VIT

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

J Component report

Programme : B.Tech (CSE).

Course Title : Artificial Intelligence.

Course Code : CSE3013.

Slot : B2+TB2.

Title: ***COVID Protocol Validation using Face mask and Social Distance Detection***

Team Members:

Amar Dixit | 19BCE1875.

Abhinav Vijayakumar | 19BCE1311.

Kunal Jaiswal | 19BCE1321.

Faculty: Dr.Padmavathy T V

Sign:

Date: 26|4|2022

DECLARATION

I hereby declare that the report titled "**COVID Protocol Validation using Face mask and Social Distance Detection**" submitted by me to VIT Chennai is a record of bona-fide work undertaken by me under the supervision of **Dr. Padmavathy T V**, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai.

Abhinav Vijayakumar (19BCE1311)
Amar Dixit (19BCE1875)
Kunal Jaiswal (19BCE1321)

Signature of the Candidate

CERTIFICATE

Certified that this project report entitled "**COVID Protocol Validation using Face mask and Social Distance Detection"** is a bonafide work of **Abhinav Vijayakumar(19BCE1311), Amar Dixit(19BCE1875), Kunal Jaiswal(19BCE1321)** and they carried out the Project work under my supervision and guidance for CSE3013 – Artificial Intelligence (AI).

Dr. Padmavathy T V
SCOPE, VIT Chennai

INDEX

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, *Dr. Padmavathy T V*, Professor, SCOPE, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to the Dean of the SCOPE, VIT Chennai, for extending the facilities of the School towards our project and for the unstinting support. We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

Abhinav Vijayakumar (19BCE1311)
Amar Dixit (19BCE1875)
Kunal Jaiswal (19BCE1321)

Signature of the Candidate

COVID Protocol Validation using Face mask and Social Distance Detection

Abhinav Vijayakumar, Amar Dixit, Kunal Jaiswal
19BCE1311 19BCE1875 19BCE1321

ABSTRACT

A novel virus has caused a world pandemic and huge life losses. Declared by the World Health Organization (WHO), this coronavirus originated from Wuhan, China in late December 2019. Upon thorough research, the virus has been observed as pathogenic and transmissible by air or by coming in close contact with an infected person. To avoid the spread of this virus, many measures have been suggested, such as maintaining a social distance, that is, maintaining a proper physical distance between people and lessening close contact with each other, and wearing a face mask to avoid the droplets from transmitting through the air.

Therefore, this project focuses and aims its study towards implementing a Social Distancing and Face Mask Detection System. This system will implement object detection and facial recognition in the video footages of pedestrians. Pre- trained models such as the CNN, ResNet Classifier and DSFD are used. People violating the minimum distance were detected as well as faces without face-masks were detected. An overall results board is displayed in the output containing the number of people violating or non-violating the respective measures. After implementing and deploying the models, this research project achieved a confidence score of 100%. Therefore, this research project concludes with proven facts that social distancing and wearing face masks helps reduce the spread of the virus and thus builds a model to help detect these measures.

INTRODUCTION

An Overview of COVID-19

Ever since being discovered first in Wuhan, China in December 2019, the whole world has been affected by the **CO**rona **VI**rus **D**isease **2019**, or COVID-19, causing the World Health Organization (WHO)[1] to declare a pandemic. The coronavirus is a novel disease and it has spread rapidly over 213 countries and territories resulting in an immense impact upon mankind. As of August 16, there are 21,641,369 active cases which might even increase in just a matter of time. With more than 769,481 deaths being reported to date[2], this virus is said to cause a pandemic. The origin of this virus is most likely said to be from bats or snakes but there has been no intermediate host detected yet. The Chinese researchers and health authorities assume that the virus may have been originated and spread by an animal species that was infected and spread to humans via being trafficked illegally in the seafood market in Wuhan. Researchers are still trying to discover the origin of this virus. The coronavirus causes respiratory and gastrointestinal infections and can be categorized into four different types which are Delta coronavirus, Gamma coronavirus, Alpha coronavirus and the Beta coronavirus. Birds are infected by the first two types whereas mammals are infected by the latter two (Fong, Dey and Chaki, 2020). While medical researchers are primarily working and preparing to develop a vaccine or any type of drug to help in the prevention of this virus, there are other possible social ways that may help in limiting the rapid spread of the virus within individuals.

According to the researchers, this virus can spread while coming in close contact with the person who is affected. Thus, it being alike all the other infectious respiratory diseases, researchers suggest close contact be withdrawn under any and all circumstances as the possibility of the virus transmitting through airborne ways and droplets is quite likely to occur. Therefore, to overcome the spread of coronavirus, researchers have come up with a new social norm, Social Distancing or Physical Distancing. This measure has been recognised and acknowledged to be a very useful measure in controlling the widespread of the virus. It ensures minimal human interaction within close distances on a national scale. The evaluation of the ways in how this social measure has been effective is investigated by studies by analyzing the confirmed and death cases in Italy and Spain during the national lockdown and following the upliftment of the lockdown with the help of statistics on a time-series (Tobías, 2020; Saez, Tobias, Varga and Barceló, 2020). The outcomes depicted that there were considerably lower reported cases after implementing social distancing in the two countries.

Background and Motivation

Social distancing has been an effective social measure in terms of normalizing the curve. It helps avoid any direct contact amongst individuals and assists in reducing any sorts of transmission of the droplets containing the virus via the human breath. Some studies are determined towards the exploration of trajectories that transmit the droplets via human breathing ways such as coughing, talking, sneezing and eating. While a few studies considered the droplet size to be correlated with the pathogens of the contagious disease, where they considered large droplets to be the ones that were to carry the microorganisms from the infected person (Christian et al., 2004; Mangili and Gendreau, 2005), other studies assumed small droplets of the particles that were in the nucleus form to be the ones to spread distantly. This was a topic of debate for a long time as questions like "what is a safe distance to maintain social distancing?" and "how further can the droplets spread?" arose. Hence, getting a considerable answer to all these questions was difficult as other factors also mattered, such as transmission through the surrounding air by evaporation.

Factors like biological, medical and engineering also needed to be considered to answer the above- mentioned questions. Thus, after considering all these factors, the suggested distance for a safe social distance was recommended to be at least 6 feet, that is, 2 meters. Some believed that this distance was not enough and needed to be increased (Setti et al., 2020). Although, the final distance was considered to be 2 meters and was implemented worldwide making it a successful parameter, now the distance has been reduced to 1 meter3 in some of the recovering countries. Therefore, considering all these factors, this research project proposes an AI based detection system that helps in detecting any kinds of violations. This research intends on building a model that can be applied in real-time systems and thus help in avoiding the spread of the virus.

COVID-19 CRISIS IN INDIA:

There have been record numbers of cases and deaths

Case numbers and deaths in India are continuing to rise fast, fuelled by a new variant.

The country recorded the world's highest single-day total on Thursday.

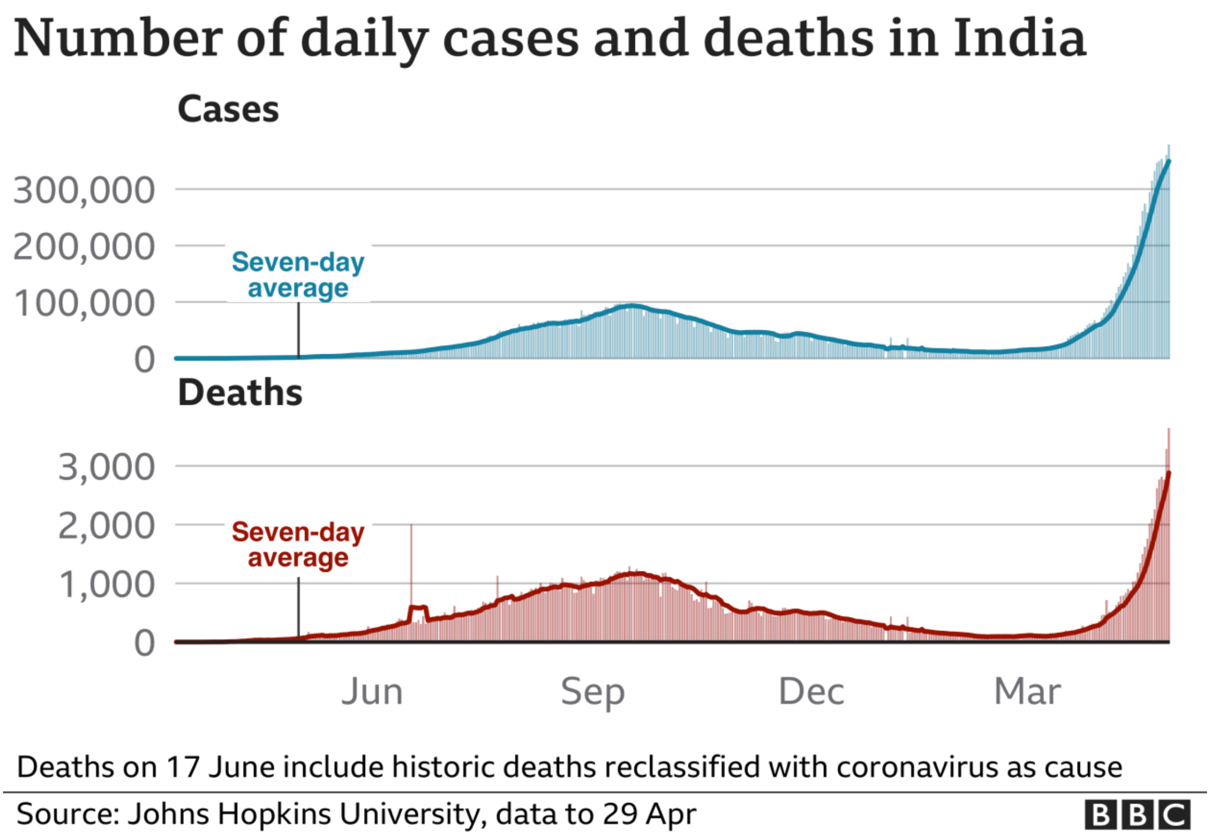


FIGURE 1

But the true numbers of cases and deaths are likely to be higher than the numbers provided by authorities, with many people avoiding testing or struggling to access it. Many deaths in rural areas also go unregistered.

Testing is increasing, but so too are the number of positive results.

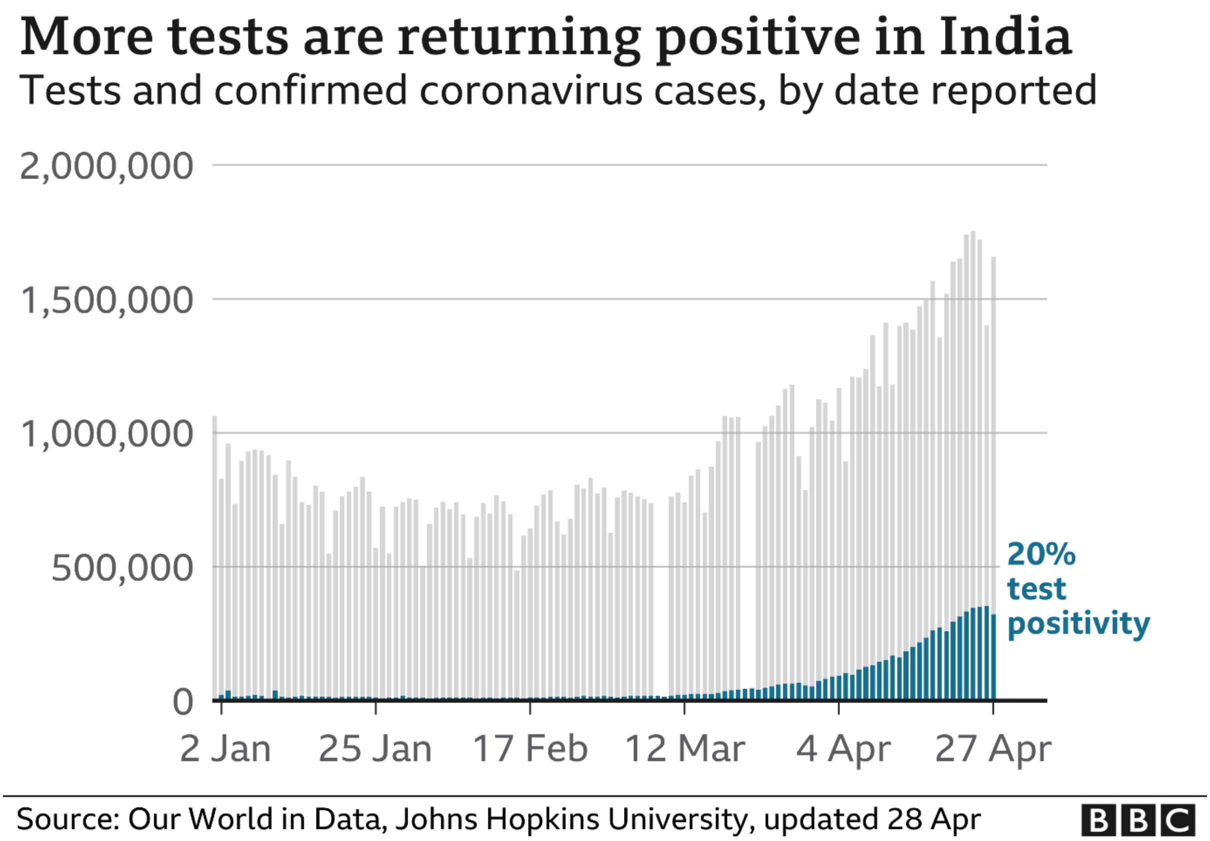


FIGURE 2

Last year, the World Health Organization recommended countries needed to get the positive test rate below 5% for at least two weeks before considering easing restrictions. The rate in India is now around 20%.

A high percentage of positive tests suggests high infection rates and the likelihood of many more people in the community with coronavirus undetected, according to Johns Hopkins University.

In total, India has confirmed more than 18 million infections and 200,000 deaths. Virologists say they expect the rate of infections to continue to increase for another two to three weeks.

There are very few critical care beds

The country has a chronic shortage of space on its intensive care wards, with many patients' families forced to drive for miles to try to find a bed for their loved one.

In Delhi - a region of about 20 million people - hospitals are full and are turning away new patients.

Doctors have described how people are dying on the streets outside hospitals as the country struggles to cope.

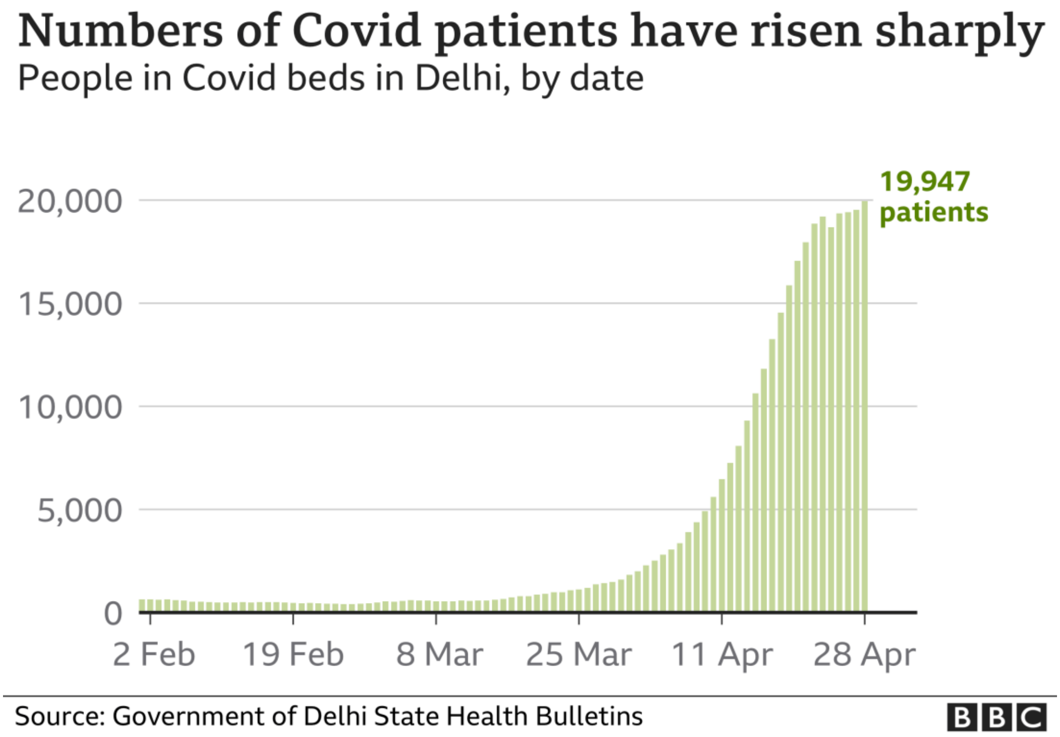


FIGURE 3

Some streets outside medical facilities have become crowded with the seriously ill, their loved ones trying to arrange stretchers and oxygen supplies for them as they plead with hospital authorities for a place inside.

"We have been roaming around for three days searching for a bed," one man told Reuters news agency as his wife sat immobile on the pavement.

FIGURE 4

On Monday, the government announced military medical infrastructure would be made available to civilians and retired medical military personnel would be helping out in Covid health facilities.

There's a shortage of oxygen

Hospitals across India are also experiencing oxygen shortages, with some forced to put up signs warning of a lack of supplies.

The country now has the greatest demand for oxygen out of all other low, lower-middle and upper-middle-income countries, according to the PATH Oxygen Needs Tracker.

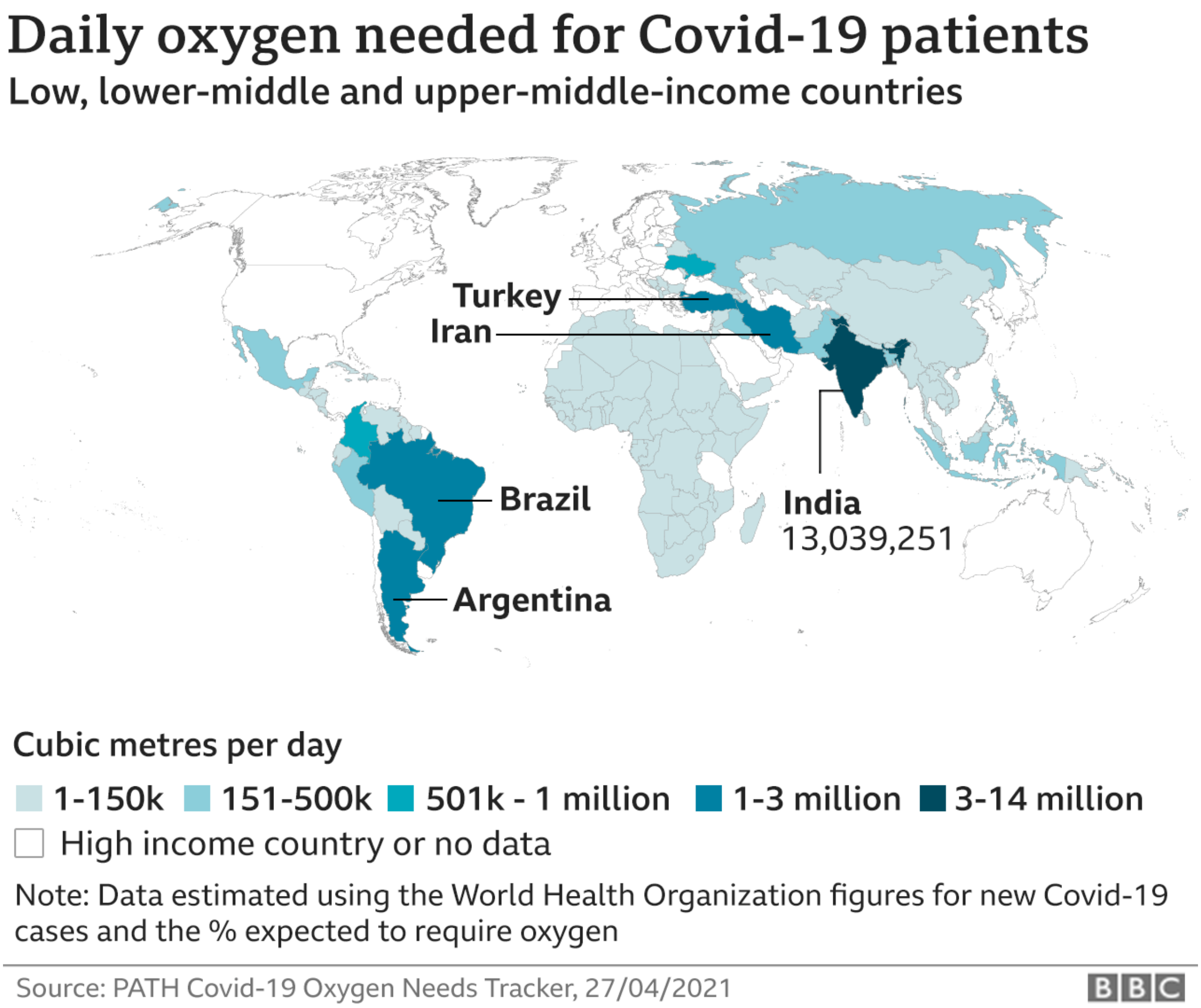


FIGURE 5

Demand has been growing between 6%-8% each day, according to PATH, an organisation that works with global institutions and businesses to tackle health problems.

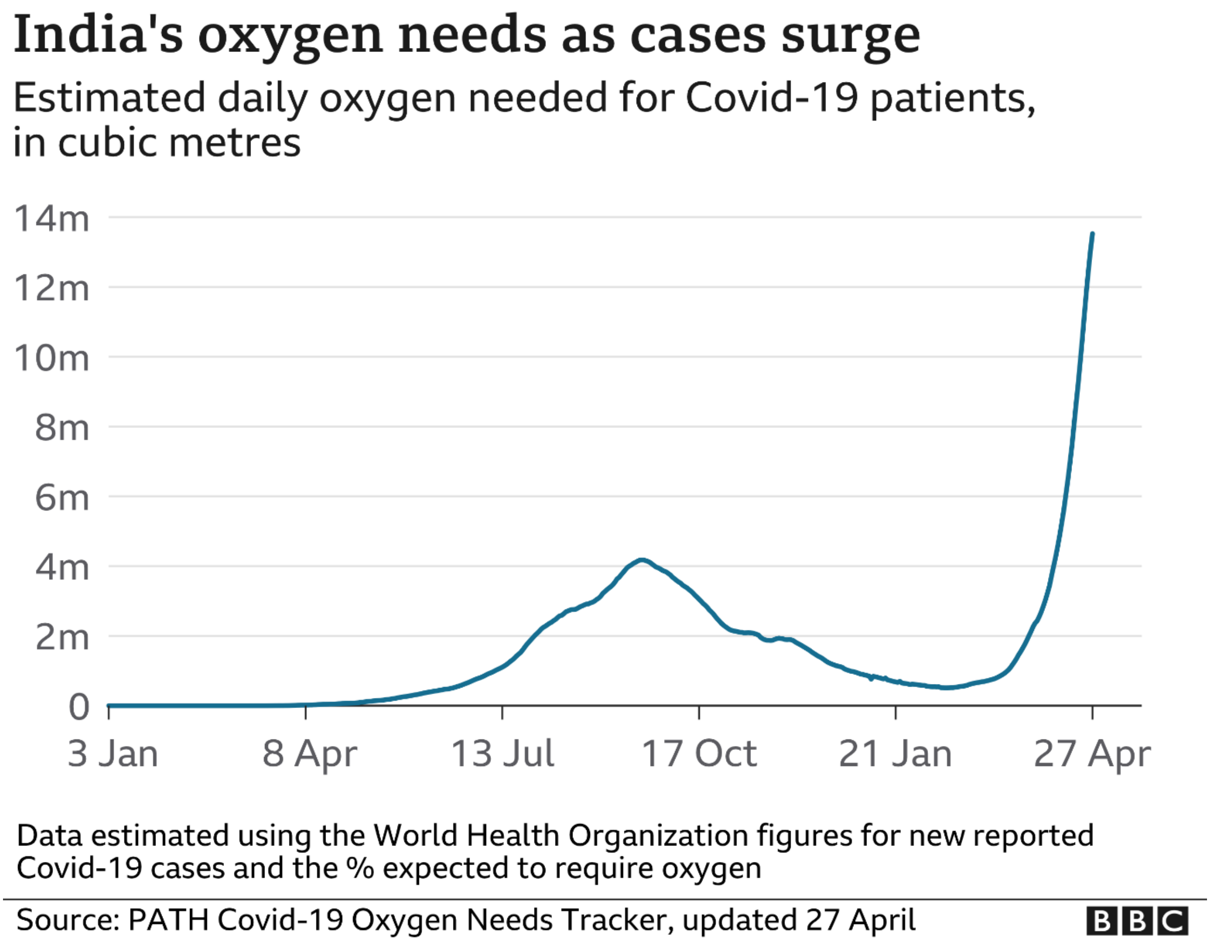
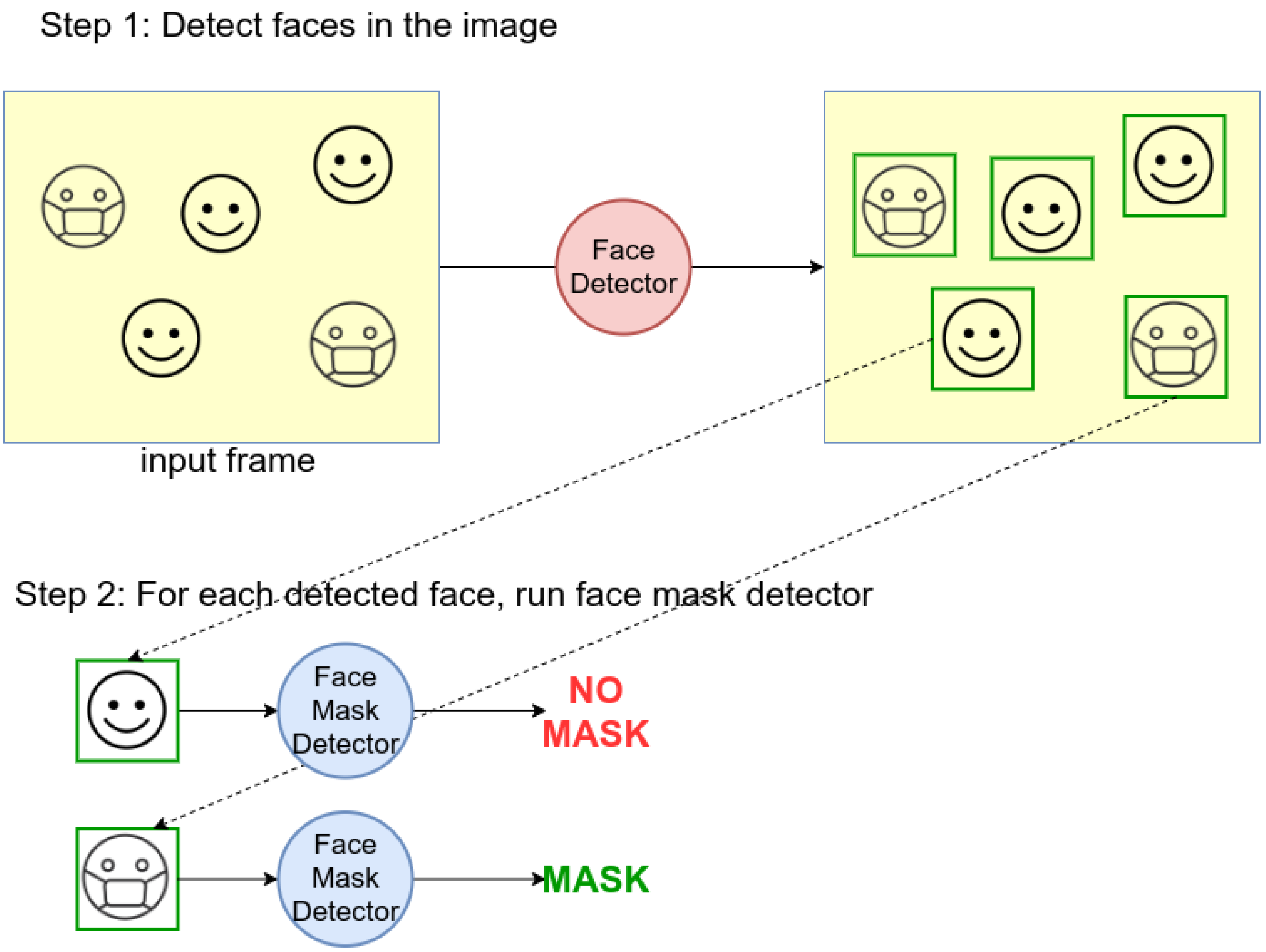


FIGURE 6

Why AI is needed for this project?

- For better performance of the system i.e better accuracy in detection.
- Faster and efficient detection.
- Cost efficient system in the long run.
- This system can also be used to detect people who try to fool the officials by not wearing a proper mask.
- AI trained models can detect people wearing masks more efficiently than human eye especially in situations like crowded places, moving vehicles etc.
- Using AI, we can track certain people who may have broken the protocol before by not wearing masks.



Applications

1. In crowded places like markets, malls, railway stations it becomes very difficult for the officials to identify people without mask. People roaming without masks can cause COVID to increase at a faster rate especially due to high population density.

2. This system can used in hospitals and clinics to check whether people entering the health centre especially the patients' wards are wearing mask or not.

3. This system can also be used to detect people who try to fool the officials by not wearing a proper mask.

Challenges/ issues

1. Quality/ Type of mask cannot be detected. It will be difficult to tell whether the person is really wearing a standard mask such as N-95
2. The system may fail if the person somehow hides his/her face and does not wear a mask, then the person may go undetected in public.
3. In case of overcrowded places, bad weather or night time(in absence of enough lighting), the accuracy of the system may be reduced and it may not detect masks accurately.

PEAS Description

- **Performance Measure** – Accuracy measure
- **Environment** – Input of images or videos from various sources such as recorded videos etc.
- **Actuators** – Detection of people with or without masks along with percentage area of face covered.
- **Sensors** – As of now static image files are input to the software. But in the future, live video feeds from CCTVs input can be made common.

Environment type

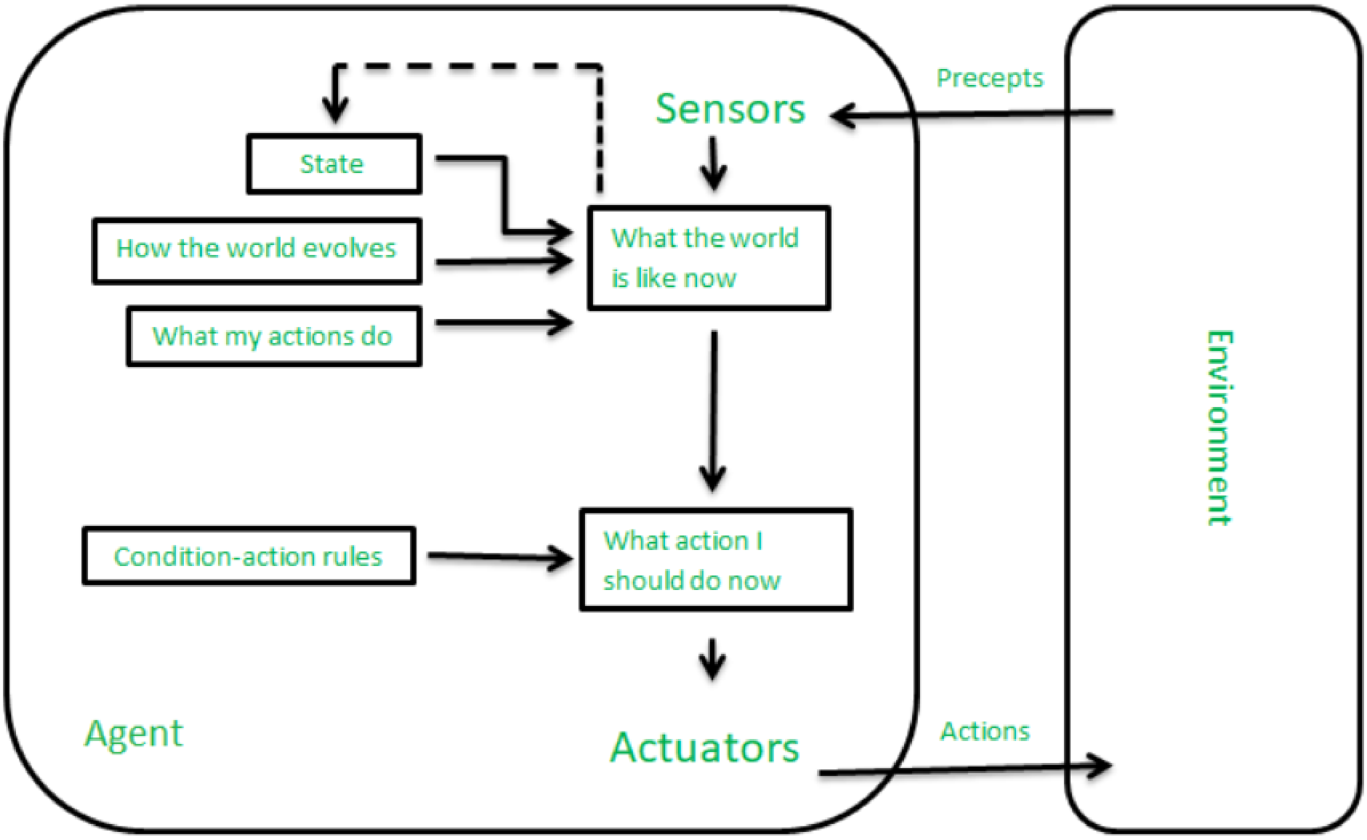
1. Deterministic
2. Dynamic
3. Complete Observation
4. Single Object Detection
5. Continuous

Architecture type

Model-based reflex agents:

It works by finding a rule whose condition matches the current situation. A model-based agent can handle partially observable environments by the use of a model about the world. The agent has to keep track of the internal state which is adjusted by each percept and that depends on the percept history. The current state is stored inside the agent which maintains some kind of structure describing the part of the world which cannot be seen.
Updating the state requires information about:

- how the world evolves independently from the agent,
- how the agent's actions affect the world.



Existing system

- Since the infectious coronavirus disease (COVID-19) was first reported in Wuhan, it has become a public health problem in China and even around the world. This pandemic is having devastating effects on societies and economies around the world. The increase in the number of COVID-19 tests gives more information about the epidemic spread, which may lead to the possibility of surrounding it to prevent further infections. However, wearing a face mask that prevents the transmission of droplets in the air and maintaining an appropriate physical distance between people, and reducing close contact with each other can still be beneficial in combating this pandemic. Therefore, this project focuses on implementing a Face Mask and Social Distancing Detection model as an embedded vision system.

- The pre-trained models such as the MobileNet, ResNet Classifier, and VGG are used in our context. People violating social distancing or not wearing masks were detected. After implementing and deploying the models, the selected one achieved a confidence score of 100%. This project also provides a comparative study of different face detection and face mask classification models. The system performance is evaluated in terms of precision, recall, F1-score, support, sensitivity, specificity, and accuracy that demonstrate the practical applicability. The system performs with F1-score of 99%, sensitivity of 99%, specificity of 99%, and an accuracy of 100%. Hence, this solution tracks the people with or without masks in a real-time scenario and ensures social distancing by generating an alarm if there is a violation in the scene or in public places. This can be used with the existing embedded camera infrastructure to enable these analytics which can be applied to various verticals, as well as in an office building or at airport terminals/gates.

Proposed Methodology

Data Collection and Preparation

This project uses a two pedestrian videos, the first is obtained online from YouTube which is provided by BriefCam and the other video footage of the Oxford Town Centre. The first dataset contains a CCTV footage of pedestrians in an area walking on the pavement and the other dataset contains a video of people walking in a busy downtown centre in Oxford, England. The Oxford Town Centre dataset has been utilized multiple times in multiple research projects. These are open datasets and can be used for various developments and research projects in the area of object detection, facial recognition, and many such other projects.

This dataset is a very unique dataset, in the sense that it uses video footage straight out of a public CCTV camera that on the contrary was assigned for public safety reasons. In this video, it shows that the pedestrians are walking or acting in a normal and unrehearsed manner. These pedestrians are just normal people walking on the road or pathway minding their own business.

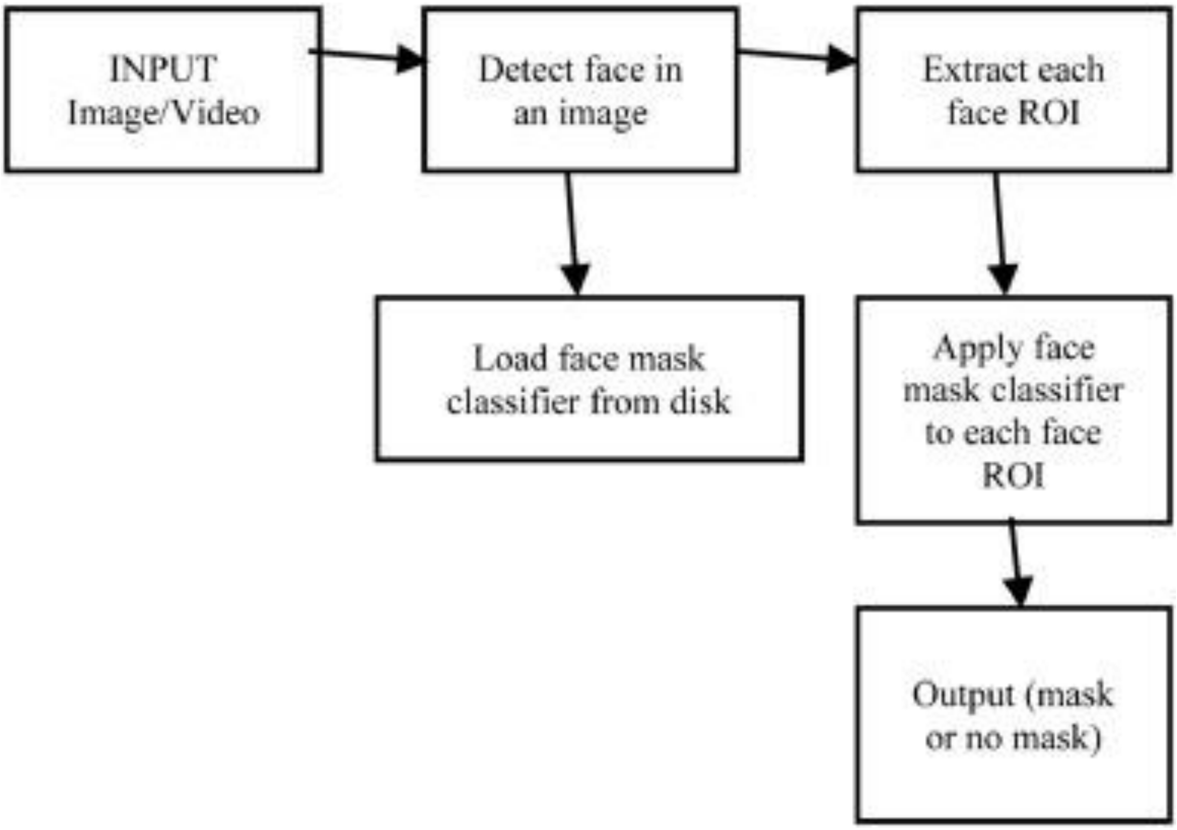
Although these people do not know about the research projects, they were aware of being under supervision cameras and it is with their consent that this dataset has been created hence not violating any ethical issues.

Data Pre-processing

This research first downloads the video footages that are available online. The video footage contains a fixed camera that detects individuals in a region of interest (ROI) and measures their distances in real-time without recording any sort of data. Moreover, this research proposes a novel approach towards detecting people and whether or not they are violating any social distancing regulations.

While detecting the interpersonal distances between the individuals present in the video, with the help of facial detection, their faces are detected to verify whether the individuals are wearing a mask or not.

ARCHITECTURE DIAGRAM



MODULES

1. Import necessary libraries
2. Loading, importing and pre-processing data
3. Using haar cascade to detect faces
4. Data Augmentation
5. Building transfer learning model
6. Testing model on test data

1. Import necessary libraries

We have imported libraries such as numpy, pandas, cv2 and distance.

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import cv2
from scipy.spatial import distance
```

2. Loading, importing and pre-processing data

```
[ ] #loading haarcascade_frontalface_default.xml
    face_model = cv2.CascadeClassifier('/content/gdrive/MyDrive/Kaggle/haarcascade_frontalface_default.xml')
```

3. Using Haar cascade to detect faces

It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. We used a Haar Cascade Model trained to detect faces in order to obtain the bounding box coordinates of faces in an image.

```
#plotting
for (x,y,w,h) in faces:
    cv2.rectangle(out_img,(x,y),(x+w,y+h),(0,0,255),1)
plt.figure(figsize=(12,12))
plt.imshow(out_img)
```

```
<matplotlib.image.AxesImage at 0x7f016ebecc90>
```

4. Data Augmentation

Data augmentation in data analysis are techniques used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from existing data. It acts as a regularizer and helps reduce overfitting when training a machine learning model.

It is a collection of techniques that can be used to extend the dataset size and improve the quality of images in the dataset by a required amount. Logically it is used to make the deep learning model independent of the counterfeit features of the data space.

```
Found 10000 images belonging to 2 classes.
Found 800 images belonging to 2 classes.
Found 800 images belonging to 2 classes.
```

5. Building transfer learning model

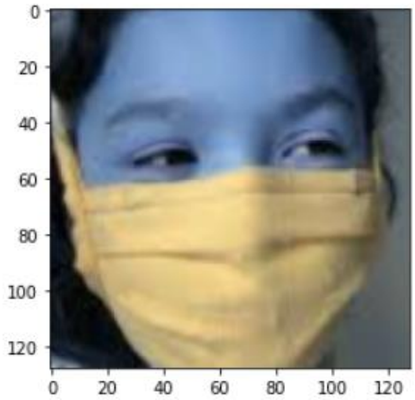
Transfer learning (TL) is a research problem in machine learning (ML) that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. For example, knowledge gained while learning to recognize faces and masks could apply when trying to detect masks on faces.

```
80142336/80134624 [==============================] - 1s 0us/step
80150528/80134624 [==============================] - 1s 0us/step
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg19 (Functional)          (None, 4, 4, 512)         20024384

 flatten (Flatten)           (None, 8192)              0

 dense (Dense)               (None, 2)                 16386

=================================================================
Total params: 20,040,770
Trainable params: 16,386
Non-trainable params: 20,024,384
_________________________________________________________________
```

6. Testing model on test data

- Cropping image to 128 pixels to only cover the face region:

```
sample_mask_img = cv2.imread('/content/gdrive/MyDrive/Kaggle/Face Mask Dataset/Test/WithMask/1565.png')
sample_mask_img = cv2.resize(sample_mask_img,(128,128))
plt.imshow(sample_mask_img)
sample_mask_img = np.reshape(sample_mask_img,[1,128,128,3])
sample_mask_img = sample_mask_img/255.0
```



- Integrate with haar cascade

We now take crops of the faces detected in the image and use the model trained in the above section to determine whether the individual faces have a mask or not.

TECHNOLOGIES AND INTEGRATION

Algorithms Used:

SUPERVISED ML ALGORITHM

A supervised learning algorithm takes a known set of input data (the learning set) and known responses to the data (the output), and forms a model to generate reasonable predictions for the response to the new input data.

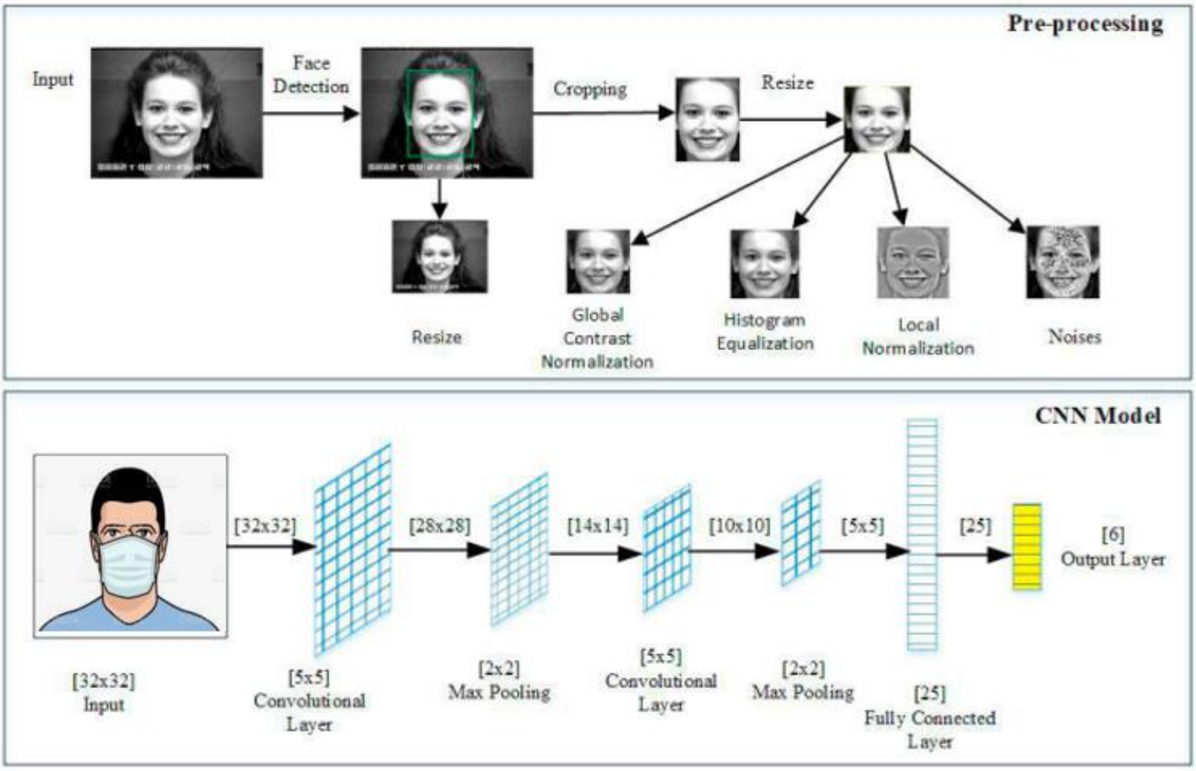
Since we are providing with the learning set of input data i.e. images of people with and without masks and our model generates responses to the provided data, therefore we can say that we are using supervised learning method.

This best approximates the relationship between input and output observable in the data.

1. CNN- Convolutional neural networks

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

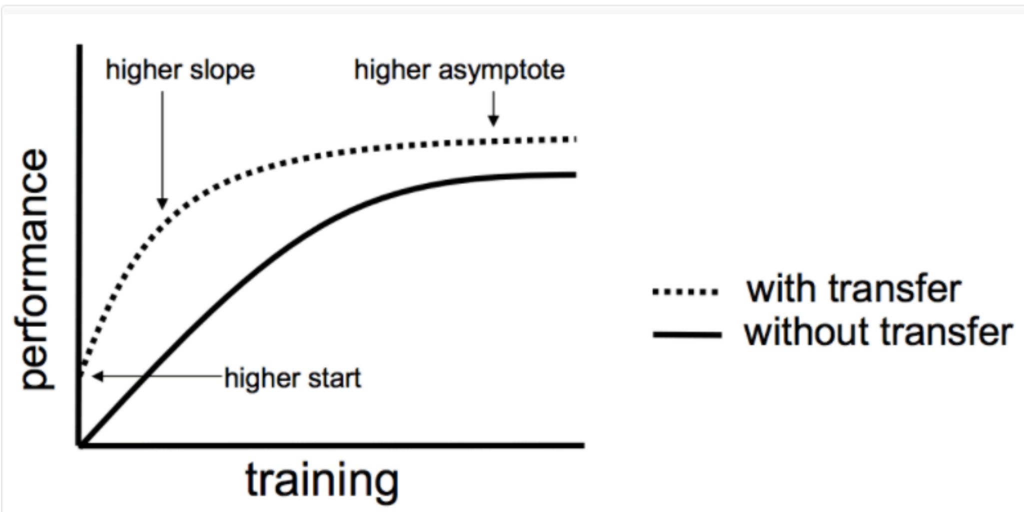
A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.



2. Transfer learning

The reuse of a pre-trained model on a new problem is known as transfer learning in machine learning. A machine uses the knowledge learned from a prior assignment to increase prediction about a new task in transfer learning.

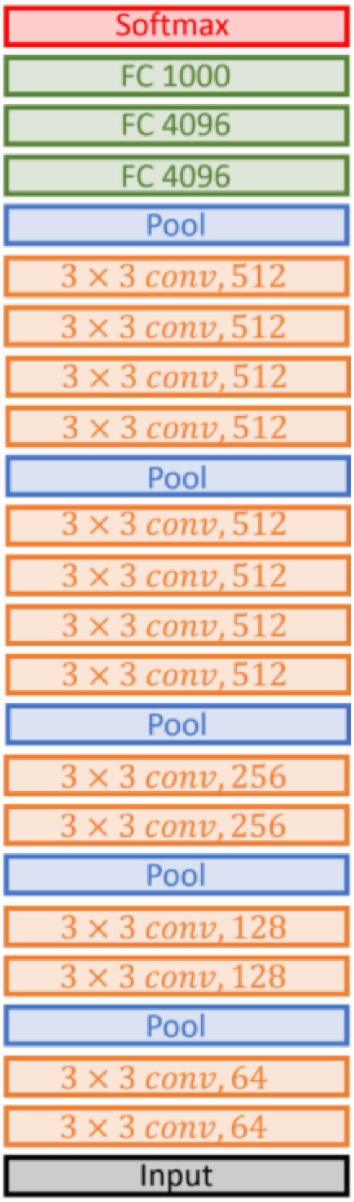
The knowledge of an already trained machine learning model is transferred to a different but closely linked problem throughout transfer learning. For example, if you trained a simple classifier to predict whether an image contains a backpack, you could use the model's training knowledge to identify other objects such as sunglasses.



3. VGG19

VGG-19 is a convolutional neural network that is 19 layers deep. You can load a pre-trained version of the network trained on more than a million images from the ImageNet database. The pre-trained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

VGG19 is a variant of VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). There are other variants of VGG like VGG11, VGG16 and others. VGG19 has 19.6 billion FLOPs.



VGG19

IMPLEMENTATION DETAILS (SOFTWARES USED)

Softwares Used:

Google Colab for compiling and executing the code.

Python Libraries Used:

- Numpy
- Pandas
- Cv2
- Scipy
- Matplotlib
- Keras
- VGG19
- ImageDataGenerator

Object Detection and Tracking

Although there has been a confusion between the two terms Image Classification and Object Detection, often meaning them to be the same, they are completely different. Image Classification performs identification of objects in images while Object Detection performs identification of the objects including its location in the images. Both of these terms are widely popular in Computer Vision tasks (Merkulova, Shavetov, Borisov and Gromov, 2019). They can be used in every field possible such as healthcare, defence, sports, and various other industries.

The next question that arises is whether Object Detection and Tracking are the same terms or not. Yes, Object Detection and Tracking are two very similar terms in the way they are functioned. They are basically designed to perform the same functionality but with a little difference. Object Detection is used to detect objects present in an image or in multiple

images where an object is stationary while Object Tracking performs detection on videos, that is, it keeps a track of the following object detected while it is moving (Porikli and Yilmaz, 2012). A video is a combination of fast-moving frames and thus identification of the objects and their location from every frame is performed by Object Tracking.
Object detection can be stated as a fundamental problem in the computer vision and the images domain. It intends to detect objects in the video that belong to specific classes such as humans, vehicles, and more. The deep neural network models like CNN have dominated the benchmarks of object detection. Pre-trained models like the MS COCO7, has more than 896K objects and over 123K images in the training and validation set and almost 80K images with more than 80 categories in the testing set. With the help of supervised learning techniques like data augmentation, these models are trained.

The best approach to building a model to perform object detection is with the help of a Sliding Window technique (Glumov, Kolomiyetz and Sergeyev, 1995). The sliding window approach is where an image is divided into particular sizes and regions and then according to the region, the respective classes are classified as shown in Figure 1. Here, in this research, the sliding window will detect all the people present in the video footage and form a bounding box around it. A CNN classifier will state a confidence value where it will represent on the certainty that the window contains an individual or not. Then for each and every region, a CNN will be passed which will extract the features and then further pass it on to the classifier and the regressor.

<matplotlib.image.AxesImage at 0x7f016ebecc90>

Face Mask Detection

Dual Shot Face Detector is utilized in this research to detect the individuals' faces. It is a method which derives from SSD and offers a Feature Enhance Module (FEM) to transfer original features to expand the single shot to the dual shot detector. Conventional face detectors such as the MTCNN or the Haar- Cascades are not useful for this research as it lacks in detecting faces that are in low-resolutions or faces that are covered. DSFD is a bit complex and heavy on the pipeline but it delivers accurate results. It is widely used where detections are in a more large-scale ranged-orientations. As this research is working on video frames, the probability of encountering blurred faces is high and thus with the help of DSFD none of them will be missed. The blurriness could occur due to various reasons such as the face being out of focus or any random rapid movements or noise during the capture of the video.

A somewhat modified ResNet50 model whose layers have been pre-trained on ImageNet is used to classify individuals based on whether they are with a mask or without a mask. The basic methodology behind this algorithm is to first, divide the video into frames and detect people and their faces in every frame, individually. Later on, the frames are combined which then again forms a video. It works as follows:

1. Capture the video.
2. Read the video by dividing it into a number of frames.
3. Else, detect persons in each frame and get the bounding boxes around them with the help of CNN.

a. If it reaches to EOF, stop.

4. Further, get the positions of the people with the help of DBSCAN to detect where the clusters are forming.
5. While detecting persons, detect their faces with the help of the DSFD model to detect whether they have masks on or not.
6. With the help of bounding boxes on the person and their faces, measure the distance between them and detect masks on them.
7. Create a results board on top of the video and display the results.
8. Create an output stream and then show the results.
9. Do this for every frame till it reaches to end of file.

Social Distance Detection

In Social Distance Detection, the bounding box information, mainly centroid information, is used to compute each bounding box centroid distance. We used Euclidean distance and calculated the distance between each detected bounding box of peoples. Following computing centroid distance, a predefined threshold is used to check either the distance among any two bounding box centroids is less than the configured number of pixels or not. If two people are close to each other and the distance value violates the minimum social distance threshold. The bounding box information is stored in a violation set, and the color of the bounding box is updated/changed to red. A centroid tracking algorithm is adopted for tracking so that it helps in tracking of those people who violate/breach the social distancing threshold. At the output, the model displays the information about the total number of social distancing violations along with detected people bounding boxes and centroids.

Code for Social Distance Detection

Detecting social distancing violations

This can be done by iterating over the coordinates of faces and calculating the distance for each possible pair, if the distance for a particular pair is less than MIN_DISTANCE then the bounding boxes for those faces are colored red. MIN_DISTANCE must be manually initialized in such a way that it corresponds to the minimum allowable distance in real life (ex. 6ft in India)

```
[ ]  MIN_DISTANCE = 130
```

```
if len(faces)>=2:
    label = [0 for i in range(len(faces))]
    for i in range(len(faces)-1):
        for j in range(i+1, len(faces)):
            dist = distance.euclidean(faces[i][:2],faces[j][:2])
            if dist<MIN_DISTANCE:
                label[i] = 1
                label[j] = 1
    new_img = cv2.cvtColor(img, cv2.COLOR_RGB2BGR) #colored output image
    for i in range(len(faces)):
        (x,y,w,h) = faces[i]
        if label[i]==1:
            cv2.rectangle(new_img,(x,y),(x+w,y+h),(255,0,0),1)
        else:
            cv2.rectangle(new_img,(x,y),(x+w,y+h),(0,255,0),1)
    plt.figure(figsize=(10,10))
    plt.imshow(new_img)

else:
    print("No. of faces detected is less than 2")
```

RESULTS AND DISCUSSIONS

After implementing and deploying the models, this research project achieved a great extent of detection of faces and face mask. The confidence score10 is the probability of a bounding box containing an object. Below figure shows the face detection.

The figure given below shows the accurate results obtained after implementing the models on the two datasets. On the top of the frame, there is a box that contains the information of whether the person I swearing a mask or not and whether the people are socially distanced, that is, whether they are implementing social distancing or not, and the number of people wearing a mask, not wearing a mask, and unknown in case the person’s face is not facing towards the camera. Here we can see that the people have been identified perfectly with bounding boxes around them. The green box means they are safe, or within the social distancing limit, that is 1m. The red box tells that the people are unsafe. The small box on the faces of the people detect whether they are wearing a mask or not. Red means they are not wearing a mask and green means that they are wearing a mask.

CONCLUSION

Thus, by understanding and studying about the virus, this project can conclude two things: social distancing can reduce the spread of the coronavirus; face coverings help prevent the infectious disease to transmit via the air. Therefore, to support this study, this research proposed an AI-based real-time approach towards the detection of social distancing and face mask. Crowd density was examined by gaining the ROI of the video frames and the count of people violating and non-violating the measures was also shown. The results obtained were accurate and real-time based. The pandemic is continuing to increase and is still going on while studying this research. Many countries are still in the quarantine phase whereas some countries have terminated the quarantine phase and are stuck to their social distancing measures. Social distancing may be implemented forever in some countries and people may have to live with the new normal. Russia has declared to have found a vaccine to cure the virus and has started the production of it[11]. If it is successful, the world might finally be able to go back to how it was.

FUTURE WORK

- In this research, the people who have close relationships or who know each other and are walking together are detected as they are violating the social distancing measure. Some argue that they should practice social distancing in public areas while some argue that they need not. Hence, this is one of the limitations in this research and can be considered for future work. This research does not make use of the bird's eye view function and hence can be considered for future work as well.

- Some places where the virus transmission is high, people are required to wear double face masks, and thus to detect whether a person is wearing a double face mask can be achieved in the future. If in the future there is no need for social distancing, this project can be reutilized and repurposed for other applications. Museums have a 2-3 feet distance policy between the artefact and the individual and thus it can be used to detect this distance.

REFERENCES

1. Aldila, D., Khoshnaw, S., Safitri, E., Anwar, Y., Bakry, A., Samiadji, B., Anugerah, D., GH, M., Ayulani, I. and Salim, S., 2020. A mathematical study on the spread of COVID-19 considering social distancing and rapid assessment: The case of Jakarta, Indonesia. Chaos, Solitons & Fractals, 139, p.110042.

2. Christian, M., Loutfy, M., McDonald, L., Martinez, K., Ofner, M., Wong, T., Wallington, T., Gold, W., Mederski, B., Green, K. and Low, D., 2004. Possible SARS Coronavirus Transmission during Cardiopulmonary Resuscitation. Emerging Infectious Diseases, 10(2), pp.287-293.

3. Deore, G., Bodhula, R., Udpikar, V. and More, V., 2016. Study of masked face detection approach in video analytics. 2016 Conference on Advances in Signal Processing (CASP),.

4. Fong, S., Dey, N. and Chaki, J., 2020. An Introduction to COVID-19. Artificial Intelligence for Coronavirus Outbreak, pp.1-22.

5. Glumov, N., Kolomiyetz, E. and Sergeyev, V., 1995. Detection of objects on the image using a sliding window mode. Optics & Laser Technology, 27(4), pp.241-249.

6. Huynh, T., 2020. Does culture matter social distancing under the COVID-19 pandemic?. Safety Science, 130, p.104872.

7. Li, G., Yang, Y. and Qu, X., 2020. Deep Learning Approaches on Pedestrian Detection in Hazy Weather. IEEE Transactions on Industrial Electronics, 67(10), pp.8889-8899.

8. Li, J., Wang, Y., Wang, C., Tai, Y., Qian, J., Yang, J., Wang, C., Li, J. and Huang, F., 2019. DSFD: Dual Shot Face Detector. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),.

9. Liao, G., Yang, G., Tong, W., Gao, W., Lv, F. and Gao, D., 2019. Study on Power Line Insulator Defect Detection via Improved Faster Region-Based Convolutional Neural Network. 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT),.

10. Lin, K., Zhao, H., Lv, J., Li, C., Liu, X., Chen, R. and Zhao, R., 2020. Face Detection and Segmentation Based on Improved Mask R-CNN. Discrete Dynamics in Nature and Society, 2020, pp.1-11.

11. Loey, M., Manogaran, G., Taha, M. and Khalifa, N., 2020. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. Measurement, 167, p.108288.

12. Mangili, A. and Gendreau, M., 2005. Transmission of infectious diseases during commercial air travel. The Lancet, 365(9463), pp.989-996.

13. Mangmang, G., 2020. Face Mask Usage Detection Using Inception Network. Journal of

Advanced Research in Dynamical and Control Systems, 12(SP7), pp.1660-1667.

14. Merkulova, I., Shavetov, S., Borisov, O. and Gromov, V., 2019. Object detection and tracking basics: Student education. IFAC-PapersOnLine, 52(9), pp.79-84.

15. Porikli, F. and Yilmaz, A., 2012. Object Detection and Tracking. Studies in Computational Intelligence, pp.3-41.

16. Pranav, K. and Manikandan, J., 2020. Design and Evaluation of a Real-time Pedestrian Detection System for Autonomous Vehicles. 2020 Zooming Innovation in Consumer Technologies Conference (ZINC).

17. Richard, M., Kok, A., de Meulder, D., Bestebroer, T., Lamers, M., Okba, N., Fentener van Vlissingen, M., Rockx, B., Haagmans, B., Koopmans, M., Fouchier, R. and Herfst, S., 2020. SARS-CoV-2 is transmitted via contact and via the air between ferrets. Nature Communications, 11(1).

18. Rothe, R., Guillaumin, M. and Van Gool, L., 2015. Non-maximum Suppression for Object Detection by Passing Messages Between Windows. Computer Vision -- ACCV 2014, pp.290-306.

19. Saeidi, M. and Ahmadi, A., 2020. A novel approach for deep pedestrian detection based on changes in camera viewing angle. Signal, Image and Video Processing, 14(6), pp.1273-1281.

20. Saez, M., Tobias, A., Varga, D. and Barceló, M., 2020. Effectiveness of the measures to flatten the epidemic curve of COVID-19. The case of Spain. Science of The Total Environment, 727, p.138761.

21. Setti, L., Passarini, F., De Gennaro, G., Barbieri, P., Perrone, M., Borelli, M., Palmisani, J., Di Gilio, A., Torboli, V., Fontana, F., Clemente, L., Pallavicini, A., Ruscio, M., Piscitelli, P. and Miani, A., 2020. SARS-Cov-2RNA found on particulate matter of Bergamo in Northern Italy: First evidence. Environmental Research, 188, p.109754.

22. Sun, C. and Zhai, Z., 2020. The efficacy of social distance and ventilation effectiveness in preventing COVID-19 transmission. Sustainable Cities and Society, 62, p.102390.

23. Tang, J., Su, Q., Su, B., Fong, S., Cao, W. and Gong, X., 2020. Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition. Computer Methods and Programs in Biomedicine, 197, p.105622.

24. Tej Chinimilli, B., T, A., Kotturi, A., Reddy Kaipu, V. and Varma Mandapati, J., 2020. Face Recognition based Attendance System using Haar Cascade and Local Binary Pattern Histogram Algorithm. 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184),.

25. Thu, T., Ngoc, P., Hai, N. and Tuan, L., 2020. Effect of the social distancing measures on the spread of COVID-19 in 10 highly infected countries. Science of The Total Environment, 742, p.140430.

26. Tobías, A., 2020. Evaluation of the lockdowns for the SARS-CoV-2 epidemic in Italy and Spain after one month follow up. Science of The Total Environment, 725, p.138539.

27. WANG, Y., ZHANG, X., YE, J., SHEN, H., LIN, Z. and TIAN, W., 2020. Mask-wearing

28. recognition in the wild. SCIENTIA SINICA Informationis, 50(7), pp.1110-1120.

29. Wu, J., Tang, B., Bragazzi, N., Nah, K. and McCarthy, Z., 2020. Quantifying the role of social distancing, personal protection and case detection in mitigating COVID-19 outbreak in Ontario, Canada. Journal of Mathematics in Industry, 10(1).

30. Xiang, J. and Zhu, G., 2017. Joint Face Detection and Facial Expression Recognition with MTCNN. 2017 4th International Conference on Information Science and Control Engineering (ICISCE),.

31. Yu, W., Kim, S., Chen, F. and Choi, J., 2020. Pedestrian Detection Based on Improved Mask R-CNN Algorithm. Advances in Intelligent Systems and Computing, pp.1515-1522.

32. Zhang, Z., Gao, J., Mao, J., Liu, Y., Anguelov, D. and Li, C., 2020. STINet: Spatio- Temporal-Interactive Network for Pedestrian Detection and Trajectory Prediction. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

SOURCE CODE

Google Drive Link for Source Code:

https://drive.google.com/file/d/1EG8ccRtM6m25BV0bxeoDNWW5BW7rD1EU/view?usp=sharing

Program Code:

```
**Social distancing**, also called **“physical distancing,”** means keeping a safe space
between yourself and other people who are not from your household.

To practice social or physical distancing, stay at least 6 feet (about 2 arm lengths) from
other people who are not from your household in both indoor and outdoor spaces.
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import cv2
from scipy.spatial import distance
# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all files
under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
### Using haar cascade to detect faces

Object Detection using Haar feature-based cascade classifiers is an effective object
detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object
Detection using a Boosted Cascade of Simple Features" in 2001. It is a machine learning
based approach where a cascade function is trained from a lot of positive and negative
images. It is then used to detect objects in other images. We'll be using a Haar Cascade
Model trained to detect faces in order to obtain the bounding box coordinates of faces in
an image.

#loading haarcascade_frontalface_default.xml
```

```
face_model =
cv2.CascadeClassifier('../input/haarcascades/haarcascade_frontalface_default.xml')
```

```
import matplotlib.pyplot as plt
#trying it out on a sample image
img = cv2.imread('../input/face-mask-detection/images/maksssksksss244.png')

img = cv2.cvtColor(img, cv2.IMREAD_GRAYSCALE)

faces = face_model.detectMultiScale(img,scaleFactor=1.1, minNeighbors=4) #returns a list
of (x,y,w,h) tuples

out_img = cv2.cvtColor(img, cv2.COLOR_RGB2BGR) #colored output image

#plotting
for (x,y,w,h) in faces:
    cv2.rectangle(out_img,(x,y),(x+w,y+h),(0,0,255),1)
plt.figure(figsize=(12,12))
plt.imshow(out_img)
```

```
### Detecting social distancing violations

This can be done by iterating over the coordinates of faces and calculating the distance
for each possible pair, if the distance for a particular pair is less than MIN_DISTANCE
then the bounding boxes for those faces are colored red. MIN_DISTANCE must be manually
initialized in such a way that it corresponds to the minimum allowable distance in real
life (ex. 6ft in India).

MIN_DISTANCE = 130
if len(faces)>=2:
    label = [0 for i in range(len(faces))]
    for i in range(len(faces)-1):
        for j in range(i+1, len(faces)):
            dist = distance.euclidean(faces[i][:2],faces[j][:2])
            if dist<MIN_DISTANCE:
                label[i] = 1
                label[j] = 1
    new_img = cv2.cvtColor(img, cv2.COLOR_RGB2BGR) #colored output image
    for i in range(len(faces)):
        (x,y,w,h) = faces[i]
        if label[i]==1:
            cv2.rectangle(new_img,(x,y),(x+w,y+h),(255,0,0),1)
        else:
            cv2.rectangle(new_img,(x,y),(x+w,y+h),(0,255,0),1)
    plt.figure(figsize=(10,10))
    plt.imshow(new_img)

else:
```

```
    print("No. of faces detected is less than 2")
```

```
### Using VGG19 for mask detection
from keras.applications.vgg19 import VGG19
from keras.applications.vgg19 import preprocess_input
from keras import Sequential
from keras.layers import Flatten, Dense
from keras.preprocessing.image import ImageDataGenerator
# Data augmentation

train_datagen = ImageDataGenerator(rescale=1.0/255, horizontal_flip=True,
zoom_range=0.2,shear_range=0.2)
train_generator =
train_datagen.flow_from_directory(directory=train_dir,target_size=(128,128),class_mode='ca
tegorical',batch_size=32)

val_datagen = ImageDataGenerator(rescale=1.0/255)
val_generator =
train_datagen.flow_from_directory(directory=val_dir,target_size=(128,128),class_mode='cate
gorical',batch_size=32)

test_datagen = ImageDataGenerator(rescale=1.0/255)
test_generator =
train_datagen.flow_from_directory(directory=val_dir,target_size=(128,128),class_mode='cate
gorical',batch_size=32)
```

```
### Building VGG19 transfer learning model.
vgg19 = VGG19(weights='imagenet',include_top=False,input_shape=(128,128,3))

for layer in vgg19.layers:
    layer.trainable = False

model = Sequential()
model.add(vgg19)
model.add(Flatten())
model.add(Dense(2,activation='sigmoid'))
model.summary()
```

```
model.compile(optimizer="adam",loss="categorical_crossentropy",metrics ="accuracy")
history = model.fit_generator(generator=train_generator,
                              steps_per_epoch=len(train_generator)//32,
                              epochs=20,validation_data=val_generator,
                              validation_steps=len(val_generator)//32)
model.evaluate_generator(test_generator)

###Our model achieved 98% accuracy on test data.
```

```
### Testing the model on the test data
sample_mask_img = cv2.imread('../input/face-mask-12k-images-dataset/Face Mask
Dataset/Test/WithMask/1565.png')
sample_mask_img = cv2.resize(sample_mask_img,(128,128))
plt.imshow(sample_mask_img)
sample_mask_img = np.reshape(sample_mask_img,[1,128,128,3])
sample_mask_img = sample_mask_img/255.0
model.predict(sample_mask_img)

### The model is able to classify if the person is wearing a mask or not.
```

```
### Save the model.
model.save('masknet.h5')
```

```
### Integrating with haar cascade
##We now take crops of the faces detected in the image and use the model trained in the
above section to determine whether the individual faces have a mask or not.
mask_label = {0:'MASK',1:'NO MASK'}
dist_label = {0:(0,255,0),1:(255,0,0)}

if len(faces)>=2:
    label = [0 for i in range(len(faces))]
    for i in range(len(faces)-1):
        for j in range(i+1, len(faces)):
            dist = distance.euclidean(faces[i][:2],faces[j][:2])
            if dist<MIN_DISTANCE:
                label[i] = 1
                label[j] = 1
    new_img = cv2.cvtColor(img, cv2.COLOR_RGB2BGR) #colored output image
    for i in range(len(faces)):
        (x,y,w,h) = faces[i]
        crop = new_img[y:y+h,x:x+w]
        crop = cv2.resize(crop,(128,128))
        crop = np.reshape(crop,[1,128,128,3])/255.0
        mask_result = model.predict(crop)
        cv2.putText(new_img,mask_label[mask_result.argmax()],(x, y-
10),cv2.FONT_HERSHEY_SIMPLEX,0.5,dist_label[label[i]],2)
        cv2.rectangle(new_img,(x,y),(x+w,y+h),dist_label[label[i]],1)
    plt.figure(figsize=(10,10))
    plt.imshow(new_img)

else:
    print("No. of faces detected is less than 2")



#### Red boxes shows violation of social distancing.
```