the spatial padding of conv. layer input is set to 1 pixel for 33 conv. layers so that the spatial resolution is maintained after convolution. Five max-pooling layers follow some of the conv. layers and perform spatial pooling. Max-pooling is achieved with stride 2 over a 2x2 pixel window.

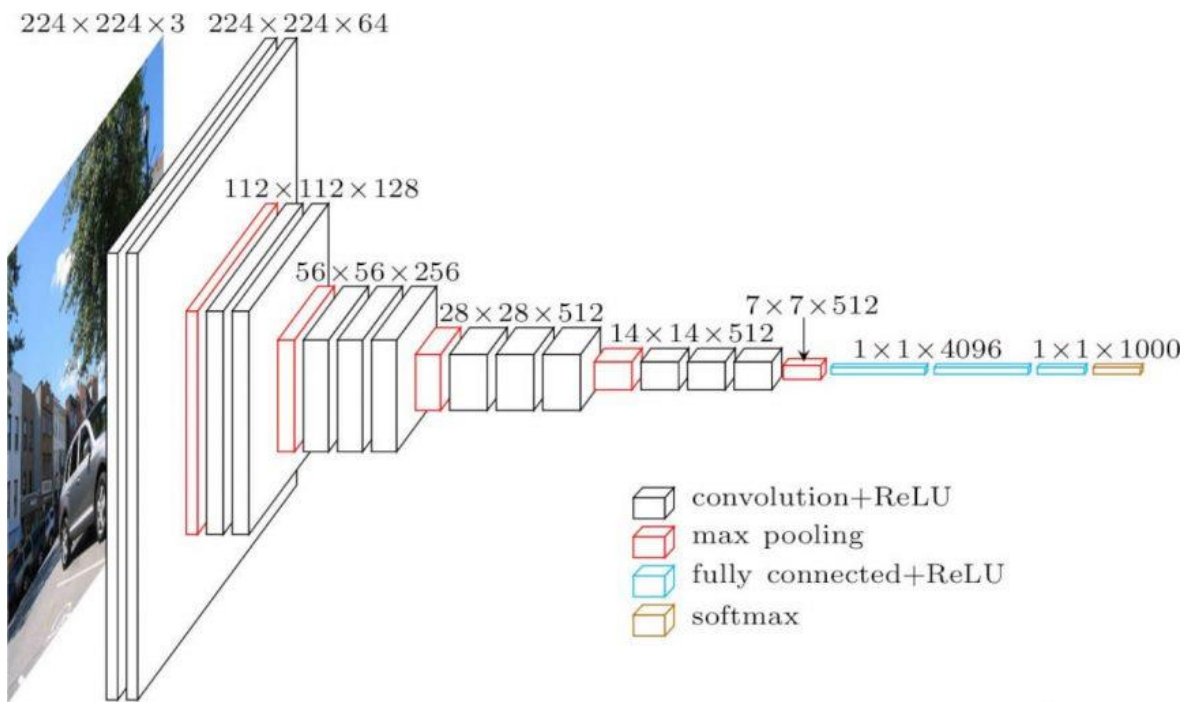


Fig: 3.3 – Architecture of VGG-16

Following a stack of convolutional layers (of varying depth in different architectures), three Fully-Connected (FC) layers are added: the first two have 4096 channels each, while the third performs 1000-way ILSVRC classification and thus has 1000 channels (one for each class). The soft-max layer is the final layer. In all networks, the completely connected layers are configured in the same way.

The rectification (ReLU) non-linearity is present in all hidden layers. It should also be noted that, with the exception of one, none of the networks contain Local Response Normalization (LRN), which does not boost performance on the ILSVRC dataset but increases memory usage and computation time.

3.3.1.2 FEATURES OF VGG 16

- It is also called the OxfordNet model, named after the Visual Geometry Group from Oxford.
- Number 16 refers that it has a total of 16 layers that has some weights.
- It only has Conv and pooling layers in it.
- It always uses a 3 x 3 Kernel for convolution.

- 2x2 size of the max pool.
- It has a total of about 138 million parameters.
- Trained on ImageNet data
- It has an accuracy of 92.7%.
- It has one more version of it called the VGG19, with a total of 19 layers with weights.

3.3.1.3 DISADVANTAGES OF VGG16

Unfortunately, there are two major drawbacks with VGG16:

- ❖ It is painfully slow to train.
- ❖ The network architecture weights themselves are quite large (concerning disk/bandwidth).

3.3.2 MobileNet

The MobileNet model, as its name suggests, is intended for use in mobile applications and is TensorFlow's first mobile computer vision model. Depth-wise separable convolutions are used by MobileNet.

As compared to a network of normal convolutions of the same depth in the nets, it greatly decreases the number of parameters. As a result, lightweight deep neural networks are developed.

A depthwise separable convolution is made from two operations.

- ❖ Depthwise convolution.
- ❖ Pointwise convolution.

3.3.2.1 MOBILENET ARCHITECTURE

A full convolutional layer is the first layer. Batch normalisation and ReLU non-linearity are applied to all layers. The final layer, on the other hand, is a completely connected layer with no non-linearity that feeds the softmax for classification. Strided convolution is used for both depth wise convolution and the first fully

convolutional layer in down sampling. When depth wise and point wise convolution are considered separate layers, MobileNet has a total of 28 layers.

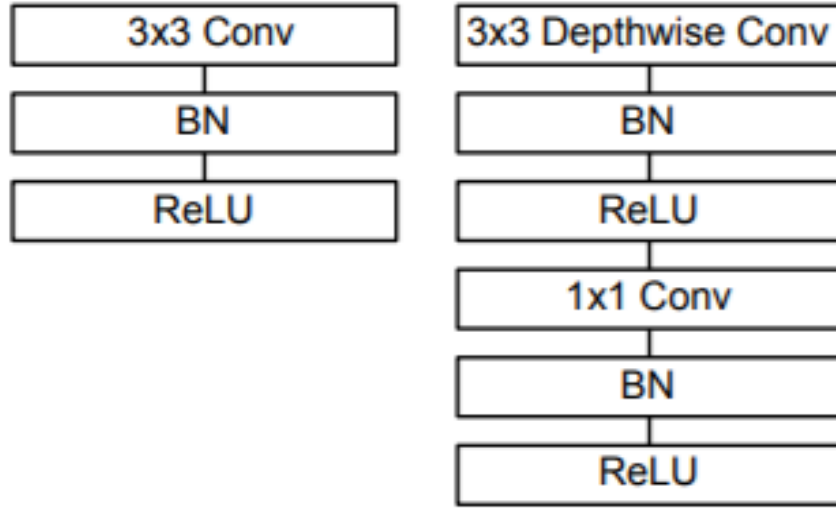


Fig: 3.4 – Layers in MobileNet

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5×	Conv dw / s1	$3 \times 3 \times 512$ dw
	Conv / s1	$1 \times 1 \times 512 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Fig: 3.5 – Architecture of MobileNet

3.3.3 Parallel CNN

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm that can take an input image, assign importance (learnable weights and biases) to various aspects/objects in the image, and discriminate between them. As compared to other classification algorithms, the amount of pre-processing needed by a CNN is significantly less. Although primitive methods require hand-engineering of filters, ConvNets can learn these filters/characteristics with enough preparation.

The architecture of a CNN is inspired by the arrangement of the Visual Cortex and is similar to the communication pattern of Neurons in the Human Brain. Individual neurons can only respond to stimulus in a small area of the visual field called the Receptive Field. A set of such fields can be stacked on top of each other to fill the entire visual field.

LAYERS USED IN THE CNN:

- **Convolution:** Convolutional layers convolve around the image to detect edges, lines, blobs of colors and other visual elements. Convolutional layers hyperparameters are the number of filters, filter size, stride, padding, and activation functions for introducing non-linearity.
- **MaxPooling:** Pooling layers reduces the dimensionality of the images by removing some of the pixels from the image. Maxpooling replaces a $n \times n$ area of an image with the maximum pixel value from that area to downsample the image.
- **Dropout:** Dropout is a simple and effective technique to prevent the neural network from overfitting during the training. Dropout is implemented by only keeping a neuron active with some probability p and setting it to 0 otherwise. This forces the network to not learn redundant information.
- **Flatten:** Flattens the output of the convolution layers to feed into the Dense layers.
- **Dense:** Dense layers are the traditional fully connected networks that maps the scores of the convolutional layers into the correct labels with some activation function (softmax used here)

ACTIVATION FUNCTIONS USED IN CNN:

Activation layers apply a non-linear operation to the output of the other layers such as convolutional layers or dense layers.

- ReLu Activation: ReLu or Rectified Linear Unit computes the function $f(x)=\max(0,x)$ to threshold the activation at 0.
- Softmax Activation: Softmax function is applied to the output layer to convert the scores into probabilities that sum to 1.

Optimizers used in CNN:

- Adam : Adam (Adaptive moment estimation) is an update to RMSProp optimizer in which the running average of both the gradients and their magnitude is used. In practice Adam is currently recommended as the default algorithm to use, and often works slightly better than RMSProp. In my experiments Adam also shows general high accuracy while adadelta learns too fast. I've used Adam in all the experiments because I felt having similar optimizer would be a better baseline for comparing the experiments.

The parallel strategy is to use the two models. The first is the parallel process of the model. The model into a different computing unit. Different machines are responsible for different parts of the network model in a distributed system. The second method is data parallelism. Another machine with multiple copies of each machine's same model is assigned a different data will calculate the result and then join. The arithmetic mean of each layer's parameters and the network is calculated after each training.

For the proposed model, i.e., the CNN model, after data pre-processing, we train the dataset with Convolutional Neural Network. We use the NumPy array data from data pre-processing to train the model. We train a sequential model with the parallel convolutional layer. We apply a parallel 2D convolutional layer to the input layer with 128 filters to the kernels of sizes 3, 5, and 7. We concatenate all the conv2D layers into a single layer with padding as same to get the convolution output size same as the input then reduce spatial dimensions using max pooling.

Then we apply a convolutional layer with 64 and 32 filters to kernel size 3x3. For the activation parameter, we have used Relu to get either 0 or 1, depending on whether the input is negative or not. Then reduce spatial dimensions using max pooling. We flatten the output of the convolutional layers into a single-dimensional array. This is connected to the 128 and 64 neurons dense layer, also called a fully-connected layer. To prevent the overfitting of the training data, we've added dropout layers with a dropout rate of 0.5. We've grouped the dataset into 2 batches by giving *batch_size*=2 since the amount of data is large and requires more storage space. We've given a total of 20 epochs for the training process.

The output layer will be of two neurons for the positive and negative results of COVID-19. So once the model is fully trained, the best model with the least loss and high accuracy is saved by setting *save_best_only=True*. The best model will be saved with an extension of '.h5'. This saved model will be loaded into the web app to predict Covid-19.

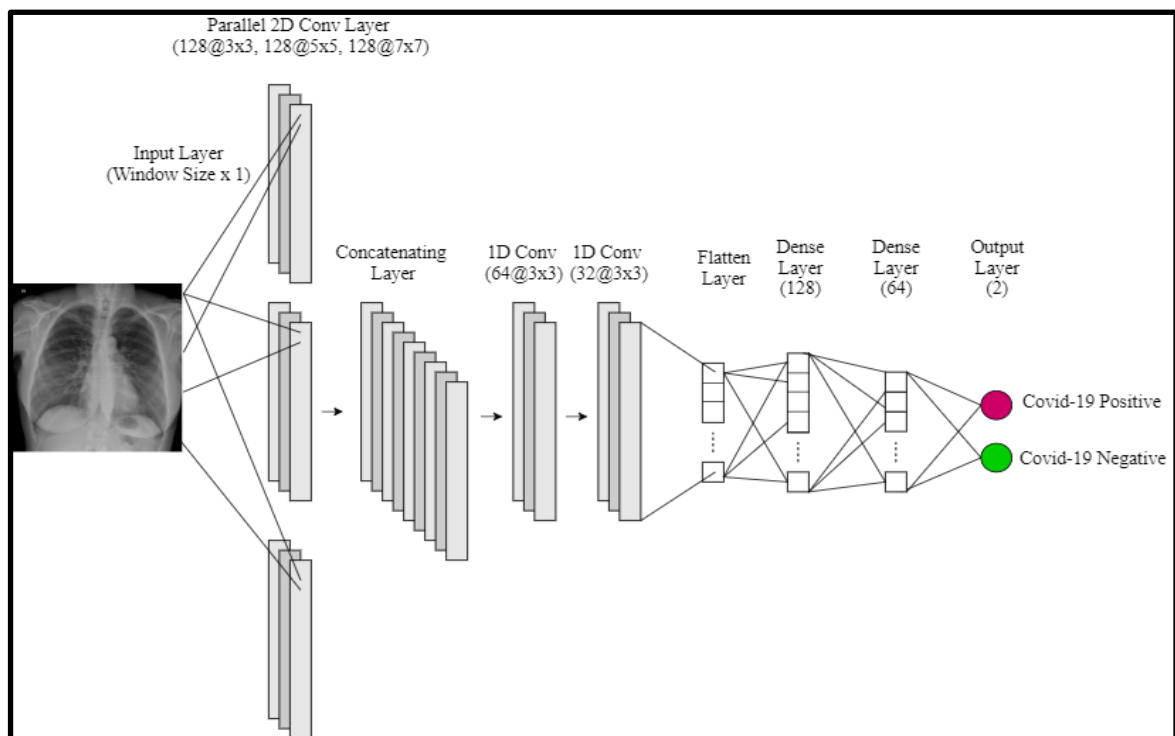


Fig: 3.6 – Architecture of proposed CNN model

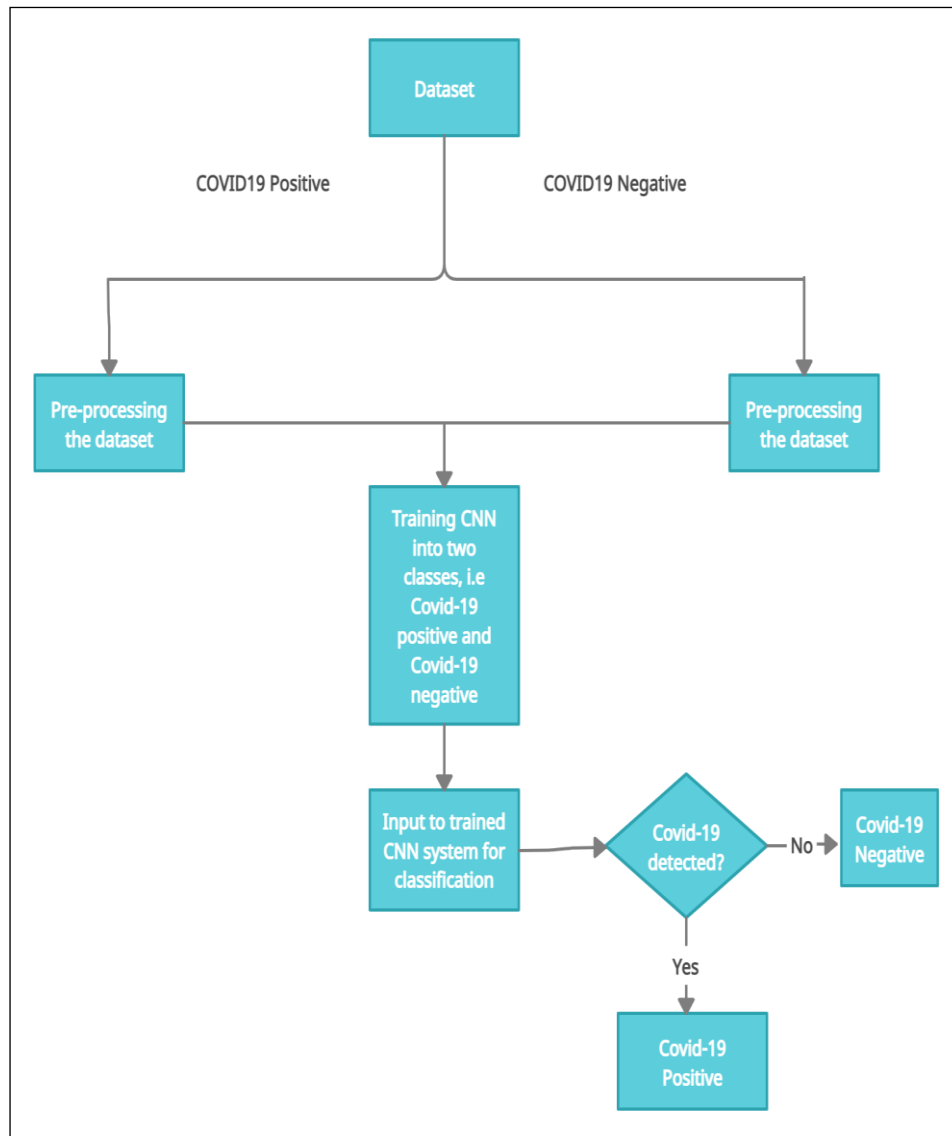


Fig: 3.7 – Dataflow of Proposed CNN

Algorithm:

- i. Preprocess the dataset images and convert them into a numpy array.
- ii. Split the numpy array dataset into training and test data using sklearn.
- iii. Apply CNN filters parallelly for 3, 5 and 7 kernel sizes respectively to extract the features.
- iv. Import layers from tensorflow keras API to create the CNN model with parallelization from scratch and insert the training dataset.
- v. Now train the models for required number of epochs with batch_size=2, until the loss is less than 0.5.

- vi. Cross validate the model using the test data. If the model is overfit or underfit, retrain the model with a different set of layers or different dataset.
- vii. By using model checkpoint, save the best_epoch model which has higher accuracy and less loss rate by specifying save_best_only=True.
- viii. Then load the saved model to the web app for predicting Covid-19.

Model: "sequential"		
Layer (type)	Output Shape	Param #
model (Functional)	(None, 100, 100, 384)	11008
conv2d_3 (Conv2D)	(None, 98, 98, 64)	221248
activation (Activation)	(None, 98, 98, 64)	0
max_pooling2d (MaxPooling2D)	(None, 49, 49, 64)	0
conv2d_4 (Conv2D)	(None, 47, 47, 32)	18464
activation_1 (Activation)	(None, 47, 47, 32)	0
max_pooling2d_1 (MaxPooling2)	(None, 23, 23, 32)	0
flatten (Flatten)	(None, 16928)	0
dropout (Dropout)	(None, 16928)	0
dense (Dense)	(None, 128)	2166912
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
dropout_2 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 2)	130
Total params: 2,426,018		
Trainable params: 2,426,018		
Non-trainable params: 0		

Fig: 3.8 – Model Summary

The model summary given above shows the layers used in our proposed model and their types, namely convolution layer, flatten layer, dense layer, activation and pooling layer.

3.4 WEB APP IMPLEMENTATION

We use flask as a backend and loaded in the model with the least validation loss from our training. We use React to make the front end of the application. React is a JavaScript library for building user interfaces. There are 3 separate components in the webpage that each performs different actions. The main App component manages the state of the app and performs the API requests to the backend. Additionally, there is an ImageSelector component that manages to get the image from the user and communicate it to the App. There is also an Info component that uses the information that the App component got from the API and presents it in a usable way on screen.

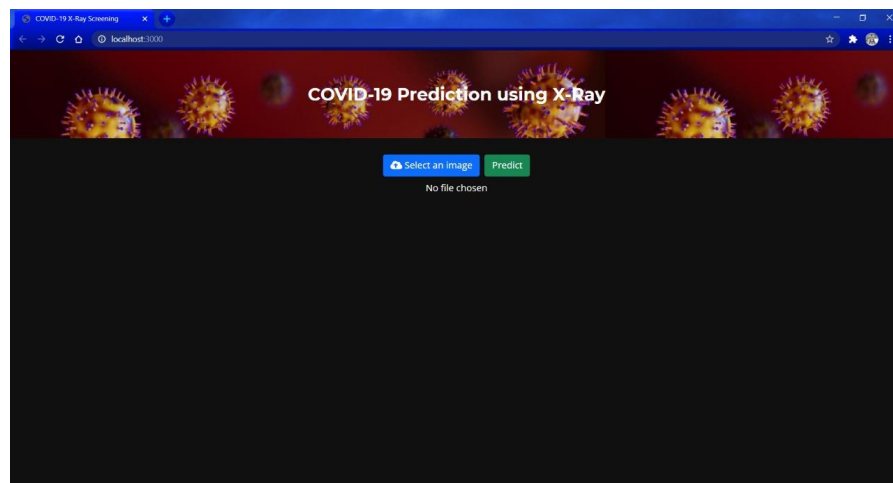


Fig: 3.9 – Output Screen

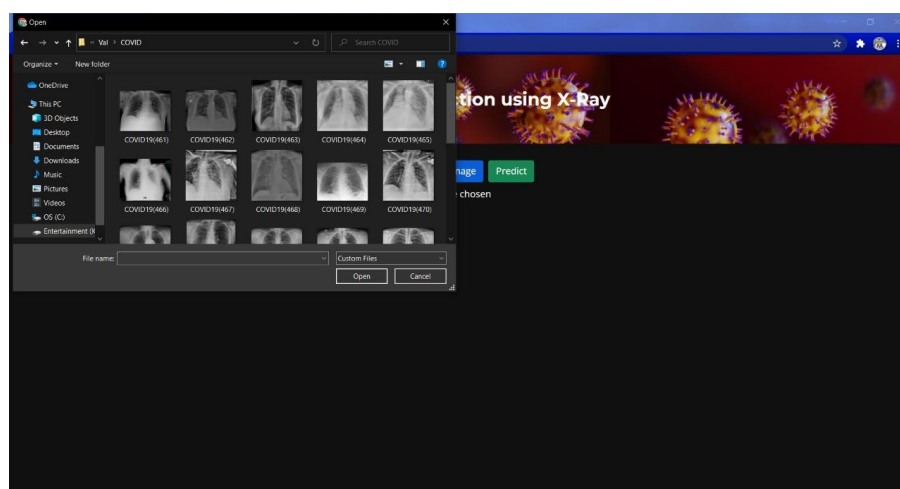


Fig: 3.10 – Selecting Image using ImageSelector

CHAPTER 4

RESULTS AND DISCUSSION

4.1 RESULT

The important goal of this proposed model is to build an effective deep learning model which predicts COVID-19 disease with higher accuracy. Here, in this proposed model, we've collected a total of 2737 X-ray images of COVID and Non-COVID chest X-Rays. The websites GitHub and Kaggle.com were used for retrieving images.

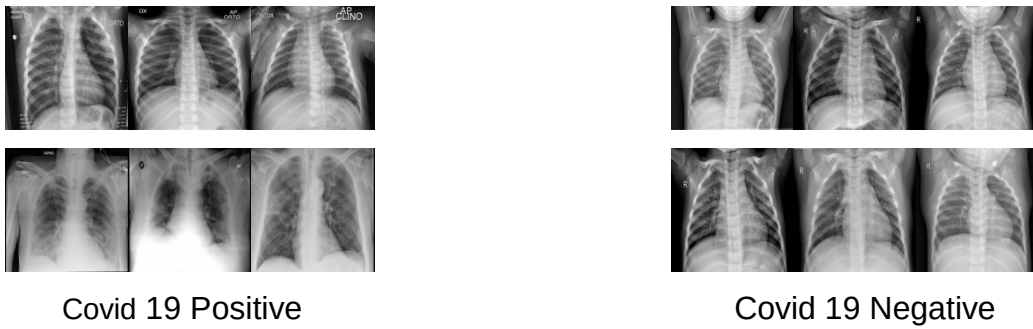
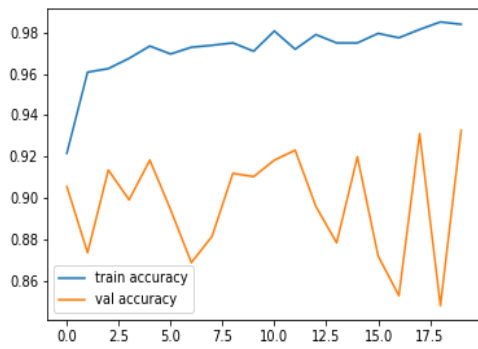


Fig: 4.1 – Sample of Covid positive and negative image

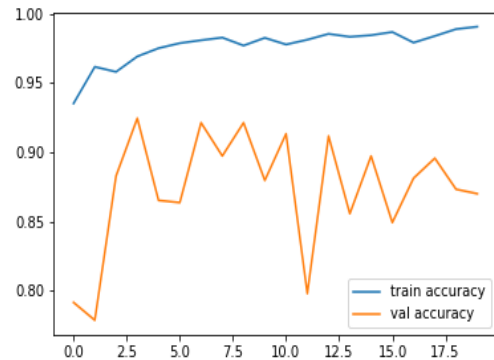
4.2 VALIDATION ACCURACY

We have used three different CNN models. We compared two pre-trained models with our proposed CNN model with a parallelization strategy. The two pre-trained models are VGG-16 and MobileNet. VGG-16 consists of 16 layers with millions of parameters. We imported these two pre-trained models and used them for training purposes separately.

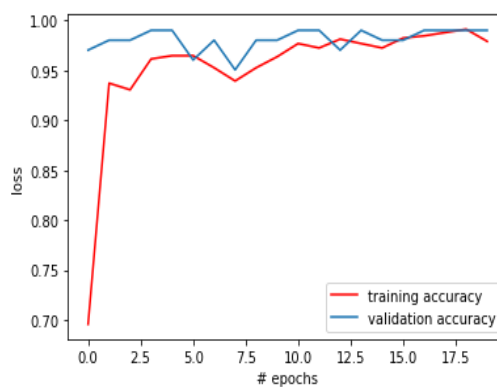
In the below images, the first two images are the accuracy with epoch graph of the pre-trained VGG-16 and MobileNet models. The image at the bottom is the proposed parallel CNN model. This clearly shows that the parallel CNN model has shown a great result with a higher accuracy rate of 98.3%, whereas the best validation accuracy of VGG-16 and MobileNet is 93% and 92% respectively.



VGG-16



MobileNet



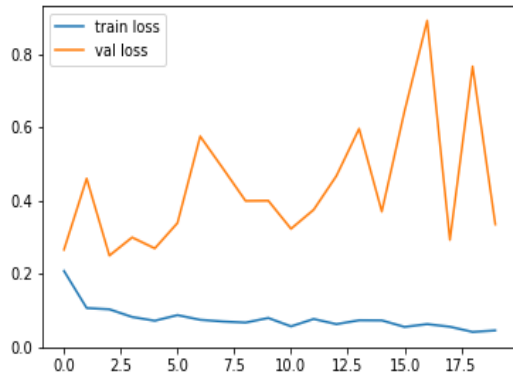
CNN

Fig: 4.2 – Validation Accuracy Graph

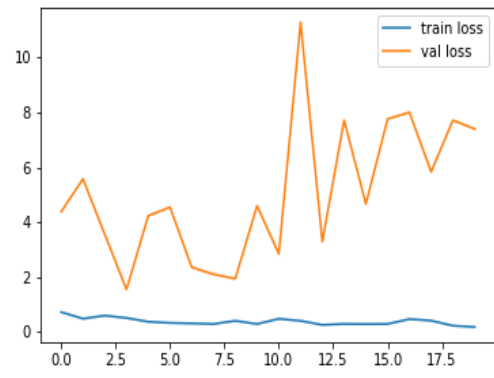
4.3 VALIDATION LOSS

In the below images, the first two images are the loss to the epoch graph of the two pre-trained VGG-16 and MobileNet models. The image at the bottom is the loss graph of our proposed parallel CNN model. This shows that the loss rate of our proposed model is less than the loss rate of the other two pre-trained models.

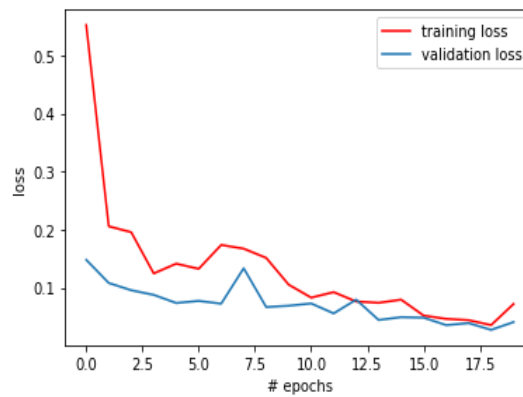
The loss rate of the best epoch in parallel CNN is 0.04 whereas the loss rate of VGG-16 and MobileNet is 0.25 and 1.53 respectively.



VGG-16



MobileNet



CNN

Fig: 4.3 – Validation Loss Graph

For better understanding, a confusion matrix is generated for each convolutional neural network used in the prediction. The confusion matrix is a metric for evaluating machine learning classification results. It is a two-dimensional matrix with four characteristics true positive (TP), false positive (FP), false negative (FN), and true negative (TN). The precision values, accuracy, recall, and F-measure are all calculated using the confusion matrix.

$$\text{Precision} = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Positive})}$$

The above is the formula for calculating precision using confusion matrix.

$$\text{Recall} = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Negative})}$$

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

The above are the formula for calculating recall and f1 using confusion matrix. The following confusion matrix shows the values of precision, recall and f1.

Predicted/ Actual	Covid	Normal	Precision	Recall	F1- score
Covid	TP(29)	FP(1)	0.966	0.906	0.935
Normal	FN(3)	TN(27)	0.900	0.964	0.930

Table: 4.1 – Confusion Matrix of VGG-16

Predicted/ Actual	Covid	Normal	Precision	Recall	F1- score
Covid	TP(27)	FP(3)	0.900	0.931	0.915
Normal	FN(2)	TN(28)	0.933	0.903	0.917

Table: 4.2 – Confusion Matrix of MobileNet

Predicted/ Actual	Covid	Normal	Precision	Recall	F1- score
Covid	TP(30)	FP(0)	1.00	0.967	0.983
Normal	FN(1)	TN(29)	0.966	1.00	0.983

Table: 4.3 – Confusion Matrix of our Proposed CNN

From the above confusion matrix of different models, accuracy can be calculated using the following formula:

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{(\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative})}$$

After calculating accuracy from the above formula, we got an accuracy of 93.3% and 91.6% for VGG-16 and MobileNet respectively which is lesser than that of our proposed CNN model which has an accuracy value of 98.3%.

We use the best model which is saved to predict the X-ray images. In the web app, using the image selector option anyone can upload their X-ray image. By clicking on the predict button, we can get the result as Covid Positive or Covid Negative with probability percentage.

CHAPTER 5

CONCLUSION

5.1 CONCLUSION

COVID-19 detected from the diagnostic test and antibody test takes a lot of time. According to the facts, even RT-PCR, the gold standard diagnostic test, has a high chance of generating false-negative results. As an auxiliary diagnostic method, using AI-based chest X-Ray detection can assist doctors in providing a more thorough patient assessment. Because the symptoms of COVID-19 and all causes of pneumonia overlap, early diagnosis of the disease is still the most difficult decision to make. To solve this, a deep learning-based model was trained in the proposed work to predict COVID-19 with a higher accuracy rate, lower loss rate, and low time duration. For a high-speed and accurate diagnosis of chest X-rays of patients, this model can be used. Primary health workers in remote areas can use this automated COVID-19 detection system. This proposed method can be improved by building a larger dataset of Covid X-rays and X-rays of other lung diseases. It also can be extended to IoT-related work like connecting them to the X-ray machine to predict the COVID-19 in real-time.

REFERENCE:

1. El-Sayed M. El-Kenawy, Abdelhameed Ibrahim, "Novel Feature Selection and Voting Classifier Algorithms for COVID-19 Classification in CT Images", IEEE Access(2020), pp.179317-179335.
2. H. Hui, X.Wu, L. Li, L. Wang, M. Niu, B. He, L. Li, H. Li, X. Yang, J. Tian, and Y. Zha, "Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: A multicentre study," ScienceDirect(2020).
3. S. Hu, Z. Niu, Y. Gao, Y. Jiang, M. Wang, L. Li, X. Xiao, E. F. Fang, J. Xia, H. Ye, W. Menpes-Smith, ad G. Yang, "Weakly supervised deep learning for COVID-19 infection detection and classification from CT images", IEEE Access, vol. 8, pp. 118869-118883, 2020.
4. Sammy V. Militante, Nanette V. Dionisio, Brandon G. Sibbaluca, "Pneumonia and COVID-19 Detection using Convolutional Neural Networks", IEEE Xplore(2020).
5. Sunwoo Lee, Dipendra Jha, Ankit Agrawal, Alok Choudhary, and Wei-Keng Liao, "Parallel Deep Convolutional Neural Network Training by Exploiting the Overlapping of Computation and Communication", IEEE Conference(2017).
6. Mehmet Sevi, İlhan AYDIN, "COVID-19 Detection Using Deep Learning Methods", IEEE Xplore(2020).
7. Furqan Rustam, Aijaz Ahmad Reshi, Arif Mehmood, Saleem Ullah, Byung-Won On, Waqar Aslam, Gyu Sang Choi, "COVID-19 Future Forecasting Using Supervised Machine Learning Models", IEEE Access(2020).
8. Muhammad E. H. Chowdhury, Amith Khandakar, Tawsifur Rahman, Rashid Mazhar, Mamun Bin Ibne Reaz, Muhammad Abdul Kadir, Nasser Al Emadi, Zaid Bin MAhbub, Khandakar Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Mohammad Tariqul Islam, "Can AI Help in Screening Viral and COVID-19 Pneumonia?", IEEE Access(2020).
9. "Chest X-rays can give COVID-19 test results in just 10 minutes: Expert", The New Indian Express.

10. Karen Simonyan, Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", arXiv.org(2015).
11. Harsh Sharma, Jai Sethia Jain, Priti Bansal, Sumit Gupta, "Feature Extraction and Classification of Chest X-Ray Images Using CNN to Detect Pneumonia", IEEE(2020).
12. H. Qiu, J. Wu, L. Hong, Y. Luo, Q. Song, and D. Chen, Clinical and epidemiological features of 36 children with coronavirus disease 2019 (COVID-19) in Zhejiang, China: an observational cohort study. The Lancet Infectious Diseases. 2020.
13. Kang Zhang, Xiahong Liu, Jun Shen, Zhihuan Li, Ye Sang, Xingwang Wu, Yunferi Zha, Wenhua Liang, Chengdi Wang, Ke Wang, Linsen Ye, Ming Gao, Zhongguo Zhou, Liang Li, and others, "Clinically Applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of COVID-19 pneumonia using computed tomography", doi:10.1016/j.cell.2020.04.045, Cell(2020).
14. D. Dong, Z. Tang, S. Wang, H. Hui, L. Gong, Y. Lu, Z. Xue, H. Liao, F. Chen, F. Yang, R. Jin, K. Wang, Z. Liu, J. Wei, W. Mu, H. Zhang, J. Jiang, J. Tian, and H. Li, "The role of imaging in the detection and management of covid-19: a review," IEEE Reviews in Biomedical Engineering, pp. 1–1, 2020.
15. M. Ilyas, H. Rehman, and A. Nat-Ali, "Detection of Covid-19 from chest x-ray images using artificial intelligence: An early review", arXiv(2020).
16. X. Ouyang, J. Huo, L. Xia, et al. "Dual-Sampling Attention Network for Diagnosis of COVID-19 from Community-Acquired Pneumonia", IEEE Transactions on Medical Imaging, 2020.
17. Mugunthan. S. R. "Decision Tree Based Interference Recognition for Fog Enabled IOT Architecture" Journal of trends in Computer Science and Smart technology (TCSST) Vol.02, No. 01 (2020), Pages: 15-25.
18. M. Abdel-Basset, R. Mohamed, M. Elhoseny, R. K. Chakraborty, and M. Ryan, "A hybrid covid-19 detection model using an improved marine

- predators algorithm and a ranking-based diversity reduction strategy," IEEE Access, vol. 8, pp. 79 521–79 540, 2020.
19. Suma. V, and Shavige Malleshwara Hills, "Data Mining based Prediction of Demand in Indian Market for Refurbished Electronics" Journal of Soft Computing Paradigm (JSCP) 2, No. 02 (2020): 101-110.
20. F. Petropoulos and S. Makridakis, "Forecasting the novel coronavirus COVID-19", PLOS ONE, vol. 15, no. 3, Mar 2020.

APPENDICES

A.SAMPLE CODE

```
//Data collection
import pandas as pd
import os
import numpy as np
import cv2
#Positive-Dataset1
datapath1='covid-chestxray-dataset-master'
dataset_path='dataset'
folders=os.listdir(dataset_path)
print(folders)
dataset=pd.read_csv(os.path.join(datapath1,'metadata.csv'))
views=dataset['view']
image_names=dataset['filename']
xray_index=np.where(views=='PA')[0]
xray_image_names=image_names[xray_index]
for xray_image_name in xray_image_names:
    image=cv2.imread(os.path.join(datapath1,'images',xray_image_name))
    try:
        cv2.imwrite(os.path.join('PA',xray_image_name),image)
    except Exception as e:
        print(e)
findings=dataset['finding']
positives_index=np.where(findings=='Pneumonia/Viral/COVID-19')[0]
positive_image_names=image_names[positives_index]
for positive_image_name in positive_image_names:
    image=cv2.imread(os.path.join('PA',positive_image_name))
    try:
        cv2.imwrite(os.path.join(dataset_path,folders[1],positive_image_name),image
    )
    except Exception as e:
        print(e)
```

```

#Dataset2
import shutil
source = "D:\Covid 19 Prediction\COVID-19 Radiography Database\COVID"
destination = "D:\Covid 19 Prediction\dataset\Covid19 Positive"
files = os.listdir(source)
for file in files:
    new_path = shutil.move(f"{source}/{file}", destination)
    print(new_path)

source = r"D:\Covid 19 Prediction\COVID-19 Radiography
Database\NORMAL"

destination = "D:\Covid 19 Prediction\dataset\Covid19 Negative"
files = os.listdir(source)
for file in files:
    new_path = shutil.move(f"{source}/{file}", destination)
    print(new_path)

//Data preprocessing
import cv2
import os
data_path='dataset'
categories=os.listdir(data_path)
labels=[i for i in range(len(categories))]
label_dict=dict(zip(categories,labels)) #empty dictionary
print(label_dict)
print(categories)
print(labels)
img_size=100
data=[]
target=[]

for category in categories:
    folder_path=os.path.join(data_path,category)
    img_names=os.listdir(folder_path)

```

```

for img_name in img_names:
    img_path=os.path.join(folder_path,img_name)
    img=cv2.imread(img_path)

    try:
        gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
        #Coverting the image into gray scale
        resized=cv2.resize(gray,(img_size,img_size))
        #resizing the gray scale into 100x100, since we need a fixed
common size for all the images in the dataset
        data.append(resized)
        target.append(label_dict[category])
        #appending the image and the label(categorized) into the list
(dataset)

    except Exception as e:
        print('Exception:',e)

import numpy as np
#To get a range from 0 to 1 divide value by 255
data=np.array(data)/255.0
#Reshape the array to 4D for training
data=np.reshape(data,(data.shape[0],img_size,img_size,1))
target=np.array(target)

from keras.utils import np_utils

new_target=np_utils.to_categorical(target)
np.save('data',data)
np.save('target',new_target)

//Training the CNN
import numpy as np

```

```

data=np.load('data.npy')
target=np.load('target.npy')
from keras.models import Sequential,Model
from keras.layers import Dense, Dropout, Flatten
from keras.layers import Conv2D,Activation,MaxPooling2D
from keras.layers import Concatenate
from keras import Input
from keras.callbacks import ModelCheckpoint

input_shape=data.shape[1:] #50,50,1
inp=Input(shape=input_shape)
convs=[]
parrallel_kernels=[3,5,7]

for k in range(len(parrallel_kernels)):
    conv = Conv2D(128,
        parrallel_kernels[k],padding='same',activation='relu',input_shape=input_shape,
        strides=1)(inp)
    convs.append(conv)

out = Concatenate()(convs)
conv_model = Model(inputs=inp, outputs=out)

model = Sequential()
model.add(conv_model)

model.add(Conv2D(64,(3,3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Conv2D(32,(3,3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())

```

```

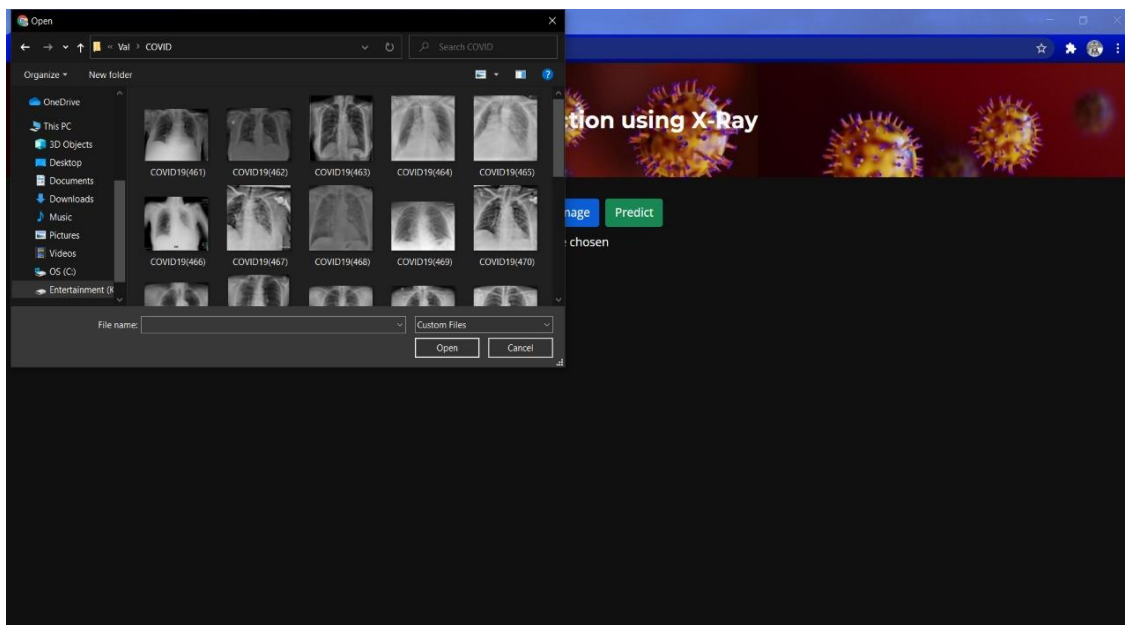
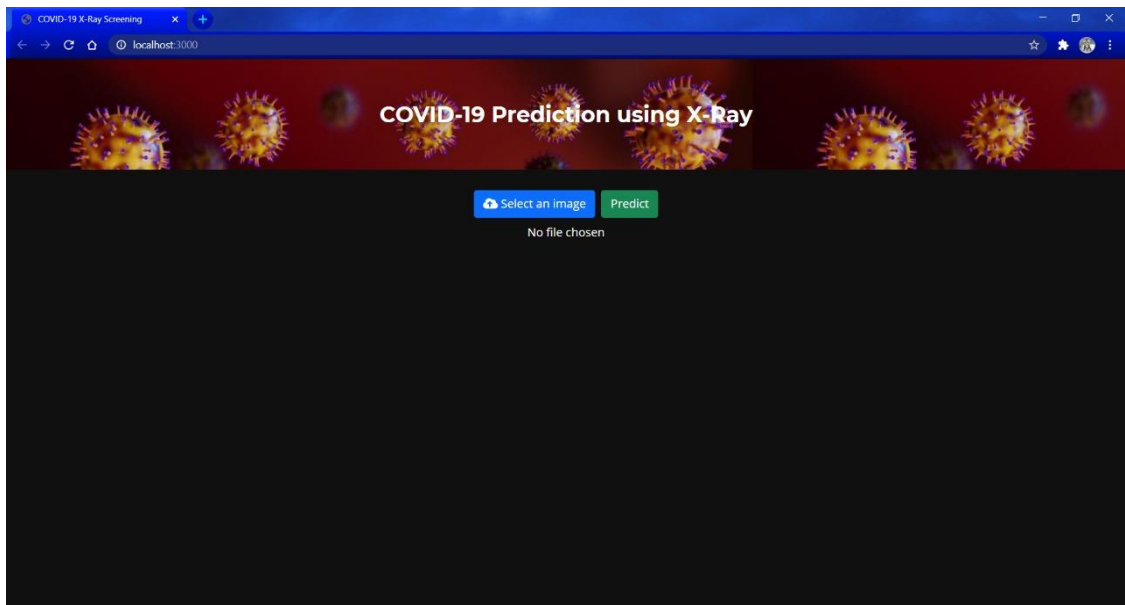
model.add(Dropout(0.5))
model.add(Dense(128,activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64,activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(2,input_dim=128,activation='softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])

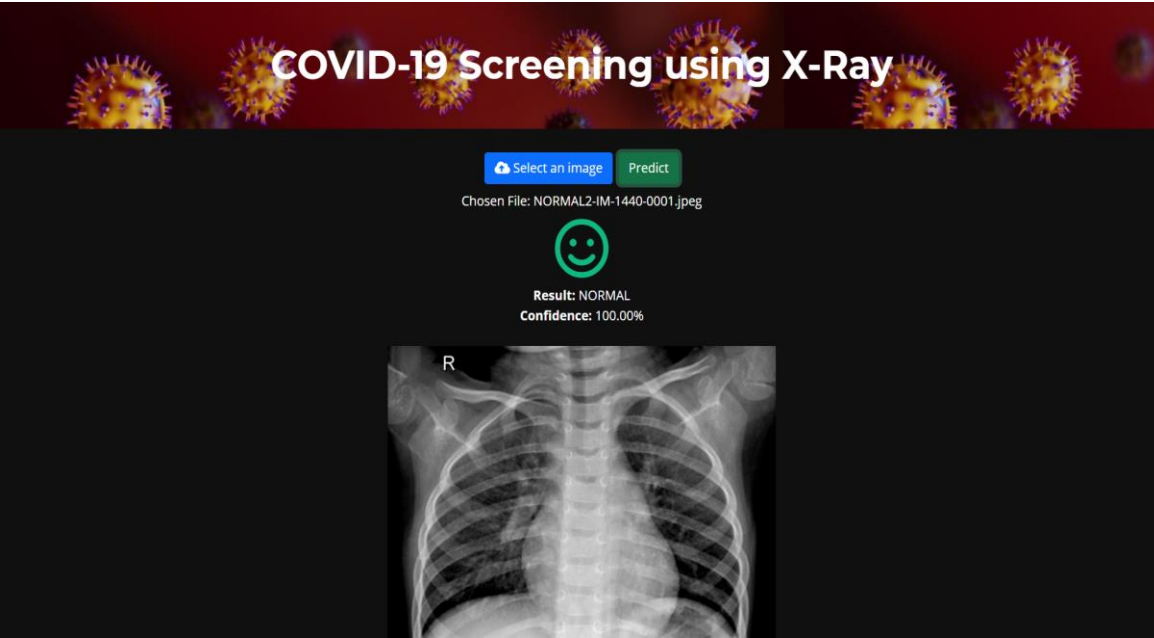
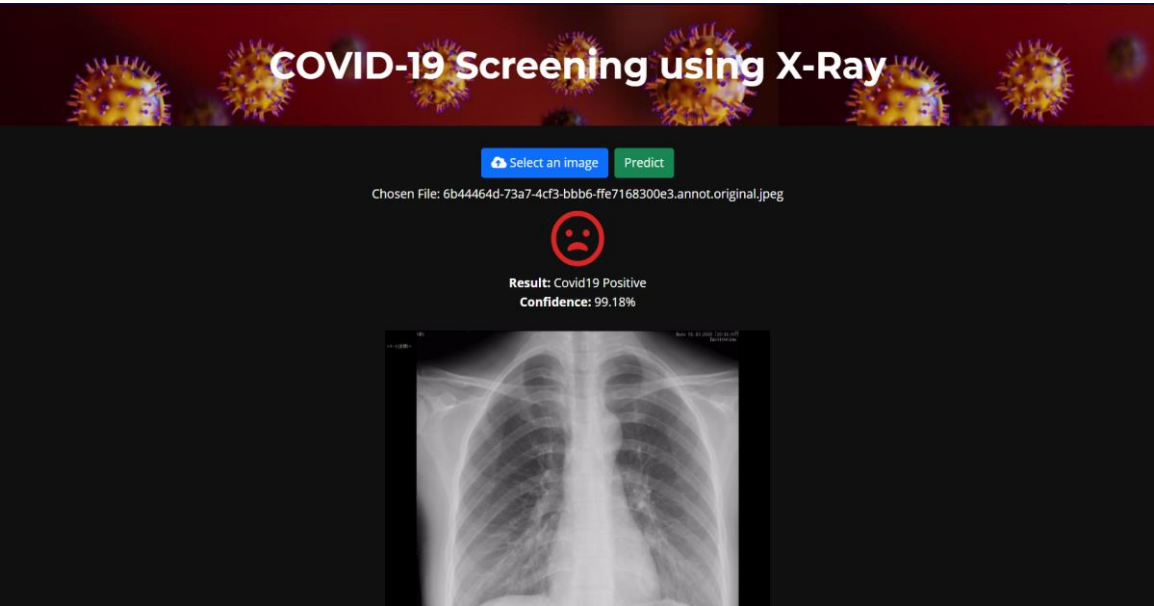
model.summary()
from sklearn.model_selection import train_test_split

train_data,test_data,train_target,test_target=train_test_split(data,target,test_size=0.1)
checkpoint = ModelCheckpoint('model-{epoch:03d}.h5',monitor='val_loss',verbose=0,save_best_only=True,mode='auto')
history=model.fit(train_data,train_target,epochs=10,batch_size = 2,callbacks=[checkpoint],validation_split=0.1)
from matplotlib import pyplot as plt
plt.plot(history.history['loss'],'r',label='training loss')
plt.plot(history.history['val_loss'],label='validation loss')
plt.xlabel('# epochs')
plt.ylabel('loss')
plt.legend()
plt.show()
plt.plot(history.history['accuracy'],'r',label='training accuracy')
plt.plot(history.history['val_accuracy'],label='validation accuracy')
plt.xlabel('# epochs')
plt.ylabel('loss')
plt.legend()
plt.show()
print(model.evaluate(test_data,test_target))

```

B.SCREENSHOTS





PUBLICATION STATUS

Submitted to ICOEI 2021 Conference

Conference Date: June 3 to 5

Acceptance Letter:

 ICOEI 2021	5th International Conference on Trends in Electronics and Informatics (ICOEI 2021) 3-5, June 2021 icoei.com/2021 icoei.conf@gmail.com
<u>Letter of Acceptance</u>	
To G. Aparna, Dr. S. Gowri, R. Bharathi, Department of Information Technology, Sathyabama Institute of Science and Technology Chennai, India.	
Greetings!!	
On behalf of the SCAD College of Engineering and Technology Tirunelveli, we take great pleasure in inviting you to attend the "5th International Conference on Trends in Electronics and Informatics (ICOEI 2021)" which is going to be held during 3-5, June 2021 at Tirunelveli, India.	
We are glad to inform you that your submitted article entitled "COVID-19 Prediction using X-Ray Images" [Paper id:ICOEI-064] has been accepted for publication at ICOEI 2021. We welcome you to join us and share your research and views. This conference will offer you an unforgettable experience in exploring new opportunities.	
We look forward to seeing you at Tirunelveli, India.	
With Thanks,	
Yours' Sincerely	
 Dr. P. Ebby Darney, Principal, SCAD College of Engineering and Technology, Tirunelveli, TamilNadu, India.	
Proceedings by  ICOEI Previous Publications 2020 2019 2018 2017	