ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

AUTOMATIC PREDICTION OF CORONA VIRUS USING CHEST X-RAY AND CT-SCAN

A.Suraj kumar[1], D.Jhansi Lakshmi[2], B.Lakshmi Prasanna[3], B.Narasimha raju[4], V.S.Tejaswi [5].
1 Professor, Dept. of CSE, NS Raju Institute of Technology Visakhapatnam, Andhra Pradesh, India
2,3,4,5 Student, Dept of CSE, NS Raju Institute of Technology Visakhapatnam, Andhra Pradesh, India
sursj.cse@nsrit.edu.in, jhansilakshmi256@gmail.com, lakshmiprasannabaliji@gmail.com, naveenkumar15698@gmail.com, tejaswivadlamani@gmail.com.

Abstract
Currently, the detection of Corona virus disease 2019 (COVID-19) is one of the main challenges in the world, given the rapid spread of the disease. Recent statistics indicate that the number of people diagnosed with COVID-19 is increasing exponentially, with more than 1.6 million confirmed cases; the disease is spreading to many countries across the world. We analyze the incidence of COVID-19 distribution across the world. We present an artificial-intelligence technique based on a deep convolutional neural network (CNN) to detect COVID-19 The chest CT-based COVID-19 classification of the suspected patients requires radiologists and considerable amounts of their times as the number of COVID-19 suspects increases at a rapid rate. Moreover, it has been found that the COVID-19-infected patients show some patterns on chest CT images that are not easily detectable by the human eye Therefore, an automatic detection tool is much needed to assist in screening. To develop an artificial-intelligence tool based on a deep convolutional neural network (CNN) which can examine chest X-ray images to identify such Covid patients which be available quickly and at low cost.

Keywords: Automatic Prediction, Corona Virus and Chest X-Ray and Ct-Scan

1. Introduction
Corona virus Disease 2019 (COVID-19) is a highly contagious disease that spreads from one person to another with the appearance of respiratory distress [1]. It has been spread all over the world since December 2019 and so far, it has infected more than millions of people. The clinical tests like reverse transcription-polymerase chain reaction (RT-PCR) are usually used for classifying the suspected patients, but medical imaging techniques such as computed tomography (CT) has also been used to detect and evaluate COVID-19. The chest CT-based COVID-19 classification of the suspected patients requires radiologists and considerable amounts of their time as the number of COVID-19 suspects increases at a rapid rate. Moreover, it has been found that the COVID-19-infected patients show some patterns on chest CT images that are not easily detectable by the human eye. Therefore, an automatic detection tool is much needed to assist in screening COVID-19 pneumonia using chest CT imaging [3,4].

Cite this article as: A.Suraj kumar, D.Jhansi Lakshmi, B.Lakshmi Prasanna, B.Narasimha raju & V.S.Tejaswi., "Automatic Prediction of Corona Virus Using Chest X-Ray and CT-Scan", International Journal of Research in Advanced Computer Science Engineering, (IJRACSE), Volume 6 Issue 11, April 2021, Page 31-43.

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Like many other methodological innovations, artificial intelligence (AI) has been applied to the timely, rapid, and effective diagnosis of COVID-19 using chest CT images. AI-Driven tools may provide automated and fast approaches for the detection and classification of COVID-19 on chest CT.

In this study, we developed a deep learning-based Convolutional Neural Network (CNN) model to classify COVID-19 cases from healthy cases. The preprocessing steps consisted of transforming image intensities into the Hounsfield unit, extracting the lung part by image processing techniques, equalizing histograms with a transformation created by the intensities of the infection regions, and stretching the contrast of the images. Then, after training and evaluating several deep learning networks, such as ResNet, Inception, VGG16, Dense Net, and Exception by the images of 80% of the cases in each group, VGG16 was chosen as our base model, since it is less complicated compared to the other models. The two-class output was obtained using pooling, dropout, and dense layers. Finally, our model was tested by the CT images of 20% of the cases in each group. The results showed 89.78% precision, 91.50% sensitivity, 88.66% specificity, 0.9063 F1-Score, and 90.14% accuracy.

1.1 Deep Learning:

Deep learning is an artificial intelligence (AI) function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural network.

- Deep learning is an AI function that mimics the workings of the human brain in processing data for use in detecting objects, recognizing speech, translating languages, and making decisions.
- Deep learning AI is able to learn without human supervision, drawing from data that is both unstructured and unlabeled.
- Deep learning, a form of machine learning, can be used to help detect fraud or money laundering, among other functions.

How Deep Learning Works

Deep learning has evolved hand-in-hand with the digital era, which has brought about an explosion of data in all forms and from every region of the world. This data, known simply as big data, is drawn from sources like social media, internet search engines, e-commerce platforms, and online cinemas, among others. This enormous amount of data is readily accessible and can be shared through fintech applications like cloud computing. However, the data, which normally is unstructured, is so vast that it could take decades for humans to comprehend it and extract relevant information. Companies realize the incredible potential that can result from unraveling this wealth of information and are increasingly adapting to AI systems for automated support. Deep learning unravels huge amounts of unstructured data that would normally take humans decades to understand and process.

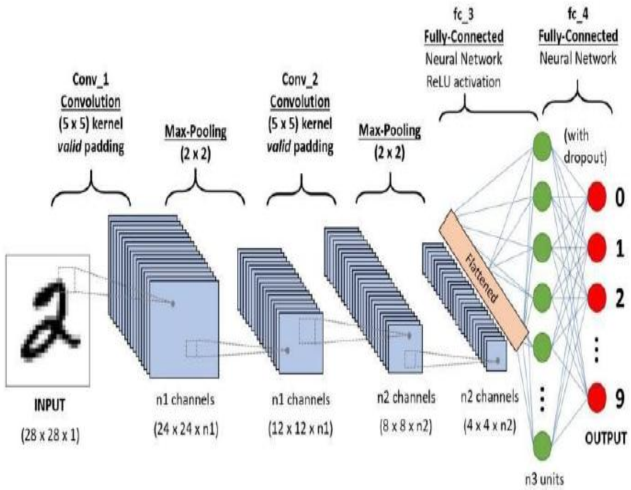
ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

A Deep Learning Example

Using the fraud detection system mentioned above with machine learning, one can create a deep learning example. If the machine learning system created a model with parameters built around the number of dollars a user sends or receives, the deep-learning method can start building on the results offered by machine learning. Each layer of its neural network builds on its previous layer with added data like a retailer, sender, user, social media event, credit score, IP address, and a host of other features that may take years to connect together if processed by a human being. Deep learning algorithms are trained to not just create patterns from all transactions, but also know when a pattern is signaling the need for a fraudulent investigation. The final layer relays a signal to an analyst who may freeze the user's account until all pending investigations are finalized. Deep learning is used across all industries for a number of different tasks. Commercial apps that use image recognition, open-source platforms with consumer recommendation apps, and medical research tools that explore the possibility of reusing drugs for new ailments are a few of the examples of deep learning incorporation.

1.2 Neural Networks

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another. Artificial neural networks (ANNs) are comprised of a node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network. Neural networks rely on training data to learn and improve their accuracy over time. However, once these learning algorithms are fine-tuned for accuracy, they are powerful tools in computer science and artificial intelligence, allowing us to classify and cluster data at a high velocity. Tasks in speech recognition or image recognition can take minutes versus hours when compared to the manual identification by human experts. One of the most well-known neural networks is Google's search algorithm.



2. System Analysis

2.1 Existing System

In Most of the Existing System a support vector machine (SVM) model for the classification of patients with COVID-19 and

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

other types of pneumonia using CT chest images. This was done by extracting textural and histogram features of the infections and obtaining a radionics features vector from each sample.

2.1.1 Disadvantages of Existing System

● CXR-based detection of Covid-19 patients is that trained doctors may not be available all the time

● COVID-19 detection accuracy from X-ray images is very less

● Samples are very less to find the detection of disease

2.2 Proposed System

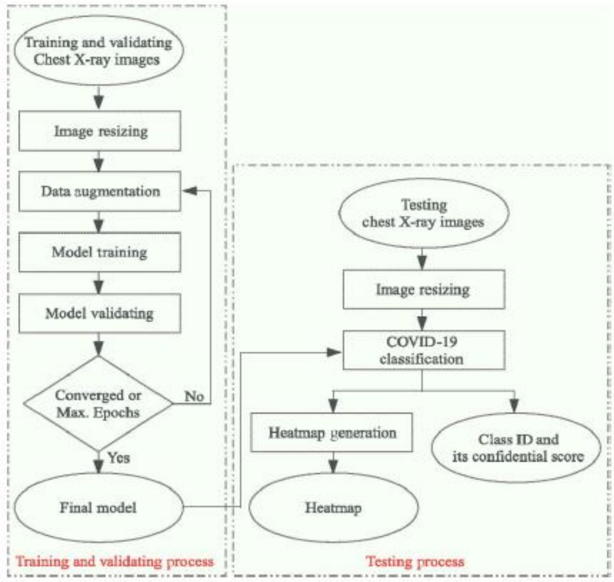
We present an artificial-intelligence technique based on a deep convolution neural network (CNN) to detect COVID19 patients using real-world datasets. Our system examines chest X-ray images to identify such patients. Three forecasting methods—the prophet algorithm (PA), autoregressive integrated moving average (ARIMA) model, and long short-term memory neural network (LSTM) were adopted to predict the numbers of COVID-19 confirmations.

2.2.1 Advantages of Proposed System

● Multiple architures are applied

● Accuracy is better

● We predict the numbers of COVID-19 confirmations, recoveries, and deaths over the next 7days.



3. Requirement Analysis and Specification

3.1 Objective:

To develop an artificial-intelligence tool based on a deep convolutional neural network (CNN) which can examine chest X-ray images to identify such Covid patients which be available quickly and at low costs

3.2 Functional Requirement:

In Software engineering, a functional requirement defines a function of a software system or its components. A function is described as set of inputs, the behavior and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describing all the cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

3.2.1 Functional Requirements for This Project

The functional requirements describe the interaction between the system and its environment independent of its implementation

MODULE 1: Res Net Architecture

Residual neural network (ResNet) is proposed by He et al. in 2015 [HZR16]. The hypothesis behind ResNet is that deeper networks are harder to optimize, since the deeper model should be able to perform as well as the shallower model by copying the learned parameters from the shallower model and setting additional layers to identity mapping. To help optimize deeper models, residual blocks are designed to fit a residual mapping $F(x)$ instead of the desired underlying mapping $H(x)$, and full ResNet architecture is built by stacking residual blocks. More specifically, every residual block has two 3×3 convolutional layers. Periodically, the number of filters are doubled and spatial downsampling is operated. Figure 3.1 illustrates the structure within the residual block shows an example of ResNet architecture - ResNet with 18 layers

4. System Design

4.1 UML Diagrams

UML stands for Unified Modelling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with unified modelling language. The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS: The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular p rogramming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices

4.2 Building Blocks of Uml:

The vocabulary of the UML encompasses 3 kinds of building blocks Things are the data

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

abstractions that are first class citizens in a model. Things are of 4 types Structural Things, Behavioural Things, Grouping Things, notational Things.

4.3 Relationships:

Relationships tie the things together. Relationships in the UML are Dependency, Association, Generalization, Specialization.

4.4 UML Diagrams:

A diagram is the graphical presentation of a set of elements, most often rendered as a connected graph of vertices (things) and arcs (relationships).

There are two types of diagrams, they are:

➔ Structural Diagram

➔ Behavioural Diagrams

Structural Diagrams:

The UML's four structural diagrams exist to visualize, specify, construct and document the static aspects of a system. I can View the static parts of a system using one of the following diagrams.

Structural diagrams consist of Class Diagram, Object Diagram, Component Diagram, and Deployment Diagram.

Behavioural Diagrams:

The UML's five behavioural diagrams are used to visualize, specify, construct, and document the dynamic aspects of a system. The UML's behavioural diagrams are roughly organized around the major ways which can model the dynamics of a system. Behavioural diagrams consists of Use case Diagram, Sequence Diagram, Collaboration Diagram, State chart Diagram, Activity Diagram.

5. Implementation

5.1 MODULE IMPLEMENTATION

5.1.1 Preprocessing

A set of noise-removal functions accompanied with morphological operations that result in clear image of chest ct scan image after passing through high pass filter is the basic idea behind the proposed algorithm. The set of morphological operations used will decide the clarity and quality of the chest ct scan image image After the original image undergoes pre-processing transformations These basic pre-processing transformations include:

● Changing the image to greyscale, as we need to find contour of the final image which works on greyscale images.

● Applying low pass filter, to remove any noise, if present, in the image.

● Applying high pass filter, to obtain sharpened image with clear-defined boundaries.

5.1.2 Segmentation

The K-means clustering algorithm. The main idea of this approach is to assign each point or data to a cluster whose center (centroid) is the closest. The center of each cluster is the average of all the points in this cluster classification we define a class for the black background of the image with cerebrospinal fluid, a second class for white matter and a third class for gray matter.

5.1.3 DEEP LEARNING

In this module we build a model of deep learning architecture which consist 16 and 32 Convolutions layer with filter mask and padding filter and Maxpooling layer with filter

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

mask size of 3*3*16 and 3*3*32. The proposed deep learning network is focused on the chest components (grey and white matter) for healthy and unhealthy cases

Chest abnormalities classification

In this modules All chest components segmented are classified as normal or abnormal cases maccording to the deep learning network,In this chapter, we will discuss the model architecture, transfer learning concept, implementation details, performance evaluation metrics, and the strategy for interpreting model

5.1.4 Res Net ARCHITECTURE

Residual neural network (ResNet) is proposed by He et al. in 2015 [HZR16]. The hypothesis behind ResNet is that deeper networks are harder to optimize, since the deeper model should be able to perform as well as the shallower model by copying the learned parameters from the shallower model and setting additional layers to identity mapping. To help optimize deeper models, residual blocks are designed to fit a residual mapping $F(x)$ instead of the desired underlying mapping $H(x)$, and full ResNet architecture is built by stacking residual blocks. More specifically, every residual block has two 3×3 convolutional layers. Periodically, the number of filters are doubled and spatial downsampling is operated. Figure 3.1 illustrates the structure within the residual block.

6. Sample Code

```
from flask import Flask, render_template, request, session, redirect, url_for, flash
import os
from werkzeug.utils import secure_filename
from tensorflow.keras.models import load_model
import matplotlib.pyplot as plt
import cv2
import numpy as np
UPLOAD_FOLDER = './flask app/assets/images'
ALLOWED_EXTENSIONS = set(['png', 'jpg', 'jpeg', 'gif']) # Create Database if it doesnt exist
app = Flask( name ,static_url_path='/assets', static_folder='./flask app/assets', template_folder='./flask app')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
@app.route('/')
def root():
return render_template('index.html')
@app.route('/index.html')
def index():
return render_template('index.html')
@app.route('/contact.html')
def contact():
return render_template('contact.html')
@app.route('/news.html')
def news():
return render_template('news.html')
@app.route('/about.html')
def about():
return render_template('about.html')
@app.route('/faqs.html')
def faqs():
return render_template('faqs.html')
@app.route('/prevention.html')
def prevention()
return render_template('prevention.html')
@app.route('/upload.html')
```

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

```
def upload():
return render_template('upload.html')
@app.route('/upload_chest.html')
def upload_chest():
return render_template('upload_chest.html')
@app.route('/upload_ct.html')
def upload_ct():
return render_template('upload_ct.html')
@app.route('/uploaded_chest', methods = ['POST', 'GET'])
def uploaded_chest():
if request.method == 'POST':
# check if the post request has the file part
if 'file' not in request.files:
flash('No file part')
return redirect(request.url)
file = request.files['file']
# if user does not select file, browser also # submit a empty part without filename if file.filename == '':
flash('No selected file')
return redirect(request.url)
if file:
# filename = secure_filename(file.filename)
file.save(os.path.join(app.config['UPLOAD_FOLDER'], 'upload_chest.jpg'))
resnet_chest = load_model('models/resnet_chest.h5')
vgg_chest = load_model('models/vgg_chest.h5')
inception_chest = load_model('models/inceptionv3_chest.h5')
xception_chest = load_model('models/xception_chest.h5')
image = cv2.imread('./flask app/assets/images/upload_chest.jpg') # read file
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB) # arrange format as per keras
image = cv2.resize(image,(224,224))
image = np.array(image) / 255
image = np.expand_dims(image, axis=0)
resnet_pred = resnet_chest.predict(image)
probability = resnet_pred[0]
print("Resnet Predictions:")
if probability[0] > 0.5:
resnet_chest_pred = str('%.2f' % (probability[0]*100) + '% COVID')
else:
resnet_chest_pred = str('%.2f' % ((1-probability[0])*100) + '% NonCOVID')
print(resnet_chest_pred)
vgg_pred = vgg_chest.predict(image)
probability = vgg_pred[0]
print("VGG Predictions:")
if probability[0] > 0.5:
vgg_chest_pred = str('%.2f' % (probability[0]*100) + '% COVID')
else:
vgg_chest_pred = str('%.2f' % ((1-probability[0])*100) + '% NonCOVID')
print(vgg_chest_pred)
inception_pred = inception_chest.predict(image)
probability = inception_pred[0]
print("Inception Predictions:")
if probability[0] > 0.5:
inception_chest_pred = str('%.2f' % (probability[0]*100) + '% COVID')
else:
inception_chest_pred = str('%.2f' % ((1-probability[0])*100) + '% NonCOVID')
print(inception_chest_pred)
xception_pred = xception_chest.predict(image)
```

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

```
probability = xception_pred[0]
print("Xception Predictions:")
if probability[0] > 0.5:
xception_chest_pred = str('%.2f' % (probability[0]*100) + '% COVID')
else:
xception_chest_pred = str('%.2f' % ((1-probability[0])*100) + '% NonCOVID')
print(xception_chest_pred)
```

7. System Testing

7.1 Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

7.2 Types of Tests

7.2.1 Unit Testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.2.2 Integrations Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.2.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items: Valid Input : identified classes of valid input must be accepted. Invalid Input : identified classes of invalid input must be rejected. Functions : identified functions must be exercised. Output : identified classes of application outputs must be exercised. Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined

ISSN No : 2454-4221 (Print)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined

7.2.4 System Testing:

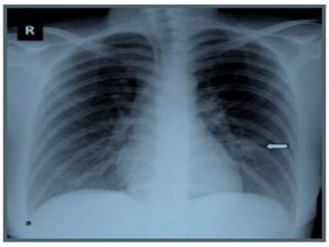
System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7.2.5 White Box Testing:

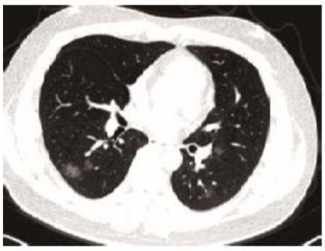
White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

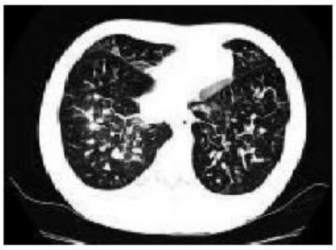
7.2.6 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated as a black box you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.


chest x-ray of covid +ve person

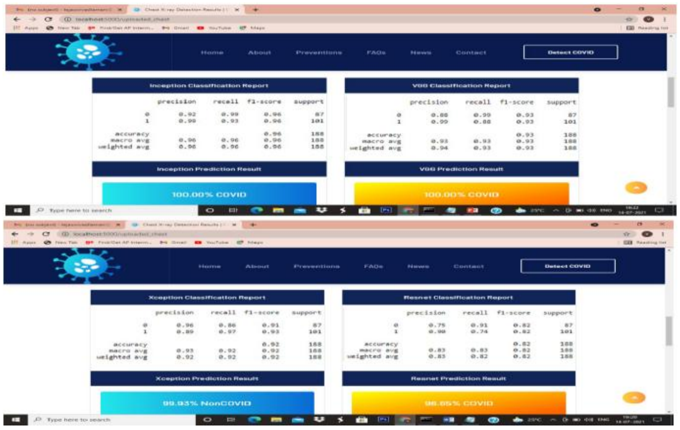

chest x-ray of covid –ve person


CT scan of Covid +ve person

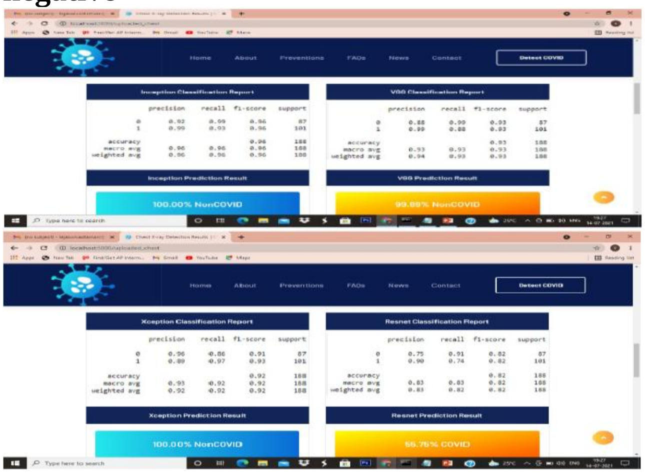

CT scan of covid –ve person

8. Output Screenshots

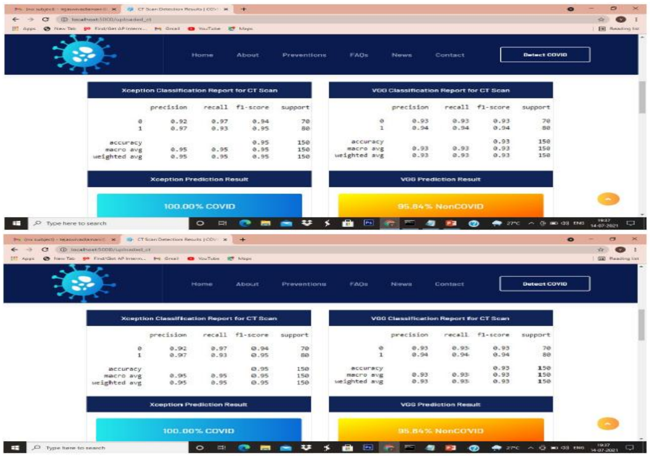
Output For Chest X-Ray Which Is Covid Positive



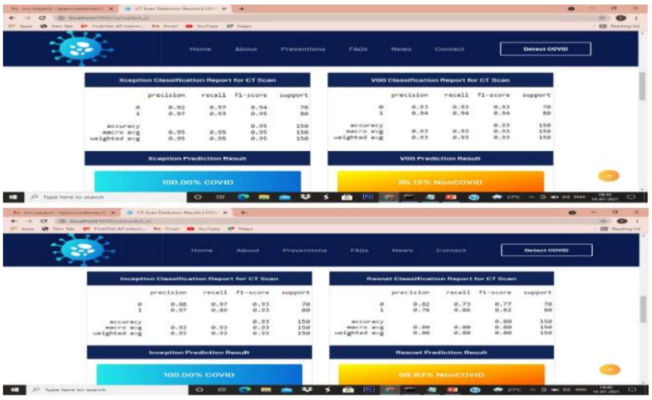
Output for chest x-ray which is covid negative



Output for CT-scan which is covid positive



Output for CT-scan which is covid negative



Conclusion

In this study, we train two deep learning models based on transfer learning concept and Res Net18 architecture for COVID-19 pneumonia detection in chest X-ray images. The first one is a binary classification model that aims to separate COVID-19 pneumonia and non- COVID-19 cases. It is able to classify all test cases correctly. The second one is a four-class classification model that aims to distinguish COVID-19 pneumonia, viral pneumonia, bacterial pneumonia and normal cases. It reaches an average accuracy, precision, sensitivity, specificity, F1-score, and AUC of 93%, 93%, 93%, 97%, 93%, and 99%, respectively. This model is able to detect all COVID-19 test cases, but it tends to classify normal cases into other classes and tends to classify viral pneumonia images into bacterial pneumonia. To shed a light on how the four-class classification model make predictions of COVID-19 pneumonia cases, we apply Grad-CAM method to generate localized map that highlights important regions the model thinks to make predictions. By comparing the highlighted regions and radiologists' descriptions of chest X-ray images, we find out that our model is more powerful when patchy areas are the main indicators of COVID-19 pneumonia. Otherwise, though our model can make accurate predictions, it tends to be unfocused and fails to highlight specific areas that indicating COVID-19 pneumonia. COVID-19 has already become a global threat and has taken away hundreds of thousands of people's lives. This study shows the feasibility of building a computer-aided diagnostic system that can help clinicians detect COVID-19 pneumonia from radiology images accurately and quickly. Moreover, model interpretation techniques allow us to further evaluate and understand models. However, the limited data adds challenge to the performance of model By collecting more chest X-ray images of COVID-19 pneumonia, other pneumonias and normal cases, the model will be more robust and powerful.

References

1. Abbas A, Abdelsamea MM, Gaber MM (2020) Classification of COVID-19 in chest X-ray images using DeTraC deep

convolutional neural network. arXiv:2003.13815

2. Apostolopoulos ID, Mpesiana TA (2020) COVID-19: Automatic detection from X−Ray images utilizing transfer learning with convolutional neural networks. Physical and Engineering Sciences in Medicine [PMC free article] [PubMed]
3. Bai HX, Hsieh B, Xiong Z, Halsey K, Choi JW, Tran TML, Pan I, Shi LB, Wang DC, Mei J, Jiang XL, Zeng QH, Egglin TK, Hu PF, Agarwal S, Xie FF, Li S, Healey T, Atalay MK, Liao WH (2020) Performance of radiologists in differentiating COVID-19 from Non-Covid-19 viral pneumonia at chest CT. Radiology, 200823 [PMC free article][PubMed]
4. Butt C, Gill J, Chun D, Babu BA (2020) Deep learning system to screen corona virus disease 2019 pneumonia. Applied Intelligence, 1
5. Chen N, Zhou M, Dong X, Qu J, Gong F, Han Y, Qiu Y, Wang J, Liu Y, Wei Y, Yu T, Zhang X, Zhang L. Epidemiological and clinical characteristics of 99 cases of 2019 novel Corona virus pneumonia in Wuhan, China: a descriptive study. *The Lancet.* 2020;395(10223):507–513. doi: 10.1016/S0140-6736(20)30211-7. [PMC free article][PubMed] [CrossRef] [Google Scholar]
6. Cohen JP, Morrison P, Dao L, Roth K, Duong TQ, Ghassemi M (2020) Covid-19 image data collection: Prospective predictions are the future. arXiv:2006.11988
7. El-Shafai W, Abd El-Samie F (2020) Extensive COVID-19 X−Ray and CT Chest Images Dataset, Mendeley Data, v3. 10.17632/8h65ywd2jr.3
8. Elasnaoui K, Chawki Y. Using X-ray images and deep learning for automated detection of corona virus disease. *J Biomol Struct Dyn.* 2020;0:1–22. doi: 10.1080/07391102.2020.1767212. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
9. Fang Y, Zhang H, Xie J, Lin M, Ying L, Pang P, Ji W. Sensitivity of chest CT for COVID-19: comparison to RT-PCR. *Radiology*. 2020;10:200432. [PMC free article][PubMed] [Google Scholar]
10. Gozes O, Frid-Adar M, Sagie N, Zhang H, Ji W, Greenspan H (2020) Corona virus detection and analysis on chest ct with deep learning. arXiv:2004.02640
11. Hassanien AE, Mahdy LN, Ezzat KA, Elmousalami HH, Ella HA (2020) Automatic x-ray covid-19 lung image classification system based on multi-level thresholding and support vector machine. medRxiv
12. Kumar R, Arora R, Bansal V, Sahayasheela VJ, Buckchash H, Imran J, Narayanan J, Pandian N, Raman B (2020) Accurate prediction of COVID-19 using chest X-Ray images through deep feature learning model with SMOTE and machine learning classifiers. medRxiv
13. Li L, Qin L, Xu Z, Yin Y, Wang X, Kong B, Bai J, Lu Y, Fang Z, Song Q, Cao K, Liu D, Wang G, Xu Q, Fang X, Zhang S, Xia J (2020) Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. Radiology

14. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR. Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput Biol Med.* 2020;121:103792. doi: 10.1016/j.compbiomed.2020.103792.[PMC free article] [PubMed] [CrossRef] [Google Scholar]