JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

Detecting Corona Syndrome Using Deep Learning Algorithms

Anil Kumar G [1], Ashwini.G [2], Narendar Singh.D [3], Nagaswetha.G [4], Pavitra.B [5]

[1, 2, 3, 4, 5] *Asst.Prof, Dept. of ECE, Anurag University, Hyderabad, Telangana, India.*

ABSTRACT

Viral pandemics are a serious threat. Corona virus causes respiratory illness in people -Severe Acute Respiratory Syndrome (SARS) corona virus and Middle East Respiratory Syndrome (MERS) corona virus. As of 2nd December, there is a total of 6,38,39,023 cases with 14,80,001 deaths in more than 210 countries across the world. (Source: Bing COVID-19 tracker). By manual testing, the genuine situation can be understood and appropriate decisions can be taken. However, there are drawbacks of manual checking out which include sparse availability of testing kits, costly and inefficient blood tests, extra time consuming; a blood test takes around 5–6 hours to generate the result.This work for the most part acquaints a down to earth approach with assistance surgeons helping us in the battle against novel Covid. In this way, the thought is to overcome these issues utilizing the Deep Learning method for better and effective treatment. We can utilize Convolutional Neural Network to produce the outcomes. This methodology is additional efficient, more affordable and simple to work.

Keywords— CNN: Convolutional Neural Networks,DL: Deep Learning,ReLU: Rectified linear unit, NumPy: Numerical Python, Pandas- Panel Data frames

I. INTRODUCTION

To put it another way, image organisation is where the PC can analyse a picture in order to identify the class it belongs to or the class it is crucial to. A class is only a name. So, for example, if you enter a picture of a cat, Image order is that cycle in which a model or PC looks at that image and determines if it is a feline by chance or by probability. Picture order was based on pixel information in its early stages. This meant that PCs would be able to decompose images into individual pixel-sized images. There's a problem here in that two shots of the same subject taken at different times might have very different results. A variety of foundations, points, and gifts are all possible. An incredible challenge for PCs to organise and categorise images. Deep learning, on the other hand, makes it simple to deal with these problems. Deep learning is a subset of computerised reasoning (AI) that allows computers to learn from data
Computer frameworks called as networks are used in the process of deep learning[1]. There are several ways in which picture characterisation may be used, and it has a lot of promise as a replacement for reliability. Here are just a few examples of why it's important. To figure out what's going on around them, self-driving cars employ a technique called image grouping. Including, but not limited to, trees, people, and traffic lights. In addition, medical services may benefit from the use of picture characterization. Using clinical images as an example, it might determine whether or not the arrangement depicts an indicator of illness. Furthermore, image grouping has become increasingly unavoidable as deep learning has progressed. Deep Learning may be used in medical treatment to identify patterns. This allows computers to anticipate the consequences of certain side effects of medications. This means that Deep Learning can predict whether or not a patient will develop a specific symptom or whether a therapy will be effective. In addition, we were able to build a model that predicts if a person is COVID-19 positive or negative using just a few Deep Learning computations. Prior to testing, the model improvement measure has just a few phases.

DATA PREPARATION

Perhaps the most arduous part of every deep learning endeavour is data preparation[3]. Because each project's central information is unique, every deep learning project is different. Everything about a deep learning project is unique because of this. It's impossible for anybody to tell you what the optimal results are or what calculations you should apply to get them. Think about your models as a whole and determine the optimal computation for your dataset to develop a standard in execution. There are a number of phases involved in the use of deep learning in the classroom. Here are the steps

i. Problem statement:

In this approach, attention is paid on understanding enough about the daring to choose the layout or frame of the gauge assignment. As a matter of fact, what is the most pressing problem in the world right now? It includes obtaining the data that is recognised as important in creating a gauge and explicitly expressing the structure that the assumption will take. It might also include talking to people who are involved in the project or who have a lot of authority in the area. Having an idea of what to anticipate can help you prioritise which data to collect.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

Classification: Answering two-fold Yes/No questions requires a computation (cats or dogs, fortunate or horrible, sheep or goats; you get the picture) or a multiclass demand (grass, trees, or thistles; cats, canines or birds, etc,) It's also important for an evaluation to include the proper responses.

Clustering: An evaluation is required to identify the rules of classification and the number of categories. As opposed to portrayal efforts, your knowledge of social events and the criteria used to divide them is completely lacking. As an example, you may encounter this problem if you have to segment your customers and devise a unique approach for each segment based on its ascribes.

Regression: Regression: In order to arrive at a numerical value, you must do a computation. For example, if you spend a lot of time trying to figure out the right price for your product because it depends on so many factors, relapse calculations may help you figure it out

Ranking: A ranking system based on several highlights is used by certain AI algorithms. Video real-time features and product placement are two examples of how positioning may be utilised to make motion pictures more likely to be purchased based on a customer's previous search and purchase activities.Predicting COVID-19 or not is a binary classification problem and we use binary images to deal with this.

ii. Establish Data Collection Mechanisms:

The process of gathering data may take a long time. So that it may be used with the model, it is necessary to put up a real and accurate dataset. Associations and organisations that are open enough to disclose their data provide the majority of the public datasets. Kaggle and GitHub are two of the most common places to get datasets that include a broad variety of life domains like clinical benefits records, environmental records, transportation evaluation records, language and understanding arrangements, records of hardware usage, and so on. [3].

iii. Format data to make it consistent:
It's a good idea to organise your data in the same way that you organise your report. Furthermore, converting a dataset into a record plan that works best for your AI system is not a huge deal. We're referring to the integrity of the data itself. If your dataset has been physically updated by a variety of persons or if it has been compiled from many sources, it is important to ensure that all elements within a particular feature are accurately assembled

IMAGE PRE - PROCESSING

The use of PC estimates to conduct image maintenance on automated images is known as advanced picture preparation. In the realm of cutting-edge sign planning, electronic image taking care of has several advantages over traditional picture preparation. While mechanised picture taking care is useful for eradicating unfortunate distortions from images, it's also possible that an update of a large picture that includes our AI-Computer Vision models could benefit from this improved data. This would allow us to apply computations of a much broader scope to the data. A picture is nothing more than a collection of two-dimensional integers ranging from 0 to 255. When x and y are equal and vertical, f (x, y) is used to show the mathematical limit of the two co-ordinates. In order to apply pre-getting ready to your dataset, you may use the following methods

i. Read image:

Here, we first create a variable that has a reference to the path to our image collection, and then we create displays that can stack organisers with photographs inside them.

ii. Resize image:

In light of the fact that a few images captured by a camera and included in our AI calculations vary in size, we need to establish a baseline size for all images included in our AI calculations

iii. Remove noise (Denoise):

We do this to remove any unnecessary disturbance from our image. Gaussian blur is used to get this effect. Clouding a picture with a Gaussian limit results in a Gaussian haze (also known as Gaussian smoothing). By and large, it is a frequently used impact in plan programming, often to lessen the visual noise

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

II. IMAGE PROCESSING

2.1.1 IMAGE RESIZING

The vast majority of convolutional neural networks [5] are pre-programmed to only be able to recognise images of a certain size. These two or three annoyances throughout the collecting of information and the delivery of model. Re-shaping information images into manageable associations is a critical strategy in breaking over this barrier. Channel flipping may be detected using images, but the model's accuracy is sacrificed.

2.1.2 FILTERING PROCESS

Learned channels are used by CNN[2] to convolve the previous layer's part maps. There is a spatial link between the piles of loads that make up a channel. The purpose of using a channel less noticeable than the data is to enable the data group to copy a comparison channel (set of burdens) on various occurrences at various spotlights on the information. Specifically, the channel is applied to each covering portion or channel approximated fix of the data from left to right, completely

When imagining channels, you will use these methods. Model.layers may be used to traverse the model's many layers (). [2]

1. In order to acquire the weights and bias values for a convolutional layer, use the function get weights ().
2. Set the filter weights to a range of 0 to 1.
3. For each convolutional layer and channel, draw the filters. There are three channels for RGB in a colour picture. The number of channels in a grayscale picture is 1.

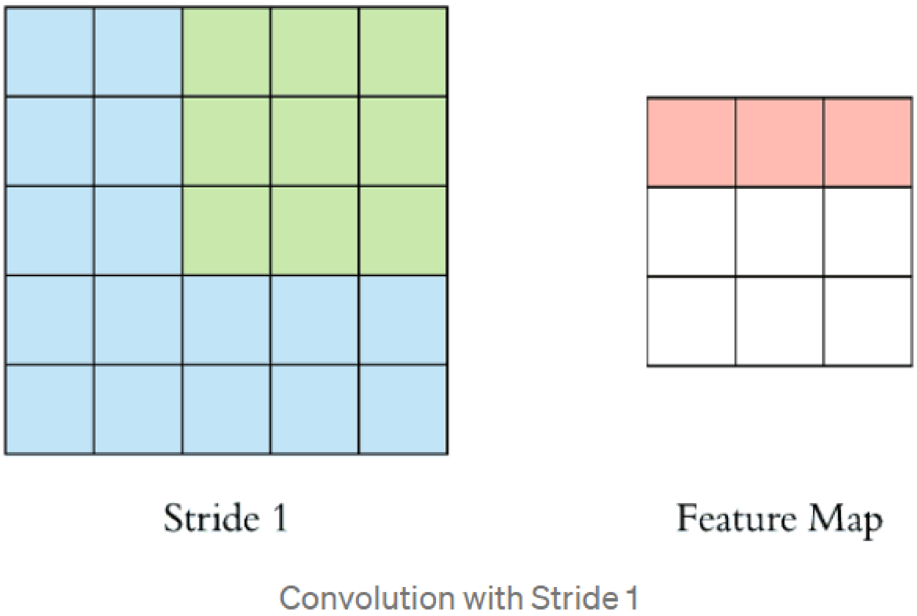

Fig 2.1. Convolution with Stride 1[2]

The number of steps we take in each phase in convolution is symbolised by the number of steps. It's one, of course. An example of the first phase is shown in Fig 2.1, which highlights maps. There is less output than input, as can be seen from the graphs. Cushioning is used to maintain the yield component as an input. Zeros are added uniformly to the information lattice in a cushioning cycle. There are extra faint squares in this illustration that represent padding. As a result, it is used to make the component of yield equal to data.

III. METHODOLOGY

3.1 SEQUENTIAL MODEL

An example of a model found in Keras is the successional model. From contribution to yield, it allows you to easily build successive levels of the organisation all together. Layers are piled on top of each other in a descending order. As illustrated in Fig. 4.7, the successive API allows you to build models layer by layer for the majority of problems

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

```
model = Sequential()
model.add(Conv2D(32,kernel_size=(3,3),activation='relu',input_shape=(224,224,3)))
model.add(Conv2D(128,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(64,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(128,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))
```

Fig 3.1. Sequential Model implementation

3.2 APPLYING 2D CONVOLUTION

The number of output channels is the first parameter supplied to the Conv2D() layer method. 2D convolution layer conv2d builds a convolution kernel that is winded with layers input to produce a tensor of outputs from the convolution process. Different CNN layers are seen in Figure 3.2

```
model = Sequential()
model.add(Conv2D(32,kernel_size=(3,3),activation='relu',input_shape=(224,224,3)))
model.add(Conv2D(128,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))
```

Fig 3.2. Applying 2D Convolution to images

3.3 MAXPOOLING 2D LAYER

There follows a 2D max pool layer. Pooling in the x and y bearings is shown as (2, 2) in this case, as well as the steps. Our model's information display is simplified by picking the largest value across the window specified by pool size for each measurement along the element's hub for our example down-modeling setup Fig. 4.9 depicts the process of maxpooling.

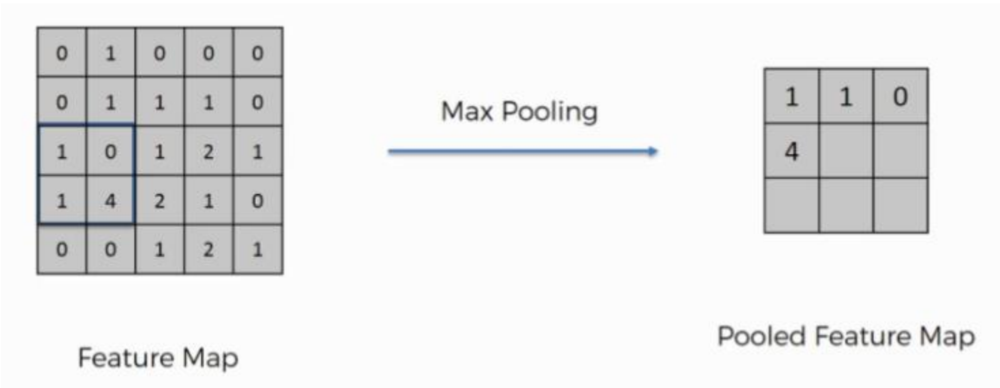

Fig 3.3. MaxPooling technique[2]

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

3.4 DROPOUT FUNCTION

To prevent overfitting, we used the methods shown in Fig. 4.10 to implement the Dropout layer, which sets input units arbitrarily to 0 with a repetition of rate at each advancement. The aim is to keep the aggregate of all information sources unchanged by increasing the information sources that are not set to zero by 1/(1 - rate).

```
model = Sequential()
model.add(Conv2D(32,kernel_size=(3,3),activation='relu',input_shape=(224,224,3)))
model.add(Conv2D(128,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))
```

Fig 3.4. Using the dropout function

3.5 IMAGE FLATTENING

Information is flattened[4] into a 1-dimensional cluster so that it may be passed on to the next layer. To create a single long component vector, we straighten the output of the convolutional layers. This layer is regarded as a totally associated layer since it is related with the last-order model. Ultimately, we put all the pixel information in one line and link it to the final layer.

```
model.add(Flatten())
model.add(Dense(64,activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1,activation='sigmoid'))
```

Fig 3.5. Deploying the image flattening technique

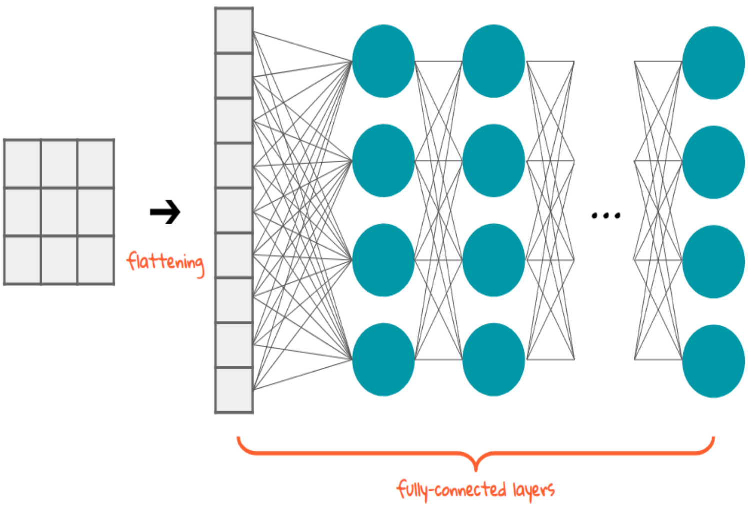

Fig 3.6. Flattening and fully-connected layers are what we have at the last stage of CNN.

IV. MOULDING TRAIN IMAGES

Making your training photos larger by altering their shape or using data-enhancing techniques is known as "moulding the train data," which is done by conducting various operations on the images. Figure 3.7 depicts this. Increasing the size of a dataset has numerous benefits over other techniques. As a consequence of data augmentation, you may conserve disc space and reduce the amount of time it takes to get results.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

```
#Moulding train images
train_datagen = image.ImageDataGenerator(rescale = 1./255, shear_range = 0.2,zoom_range = 0.2, horizontal_flip = True)

test_dataset = image.ImageDataGenerator(rescale=1./255)
```

Fig 3.7. Moulding train images

4.1 RESHAPING TEST AND VALIDATION IMAGES

Your data must be reshaped to meet the model's specifications. Our training data will be sliced to match your production needs throughout the course of training. If the original data form is precisely what you need for your model to provide the best results, you don't need to reshape it at all. The model will take care of the rest. Figures 3.7 and 3.8 show the unusual reshaping methods, such as "containing," "tiling," and "mirroring," in addition to the more typical ones, such as "interpolation" and "cropping," which are often employed with convolutional neural networks

```
#Reshaping test and validation images
train_generator = train_datagen.flow_from_directory(
    'CovidDataset/Train',
    target_size = (224,224),
    batch_size = 32,
    class_mode = 'binary')
validation_generator = test_dataset.flow_from_directory(
    'CovidDataset/Val',
    target_size = (224,224),
    batch_size = 32,
    class_mode = 'binary')
```

```
Found 192 images belonging to 2 classes.
Found 42 images belonging to 2 classes.
```

Fig 3.8. Reshaping test and Validation data

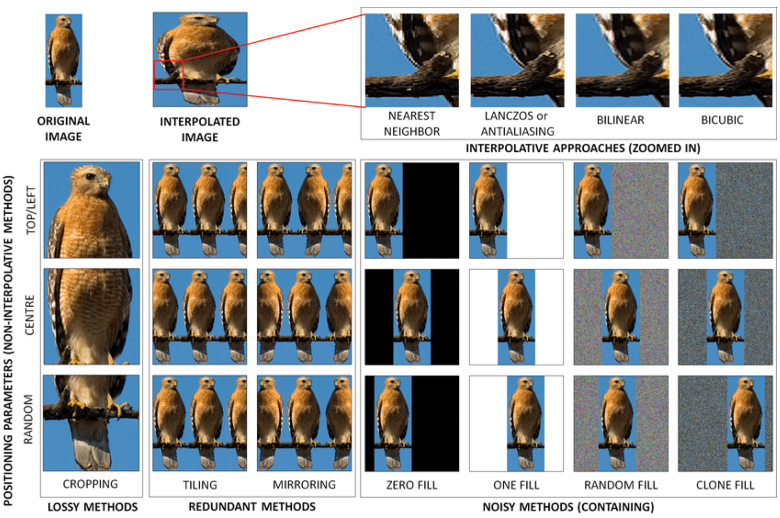

Fig 3.9 . Data Reshaping in CNN

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

4.2 TRAINING THE MODEL

It prepares the model for a fixed number of ages (emphasess on a dataset). Ages are the Integer. Number to prepare the model. An age is an emphasis over the whole x and y information gave. Note that related to introductory age, ages is to be perceived as "conclusive age". The model isn't prepared for various emphasess given by ages, yet simply until the age of record ages is reached. The model is prepared for 10 epochs as shown in Fig 3.10.

```
#Training the model
hist_new = model.fit_generator(
    train_generator,
    steps_per_epoch=8,
    epochs = 10,
    validation_data = validation_generator,
    validation_steps=2
)
```

Fig 3.10 . Training and running the model with 10 epoch

V. RESULT

Then we have to follow few steps to test our model

Step 1: Run the program. Fig 4.1 shows the basic user interface

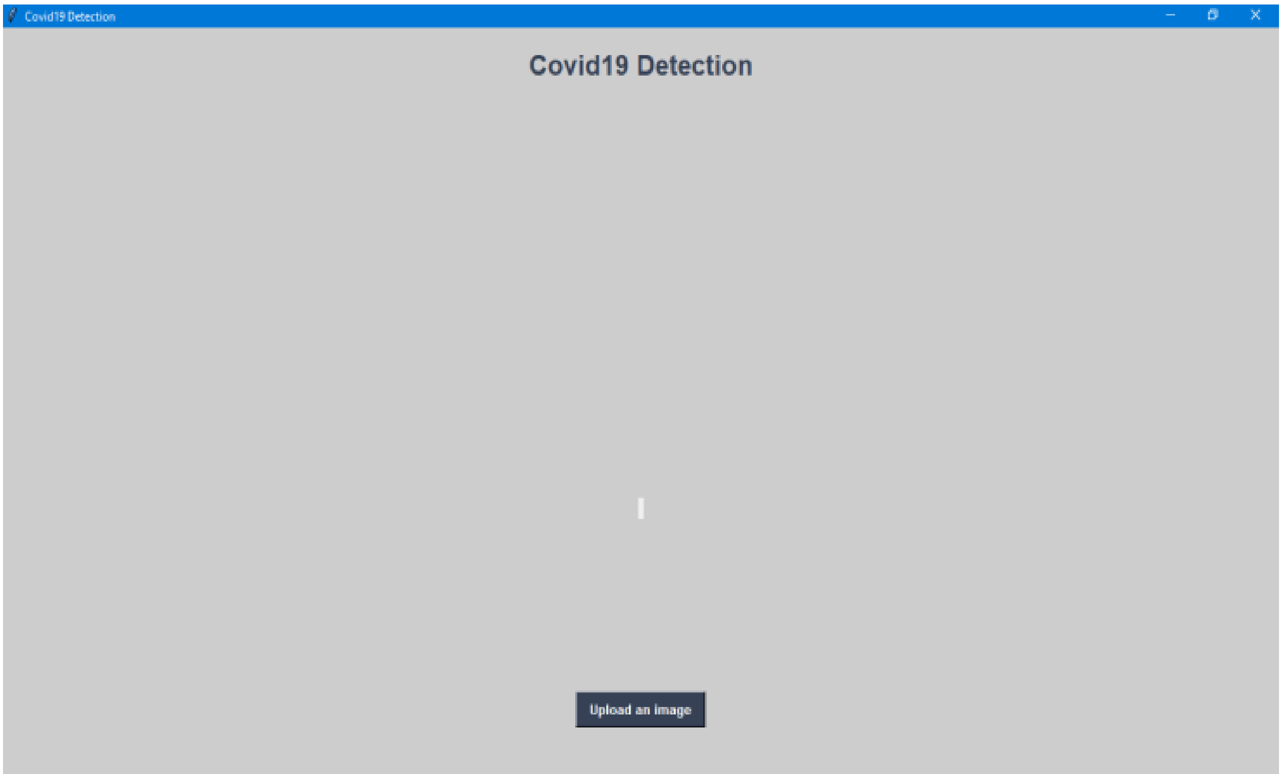

Fig 4.1. Basic GUI

Step 2: Click on “Upload an Image” and choose a file from the location as shown in Fig 4.2

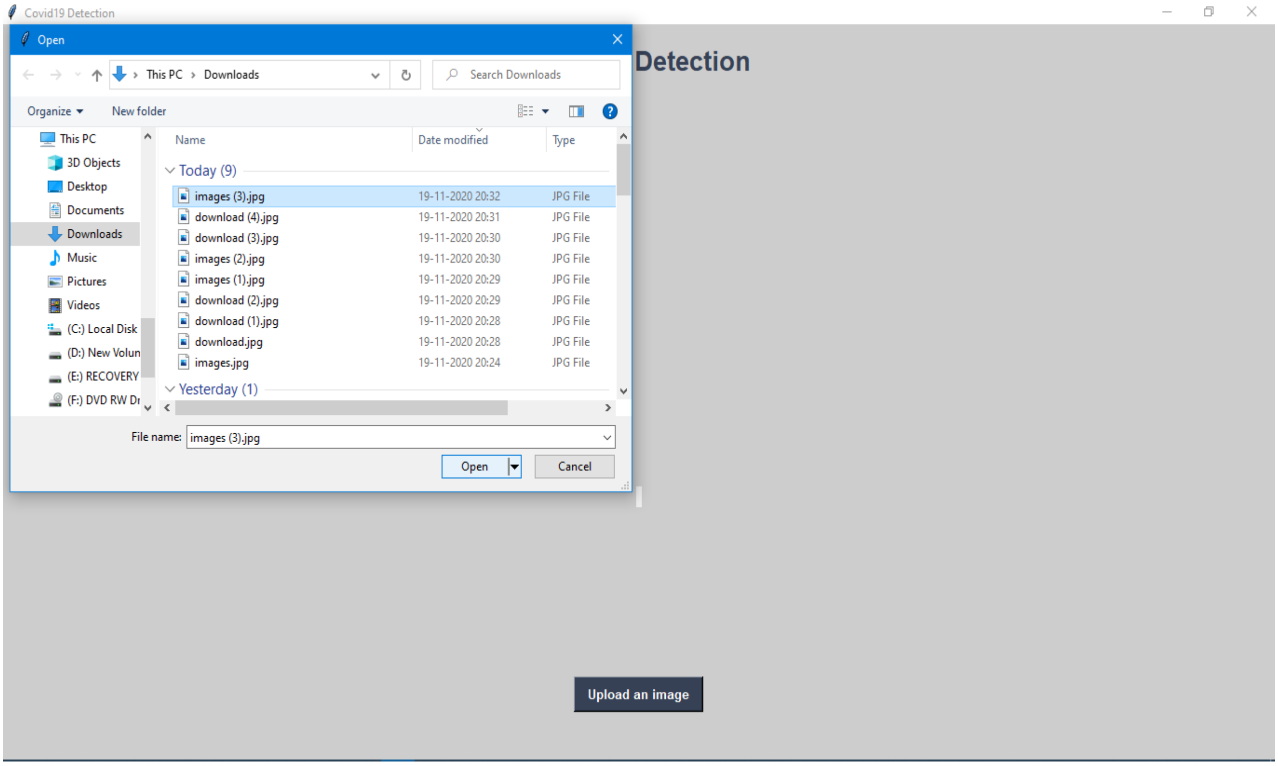

Fig 4.2 . Uploading an Image

Step 3: Once the file is loaded, click on “Classify Image” as shown in Fig 4.3 to view the output.

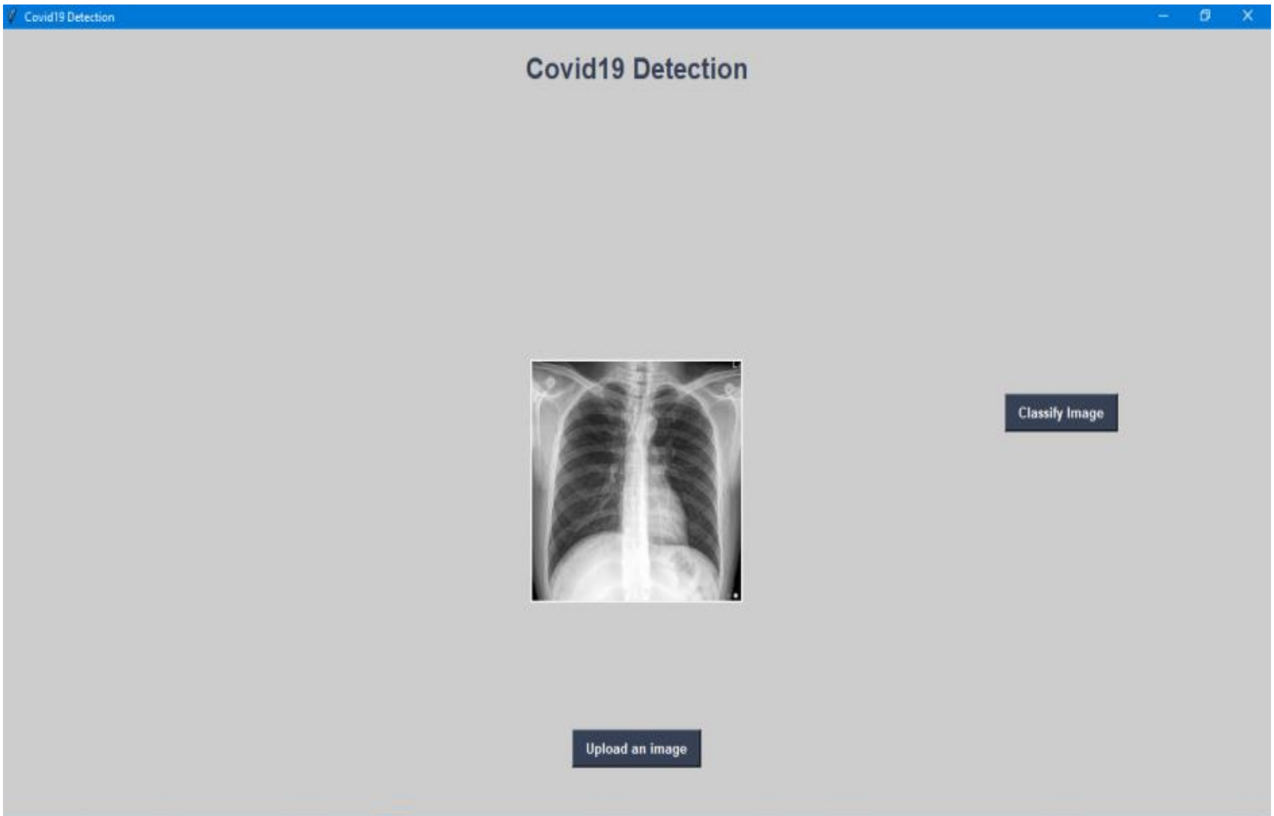

Fig 4.3 Classification of an Image

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

Test the model with different inputs to classify the image as covid or normal as shown in Fig 4.4

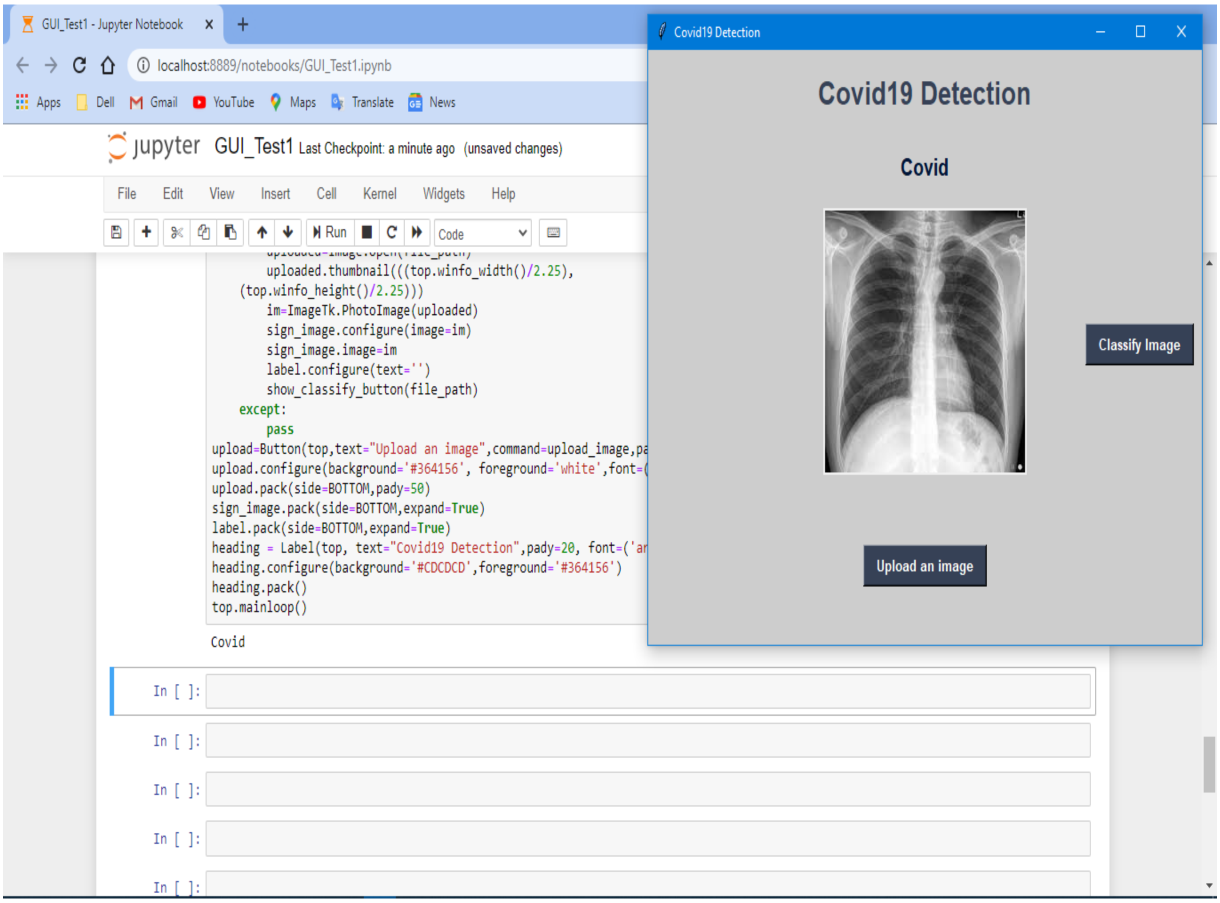

Fig 4.4. Figure showing the test results as covid or normal

Here are the results for few images other than human lungs as shown in Fig 4.5 and Fig 4.6.

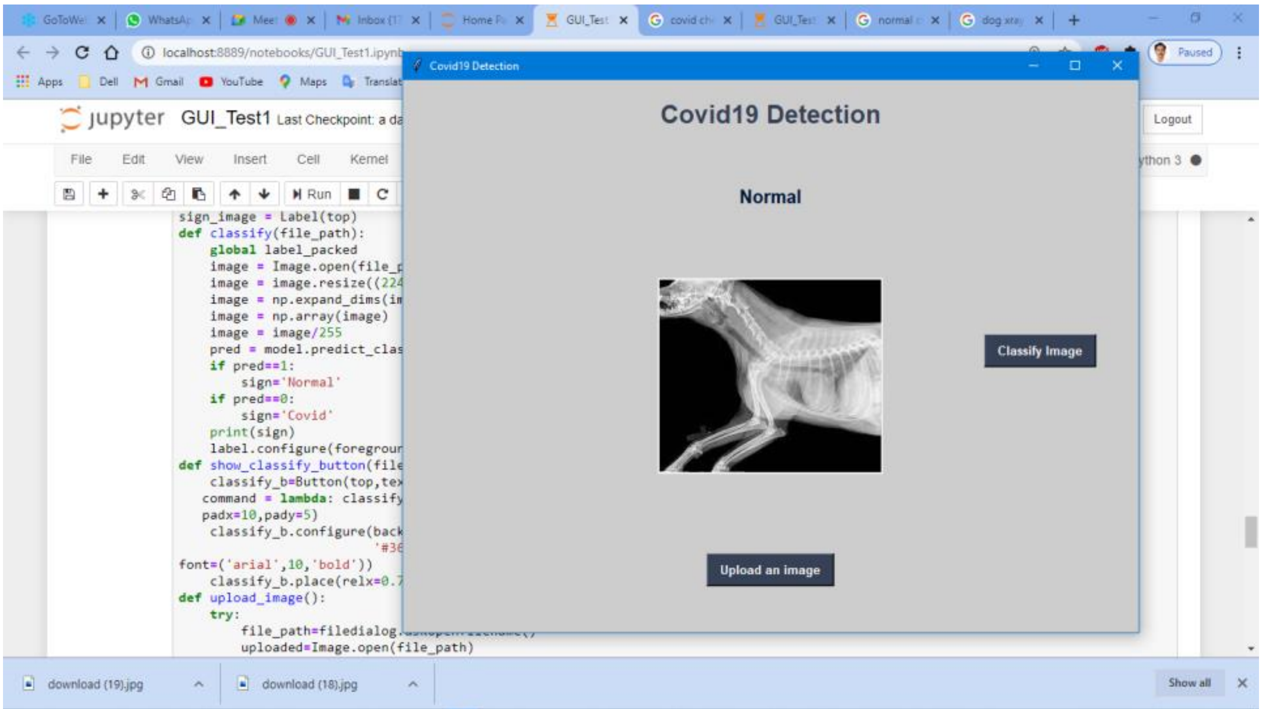

Fig 4.5. Figure showing the test results for animal x-ray

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2454 - 2463
https://publishoa.com
ISSN: 1309-3452

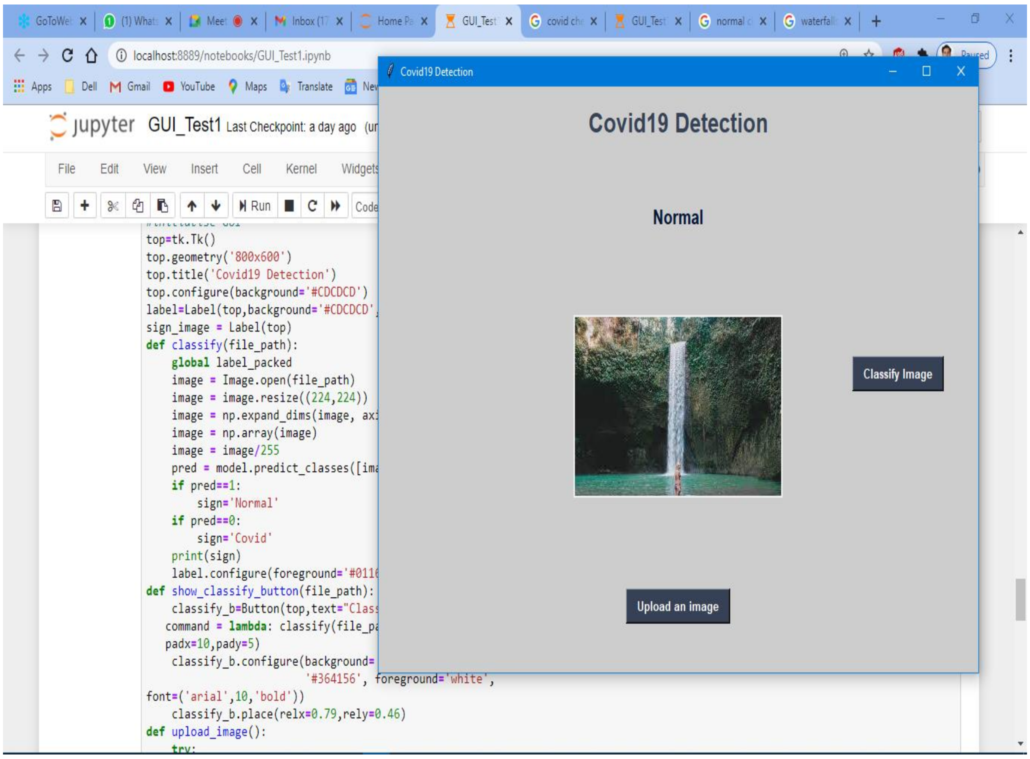

Fig 4.6. Figure showing the test results for nature image

VI. CONCLUSION

In order for qualified professionals and patients to save time and money, the recognition of COVID-19 from chest X-bar images is of paramount relevance. While the average person is unable to detect the illness, the CNN model can. A rapid and accurate evaluation of the illness may be made using this model in hospitals, allowing us to forecast its spread. In the Normal and COVID-19 classes, the results demonstrated that the convolutional neural organisation is capable of detecting COVID-19 images. In addition, we want to broaden this concept to predict the risk of infection:
According to preliminary data, the following variables increase a person's chance of contracting COVID-19:

- Age
- The patient has a history of health issues
- Hygiene practises, as a whole.
- Social customs,
- Involvement of people,
- How often you engage with others
- Climate and location
- Social and economic standing.

REFERENCES

1. Convolutional-Recursive Deep Learning for 3D Object Classification. Richard Socher, Brody Huval, Bharath Bhat, Christopher D. Manning and Andrew Y. Ng in NIPS 2012.
2. D.Narendar Singh Mrs. B.Pavitra. "IOT Enabled Patient Health Monitoring and Assistant System". European Journal of Molecular & Clinical Medicine, 7, 11, 2021, 5608-5618
3. Pavitra, B. and Singh, D. Narendar and Sharma, Sudhir Kumar, Smart Patient Assistance and Health Monitoring System Using IOT (February 21, 2020). Available at SSRN: https://ssrn.com/abstract=3553259 or http://dx.doi.org/10.2139/ssrn.3553259
4. Bengio, Y., LeCun, Y., Hinton, G. (2015). Deep Learning. Nature 521: 436-44. Goodfellow, I., Bengio, Y. and Courville, A. (2016) Deep Learning. MIT Press.
5. Going Deeper with Convolutions. Christian Szegedy, Wei Liu, Scott Reed, Vincent Vanhoucke, Andrew Rabinovitch, Yangqin Jia