

Diseases Assistant- Literature Survey

Disease Assistant

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

in

Electronics and Telecommunication Engineering

by

ABHISHEK KUMAR 117A2001

AGRAWAL HARSH SANJAY 117A2005

KHOT ADITI RAJAN 117A2051

MALHOTRA JAGJEET SINGH 117A2054

Under the Guidance of:

Prof. Pushkar Sathe



Department of Electronics and Telecommunication Engineering
SIES GRADUATE SCHOOL OF TECHNOLOGY
Nerul, Navi Mumbai 400706
University of Mumbai
2020-2021

CERTIFICATE

This is to certify that the project entitled **Disease Assistant** is a bonafide work of the following students, submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in Electronics and Telecommunication Engineering.

ABHISHEK KUMAR 117A2001

AGRAWAL HARSH SANJAY 117A2005

KHOT ADITI RAJAN 117A2051

MALHOTRA JAGJEET SINGH 117A2054

Prof. Pushkar Sathe

Internal Guide

Dr. Preeti Hemnani

Head of Department

Dr. Atul Kemkar

Principal

PROJECT REPORT APPROVAL

This project report entitled *Disease Assistant* by following students is approved for the degree of *Bachelor of Engineering* in *Electronics and Telecommunication*

ABHISHEK KUMAR SHREEVATS **117A2001**
AGRAWAL HARSH SANJAY **117A2005**
KHOT ADITI RAJAN **117A2051**
MALHOTRA JAGJEET SINGH **117A2054**
NARINDERPAL SINGH

Name of External Examiner: _____

Signature: _____

Name of Internal Examiner: _____

Signature: _____

Date:

Place:

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

ABHISHEK KUMAR SHREEVATS	117A2001	_____
AGRAWAL HARSH SANJAY	117A2005	_____
KHOT ADITI RAJAN	117A2051	_____
MALHOTRA JAGJEET SINGH	117A2054	_____
NARINDERPAL SINGH		

Signature

Date:

ACKNOWLEDGEMENT

It gives us immense pleasure to thank Dr. Atul Kemkar , our Principal for extending his support to carry out this project. We also thank our HoD, Dr. Preeti Hemnani for her support in completing the project. We wish to express our deep sense of gratitude and thanks to our Internal Guide, Prof. Pushkar M Sathe for his guidance, help and useful suggestions, which helped in completing our project work in time and for his guidance provided whenever required.

Also we would like to thank the entire faculty of Electronics and Telecommunication department for their valuable ideas and timely assistance in this project, last but not the least, we would like to thank our teaching and non teaching staff members of our college for their support, in facilitating timely completion of this project.

Project Team

ABHISHEK KUMAR SHREEVATS

AGRAWAL HARSH SANJAY

KHOT ADITI RAJAN

MALHOTRA JAGJEET SINGH NARINDERPAL SINGH

ABSTRACT

A disease becomes a pandemic when it is neglected and has a very high infection rate. Humans tend to ignore simple safety measures like social distancing and wearing a mask. Considering any highly infectious disease this is very detrimental. This project as a solution, introduces passive and active models. The passive model consists of Face-mask detection and Social distancing model. The Face is detected using Haar Cascade Frontal face classifier and Face Mask Detection system is built with OpenCV, Keras/TensorFlow. MobileNetV2 is used in Transfer Learning. The Social-distancing is measured using simple Euclidean Distance of bounding boxes. In order to increase the efficiency and visibility, the Bird's eye view model using Detectron 2.0 is implemented. An email is sent to the pedestrians that are dangerously close and are violating social-distancing. The passive model comprises an interactive chat-bot built using Telegram, collecting the user information and leading to the front-end booking page for medical assistance built using Python Flask. The patient can book doctor's appointments at desired timing and date as per the availability. In order to mitigate the problem of storage the project can be further developed and stored onto Microsoft Azure, increasing the availability, scalability and cost-effectiveness. The system can be modified to target specific symptoms since all of them are pre included. One of the main salient features of the project is parallel working. Multiple models, working parallelly on multiple images ensures no disagreement between models. Use of a destructor ensures that memory is saved. EDGE technology and an effective UI helps the user get the best interface possible.

Contents

		Page No.
Chapter 1	Introduction	1-3
	1.1 Introduction and need of the project	1-2
	1.2 Problem statement in broad sense	2
	1.3 Organization of the report	3
Chapter 2	Literature Review	4-5
Chapter 3	Report on Present Investigation	6-13
	3.1 Objectives	6
	3.2 System architecture and design	6-7
	3.3 Algorithm	8
	3.4 Methodology	9-11
	3.5 Model implementation	12-13
Chapter 4	Results and Discussions	14-20
	4.1 Passive Model Output	14-15
	4.2 Active Model Output	16-20
Chapter 5	Conclusion and Future scope	21
References		22
Plagiarism report		23-25
Publication details		25+

List of Figures

Figure No.	Figure Caption	Page No.
1	Project flowchart	9
2	Output of mask and social distancing models	14
3	Bird's eye view using Detectron 2.0	15
4	Email received by high-risk	16
5	CovidDiseaseAssistant ChatBot	17
6	Disease Assistant Booking Slot Website Page	18
7	Register Page of Disease Assistant Booking Slot Website	18
8	After Register page of Disease Assistant Booking Slot Website	19
9	Booking slot Day 1 Page	19
10	New Booking Page	20
11	After Booking Day1 slot by CDA-ID 234567	20

List of Abbreviations

In alphanumeric order

CDA-ID	Covid Disease Assistant Identity Document
CNN	Convolutional neural network
DDC19	Dynamic Risk assessment decision support framework for COVID-19
EDGE	Enhanced Data Rates for GSM (Global System for Mobile) Evolution
GPIO	General-Purpose Input/Output
GP	General Practitioners
SVM	Support Vector Machines
MTCNN	Multi-Task Cascaded Convolutional Neural Network (MTCNN)
UI	User interface

Chapter 1 - Introduction

1.1 Introduction and need of the project

Humans tend to neglect basic safety measures like practicing social distancing and wearing a mask to ensure safety of others. We tend to neglect basic hygiene like use of sanitizers, practicing human contact like handshakes without ensuring sanitary condition. Countless diseases are caused by microorganisms that persist by infecting new people. Cutting these detrimental interactions in public would enable us to eliminate ailments before they grow to be a pandemic. Such measures are essential considering the fact that even extremely common diseases like the common cold can make a perfectly healthy person ill purely based on an asymptomatic interaction that lasted a few seconds.

The proposed system has two independent parts referred to as the active and the passive part. The system has been designed in such a way to compensate for the fact that not every interaction calls for action. Only certain interactions and unhealthy practices need to be flagged to the authorities and the user itself. The main stages of the Disease Assistant are, data acquisition, synthesis and implementation. Hence, a CCTV camera network is used to record the human traffic in public places and detect the patterns and the proximity of one subject with another. Other than this, there are algorithms which account for the environmental conditions like temperature and humidity. Along with this, wearing a mask will account for the final “risk factor” for each individual in consideration. It consists of a camera and multiple deep learning algorithms that calculate risk factor. If the risk factor is high for any particular person, the active module comes into effect and suggests solutions and precautions along with an option to book doctor appointments.

In case the risk factor, an individual will have to submit their reports and other medical details to a flask chatbot before visiting the doctor. The interactive flask based chatbot provides an easy and effective user interface. If a medical consultation is deemed essential by the algorithms, only then will the person need to consult a doctor. For this, he will be given a list of authorised doctors in his locality and the timings on which they will be available for consultation. Booking an appointment and the update to the doctor along with all the reports will be sent automatically to ensure there is no added pressure on the medical fields and the system relieves pressure that medical practitioners face during a surge in patient numbers.

1.2 Problem statement in broad sense

A disease becomes a pandemic upon negligence and high infectivity. Thereby finding an effective and efficient solution is critical so as to avoid escalation of a communicable ailment that may gradually lead to a pandemic scenario. This system can be successfully implemented as safety and an effective preventive system. With this project we aim at developing a self sustaining system which helps relieve the pressure faced by medical practitioners in the long run, while ensuring that hygiene is maintained in public places.

In some Asian countries like Japan, it is mandated to follow the rules we're trying to enforce when a person suffers from any communicable ailments. We saw the Covid-19 virus become a worldwide pandemic, many researches suggest that COVID could be a start to many harder to deal viruses to come. Hence it is essential that in the future, these diseases are healed before gain enough infectivity to become a pandemic. The project identifies variable factors influencing public hygiene and discrepancies in the healthcare system. The system can be further modified to successfully target a specific disease very easily and ensure maximum accuracy and quick adaptation.

1.3 Organization of the report

The report is divided into five chapters. Each part deals with the different stages of the **Disease Assistant** . Each chapter has various parts explaining in detail

Chapter 2 provides a foundation of knowledge on the topic, identifies areas of prior scholarship to prevent duplication and give credit to other researchers. It also identifies inconsistencies: gaps in research, conflicts in previous studies, open questions left from other research. It also discusses the need for additional research to justify our research and identifies the relationship of works in context of its contribution to the topic and to other works. It places our current research within the context of existing literature making a case for why further study is needed.

Chapter 3 discusses the team's work on the project and step by step development. It discusses the objectives in mind while developing the system. It also discusses the system hardware architecture and software design. Next, it describes the methodology in detail of each working part of the system. And finally it describes the implementation techniques and deployment scenarios. **Chapter 4** discusses the results and output, and how they are to be deployed in the real world. It starts by talking about the passive model which is abstracted at deployment. Finally, it talks about the active model UI, implementation and UX.

Finally, **Chapter 5** concludes the thesis and discusses some future work.

Chapter 2 - Literature review

In recent years, the architectures based on the convolution neural network (CNN) results showed major improvements in performance which contribute to the high quality of object detection. A feasible method that consists of first identifying facial features has been proposed. The issue of occluded face detection was solved using the Multi-Task Cascaded Convolutional Neural Network (MTCNN). Extraction of facial features is then carried out using the embedding model of Google FaceNet embedded model. Eventually, the Support Vector Machine (SVM) has conducted the classification task.[1] Because the model offers better accuracy for simple masked face recognition and can not be appeased for all forms of masks, the proposed project has a trained custom dataset model that includes complex and many more occlusion sources, including hat, sunglasses, beard, long hair, mustache, and medicine. In static images, as well as in real time video streams, it can detect face masks.

A study[2] was suggested that uses object detection and tracking models to help deal with the worsening of COVID-19 cases in the social distancing remedy. The research uses the YOLO v3 object detection model to isolate humans from the background and to monitor the detected individuals using bounding boxes and assigned IDs using the Deepsort method. The pairwise vectorized L2 standard is later determined on the basis of the three-dimensional space of the function obtained by using the centroid coordinates and the bounding box dimensions. To measure the non-adoption of the social distancing protocol, the breach index or violation index term is suggested.

A similar project suggested a deep learning solution that would warn the user as soon as social distance is breached. A video stream is taken from the CCTV camera and the individuals are detected with the PoseNet model and then a rack is maintained of the number of people in the video stream. If the difference between 2 people's frames is less then the authorities in charge are alerted[3]. Since both approaches are highly sensitive to the spatial location of the camera and intended to be used in any working environment; accuracy and precision are highly desired to serve the purpose and to avoid discomfort and panic situations among people. The proposed project detects the social distancing and masks with a precision and confidence score that reduces the number of false positives. The passive model operates two tasks: firstly, face mask detection secondly; the detection of a social distance violation by individuals is detected continuously in threshold time by using a sliding window on a fast R-CNN to get maximum efficiency output. This solution can be used in places like temples, shopping complexes, metro stations, airports, etc.

Community participation is the first line of protection in the conflict against infectious diseases and an important role is provided by general practitioners (GPs). The study analyzed the key problems that need to be addressed, planned and developed a dynamic risk assessment decision support framework for COVID-19 (DDC19) [4] centered on the real scenes and processes of patients using health care. The developed DDC19 consists of three components: two main mobile terminal applications (patient end & GP end), a database system with related components

and a related support model underlying it. All mobile terminal devices are wirelessly linked to the back-end data center to send requests and transmit data. A multi-class logistic regression algorithm, a 10-fold cross-validation approach to evaluate and test the COVID-19 dynamic risk stratification model in various scenarios, is allocated as labels for the three groups of low risk, moderate risk and high risk. The system is still in the deployment and app stage, the relevant data of patients in the system and in-hospital visits cannot be obtained in a timely manner including the severity of symptoms and history of underlying chronic diseases and majorly does not include any real time communication to seek constant advice. While our projects aim at Python - Flask chat bot to collect essential data and seek emergency appointments at the hospital. When the risk calculation gives very high risk for a particular person, he is sent a link via mail to an AI Chatbot. The real time parameter that is symptom checker considers various parameters including temperature Screening which is another key symptom of COVID-19 infection.

Chapter 3 - Report on Present Investigation

3.1 Objectives

With this project we aim at developing a self sustaining system which helps relieve the pressure faced by medical practitioners in the long run. The project identifies variable factors influencing public hygiene and discrepancies in the healthcare system. The system can be further modified to successfully target a specific disease very easily and ensure maximum accuracy and quick adaptation. A disease becomes a pandemic upon negligence and high infectivity. This system can be successfully implemented as safety and an effective preventive system.

3.2 System Architecture and Design

A CCTV camera network / webcam (Pixel resolution is 1280 x 1024 (like the Sony SNC-EM600 1.3 megapixel camera), or it can be 1280 x 800, 1080p cameras that have at least a 2-megapixel sensor, to capture the video of the people moving in a office or open environment.

In their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001, Object Detection using Haar feature-based cascade classifiers is an efficient object detection method proposed by Paul Viola and Michael Jones. To obtain the bounding box coordinates of faces in an image, we will be using a Haar Cascade Model trained to detect faces. Face Mask Detection system built with OpenCV, Keras/TensorFlow. We use the MobilenetV2 in transfer learning. We remove the bottom few layers worth of weights from the pre-existing model and add a flatten and a dense layer with 2 output neurons specifying "with and without" masks. Next for social distancing checking, which 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 the threshold then the bounding boxes for those faces are colored red. Threshold must be manually initialized in such a way that it corresponds to the minimum allowable distance in real life (ex. 6ft in India). In addition to this, we'll use a sliding window on a fast R-CNN to get maximum efficiency output. The CNN will extract the features and the classifier - regressor stack will calculate the distance at maximum accuracy. State-of-the-art object detection networks to hypothesise object coordinates rely on area proposal algorithms. The running time of these detection networks has been reduced by developments such as SPPnet and Fast R-CNN, identifying region proposal computation as a bottleneck.

Also Detectron 2.0 was used and combined with the Haar cascade model, the Haar cascade model works effectively but gives error as it takes into account only persons standing in a straight line to overcome this, we combine it with Detectron 2.0 which gives us the birds eye view of the frame and calculates the distance between two persons effectively. The Detectron 2.0 is a collective object detection model which is an open source model built by FAIR that is the Facebook AI and Research and helps in object detection. It is implemented from the Detectron Model Zoo which has various pretrained models, transfer learning is applied on one particular model to obtain the desired results. An interactive chatbot will be built using Python-Flask. The reason we chose flask as the web development framework is that our complete system is coded in python. So any and all code will directly work on the framework without any major modifications.

The Telegram BotFather API is used to build a chatbot as it has built in features to interact with the user. When the risk calculation gives very high risk for a particular person, he is sent a link on the contact information provided to an AI Chatbot. Collecting information from the user, the chatbot will suggest best possible measures going forward and help in booking a doctor appointment if need be. There is a frontend booking page developed using flask which shows all available doctors in a particular location and the slots when they're free to be able to book an appointment without any hassle. Also the doctors can create new slots in the appointment booking portal according to their schedule.

3.3 Algorithm:

3.3.1 Passive model algorithm:

Input is taken from the camera and is used for detecting facial masks and to calculate a risk factor. The video is split into workable frames and bounding boxes are obtained using HAAR cascade. Transfer learning is applied on Mobilenet V2 to detect the presence of a facial mask. Results are obtained and objects are classified as wearing masks and not wearing masks. The Input video is split in frames and Detectron 2.0 is applied over each frame. To increase the accuracy we use a birds eye view model. The Birds eye model is used to obtain accurate object detection by generating a top view perspective of an image. The midpoint of bounding boxes is calculated with each point representing a particular person. Proximity is calculated using euclidean distance for each possible point pair. If the proximity values are below a certain threshold then they are considered as unhealthy interaction and marked as red. If the risk factor comes out to be high then the active model starts working.

3.3.2 Active model algorithm:

A link to an interactive Telegram chatbot is sent to the person having a high risk factor via mail. The user is provided with a unique CDA-ID using which the chatbot identifies the person. Basic information is collected by the chatbot regarding the patient health conditions. It recognises the patients conditions and starts providing suggestions accordingly. It even provides an option to book an appointment with a doctor. In case the user wants to book an appointment, he is asked to register using a userID and password. The available slots are displayed in green and booked ones are indicated in red. After booking a slot the booked slots are successfully reflected on the slot booking page.

3.4 Methodology

The overall framework of the proposed model is shown below:

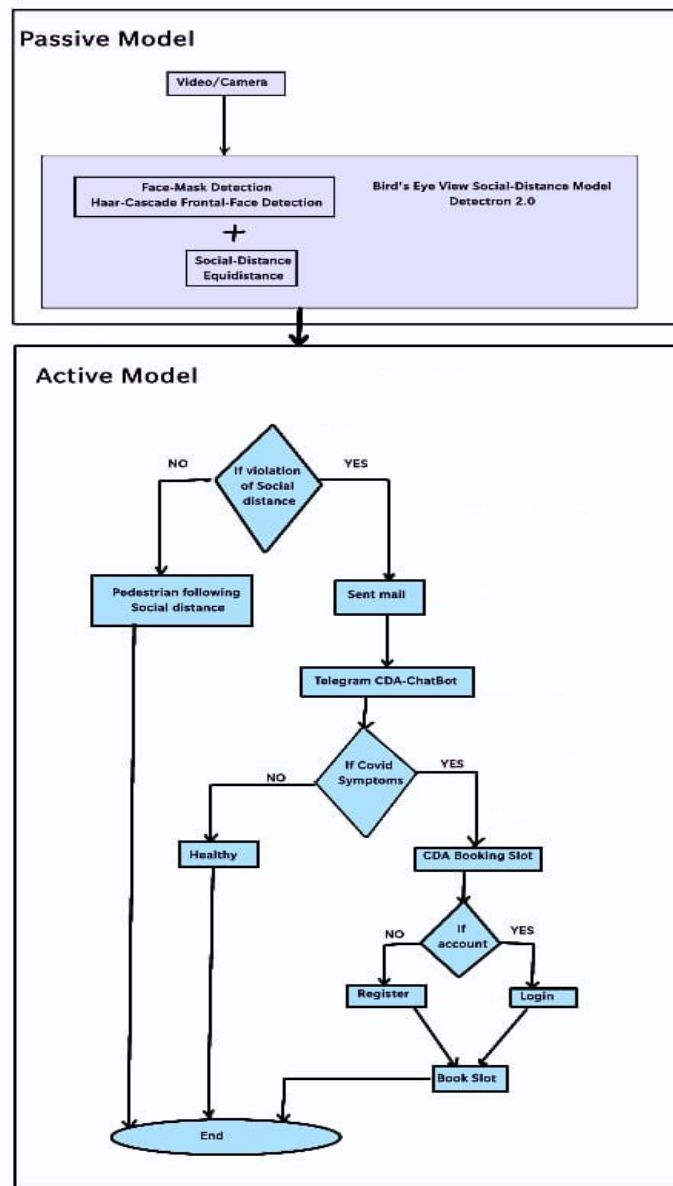


Fig 1 : Project flowchart

3.4.1 Passive model Methodology

This consists of 2 models one for facial mask recognition and one for social distancing parameters. We used Haar feature-based cascade classifiers for Object. HAAR Cascade is an object detection algorithm proposed by Paul Viola and Michael Jones to identify objects in an image or video and based on the concept of features. This is basically a machine learning based approach where a cascade function is trained using both positive and negative images. Later it is used to detect the objects in the other images. To obtain the bounding box coordinates of faces in an image, we will be using a Haar Cascade Model trained to detect faces.

We use the MobilenetV2 in transfer learning. It is a multi-layered CNN. Data sets having a very large number of images can be used in it for image classification and mobile vision. MobileNet special uses very little computation power to run or apply transfer learning to it. We remove the bottom few layers worth of weights from the pre-existing model and add a flatten and a dense layer with 2 output neurons specifying “with and without” masks. The social distancing is calculated by iterating it over the coordinates of faces and calculating the distance for each possible pair, if the distance for a particular pair is less than the threshold then the bounding boxes for those faces are colored red.

Although the Haar cascade and MobileNetV2 model gave us the required facial recognition we added a birds eye model view to further increase the accuracy. Bird Eye View transformation technique is used to generate a top view perspective of an image. This technique can be classified under digital image processing as geometrical image modification. Basically the bird's eye view has 3 parts, first we have to represent the image in a shifted coordinate system, next perform rotation of image, and then project the image on a two dimensional plane. In the beginning the entire area is divided into workable frames. Next we identify the number of persons in each frame. This is followed by finding proximity between individual subjects and marking them. If any person is found out violating the safe social distancing parameter a red box appears around them.

In addition to this we have a symptom checker which calculates a risk factor based on factors such as social and general hygiene, pre existing conditions, frequency of human interactions etc of a person. If the risk factor is above a specified safety value then the active model starts to work and the user is directed to a user friendly interactive chatbot. The passive models have been run and deployed on cloud. Specific changes can be done in order to improve the accessibility as well as scalability of the system if required.

3.4.2 Active model Methodology

Any person having a risk factor above a certain threshold is sent a mail containing the link to an interactive chatbot. It is a Telegram Chatbot implemented using BotFather. Users can interact with bots by sending them messages, inline requests and commands. One can even control the bots using HTTPS requests to the Bot Api. Each user has a unique CDA-ID using which the chatbot identifies the person. It asks a set of basic information gathering questions including the patient's health conditions. It recognises the conditions and starts providing suggestion. If the user replies as not feeling well then the person is directed to a flask based appointment booking slot .

The slot is created using Python-Flask and has an attractive frontend created using HTML and CSS. At first the user needs to register themselves using a userID, password. After registering the user has an option to choose a suitable time slot which is available. The booked slots are displayed using red slots and the available slots are shown using green. The slot editing rights remain with the admin in case of cancellation of appointments, increasing or decreasing number of slots. Booking an appointment and the update to the doctor along with all the reports will be sent automatically to ensure there is no added pressure on the medical practitioners' face during a surge in patient numbers. Once a particular slot has been booked the changes are successfully reflected on the slot booking page.

3.5 Model Implementation

3.5.1 Passive Model Implementation

Object Detection using Haar feature-based cascade classifiers is an efficient object detection method proposed by Paul Viola and Michael Jones [6]. To obtain the bounding box coordinates of faces in an image, we will be using a Haar Cascade Model trained to detect faces. Face Mask Detection system built with OpenCV, Keras/TensorFlow. We use the MobilenetV2 in transfer learning. We remove the bottom few layers worth of weights from the pre-existing model and add a flatten and a dense layer with 2 output neurons specifying “with and without” masks.

The social distancing can be checked 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 the threshold then the bounding boxes for those faces are coloured red. Threshold must be manually initialized in such a way that it corresponds to the minimum allowable distance in real life (ex. 6ft in India). In addition to this, we'll use a sliding window on a fast R-CNN to get maximum efficiency output. The CNN will extract the features and the classifier - regressor stack will calculate the distance at maximum accuracy. State-of-the-art object detection networks to hypothesise object coordinates rely on area proposal algorithms. The running time of these detection networks has been reduced by developments such as SPPnet and Fast R-CNN, identifying region proposal computation as a bottleneck.

A video consequently from the camera is read and frames are saved in the folder. The model is ready for inference after downloading the pre-trained model for object detection from Detectron 2's model zoo. The predictions are drawn on the images using Visualizer. Since a picture contains a variety of objects, we only defined classes and bounding boxes for people. After specifying the bounding boxes, we chose the bottom centre of a rectangle to represent each person in order to accurately calculate the distance, which is also invariant of a person's height. Defined a function to compute the Euclidean distance between each pair of points in an image, as well as a function to return the people who are nearest to each other based on the proximity distance. The proximity distance refers to the minimum distance between two individuals, and the proximity distance threshold is set to 100. Adjust the colour of the closest people in each picture to red. Convert the frames back to a video after recognising the closest people in - frame.

3.5.2 Active Model Implementation

In the active phase, when the passive models have calculated the most affected user, the mail will only be sent to the person with higher risk. The project is implemented in offices, schools, or colleges where the emails of the people are known and can be actively stored in the organisation's database. The mail sent to the infected person has the link to the chatbot with the consequent CovidDiseseAssistant Identity Document CDA-ID. Fig-4 shows the output of the mail received.

The Telegram Chatbot is implemented using BotFather which is the one bot that rules them all. It will assist in the development of new bots as well as the modification of existing bot's settings. The API key is accessed from the BOTFather /new chatbot. The same API key is to configure the CovidDisesaseAsisistant ChatBot. The chatbot recognises the patient with the respect to their CDA-ID and inquired for the symptoms that the patient is suffering from and consequently will provide the equivalent remedy. In case the patient is suffering from the serious illness the BookSlot Doctor's appointment website link is provided. In case there are no major symptoms an expert mail from the doctor is suggested.

The booking slot website is implemented with Bootstrap front-end framework for quicker and easy to implement web development. HTML and CSS templates are used for login, account, register, home, create_booking. The back-end is created using Python Flask, a Python-based microweb platform. It is referred to as a microframework because it does not necessitate the use of any specific resources or libraries. It doesn't have a database abstraction layer, form validation, or any other components that depend on third-party libraries to perform common tasks. The patient is required to register and login, respectively. The green slots are free and book the slot with desired timing. After the slot is booked, it will appear in red, indicating that the slot is preoccupied. The admin users, presumably the doctors and medical staff have the admin rights to add more slots or change the slot back to green once the patient is attended.

Chapter 4 - Results and Discussion

4.1 Passive Model Output

Face detection is implemented using the Haar-Cascaded Model. Face Mask Detection system built with OpenCV, Keras/TensorFlow while MobilenetV2 is used in transfer learning. Two outputs as shown below, Fig 2 shows Mask and No mask. Mask indicates that the person is wearing a mask and No mask indicates that the person is not wearing a mask, irrespective of the font color, that is green or red. The Social distance of 6 feet is indicated with a green rectangle box indicating that the person is obeying 6 feet distance and maintaining social-distance, while the red rectangle indicates that the person is not following social-distance.

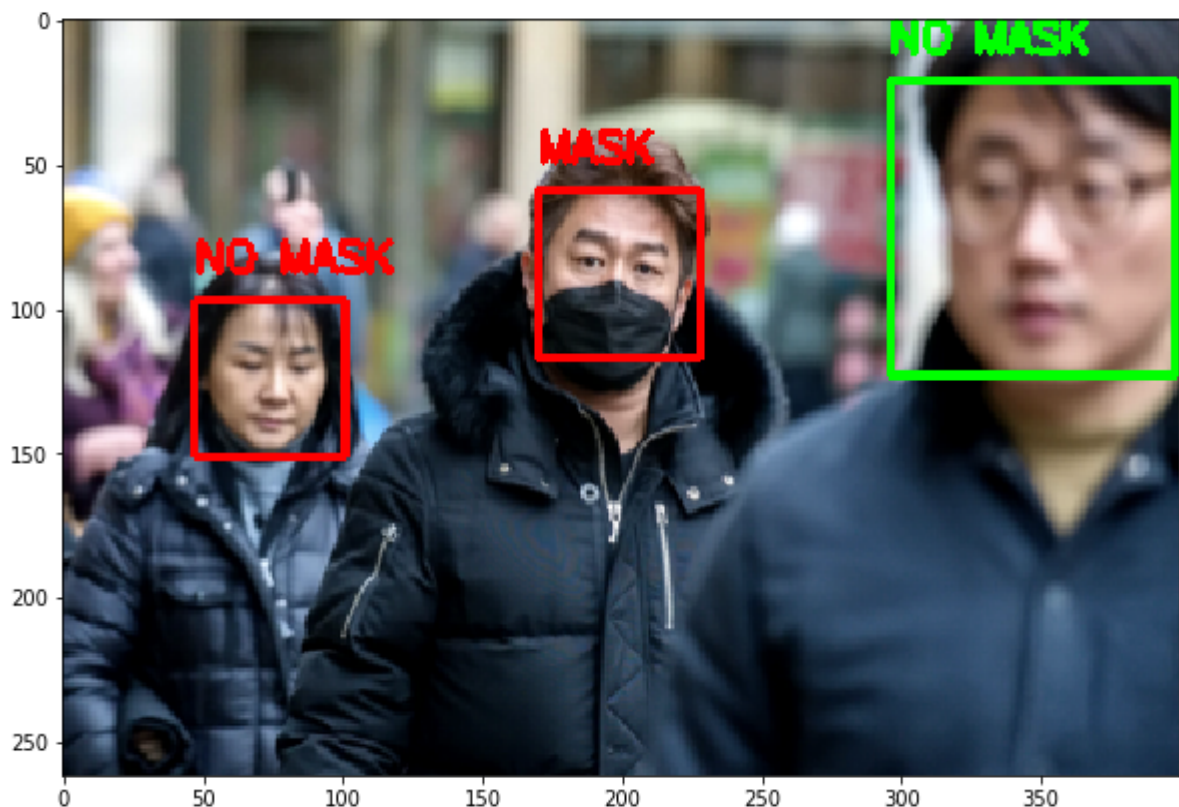


Fig 2: Output of mask and social distancing models

Bird Eye View transformation technique is used to generate a top view perspective of an image. The Bird's eye view model increases the area visibility, increasing the efficiency. The Detectron 2.0-based bird's eye view model can potentially detect where each person is in real time and return a bounding box that turns red if two people are dangerously near. The proximity distance refers to the minimum distance between two individuals, and the proximity distance threshold is set to 100. The video-stream can be fetched from any CCTV camera installed in malls, shops, and roads. As shown in Fig 3, the blue rectangle indicates that the social distance is followed, while 1 and 0 are near and are red rectangles stating that they are potentially close to each other, that is the violation of social distancing.

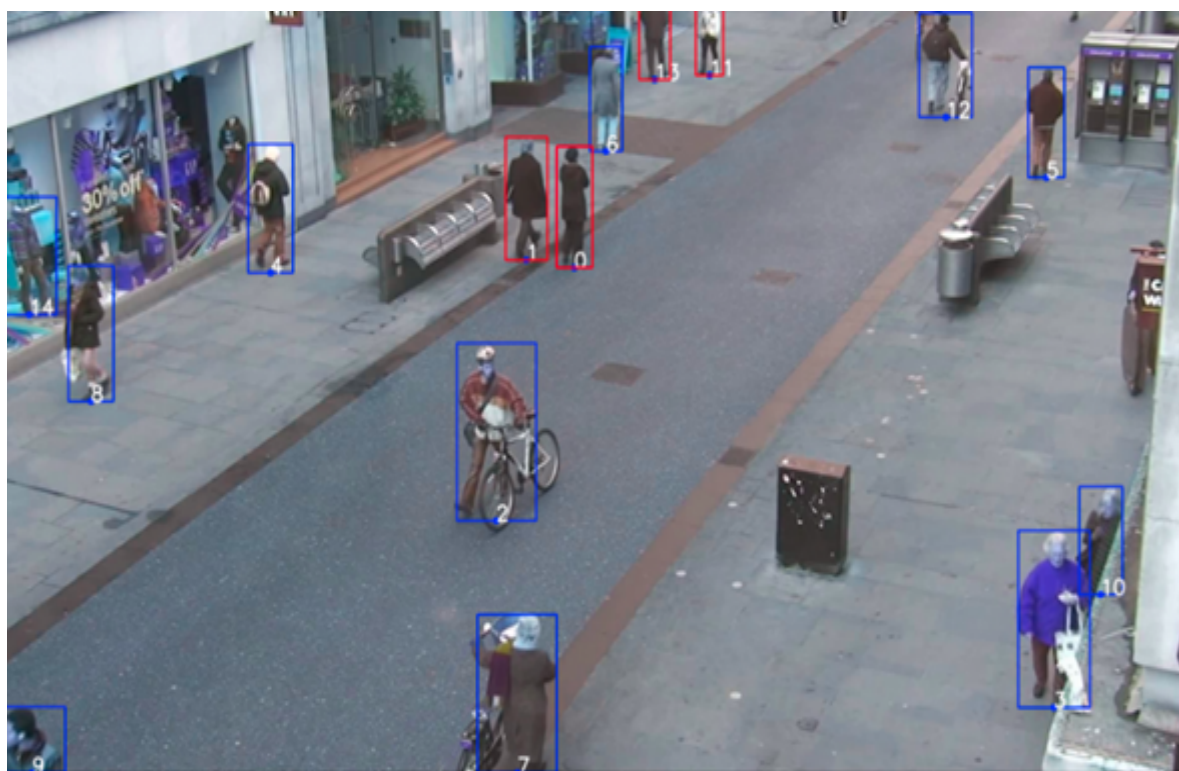


Fig 3: Bird's eye view using Detectron 2.0

4.2 Active Model Output

In the active phase, when the passive models have calculated the most affected user, the mail will only be sent to the person with higher risk. In case the social distance is violated an effective measure is taken. The mail is sent through gmail. Fig 4 shows the output of the mail received. The mail can be edited as per the requirement. The mail sent to the infected person has the link to the chatbot with the consequent CovidDisesaseAsisistant ID CDA-ID.

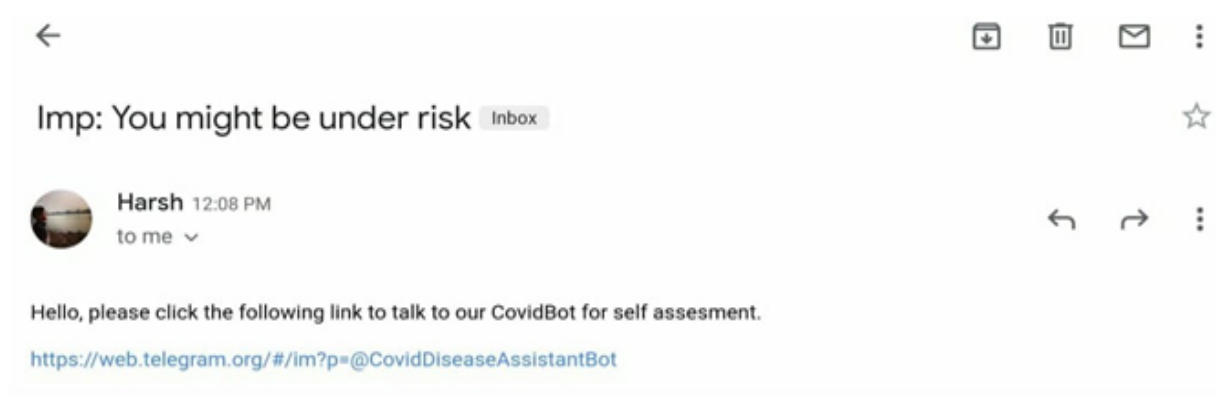


Fig 4: Email received by high-risk

Fig 5 demonstrates a sample chat with a patient, where CDA-ID number “123456” has a high risk factor and CDA-ID number “234567” is comparatively safe. The chatbot recognises the patient with the respect to their CDA-ID and inquiries for the symptoms that the patient is suffering from and consequently will provide the equivalent remedy. As shown below the bot inquires about the CDA-ID from the patient. As per the CDA-ID the person is recognised. The bot asks for healthy conditions and other symptoms, to match with the COVID-19 symptoms and take the resultant actions. If the symptoms are adverse it suggests that the patient should book an appointment for health checkup and provides the link for the CDA-BookAppointment website. In case the symptoms are mild the chat-bot suggests that “you have been home for most of the past 15 days” . Though for additional safety the CDA chatBot suggests to type “Yes” for the website link for CDA-BookAppointment.

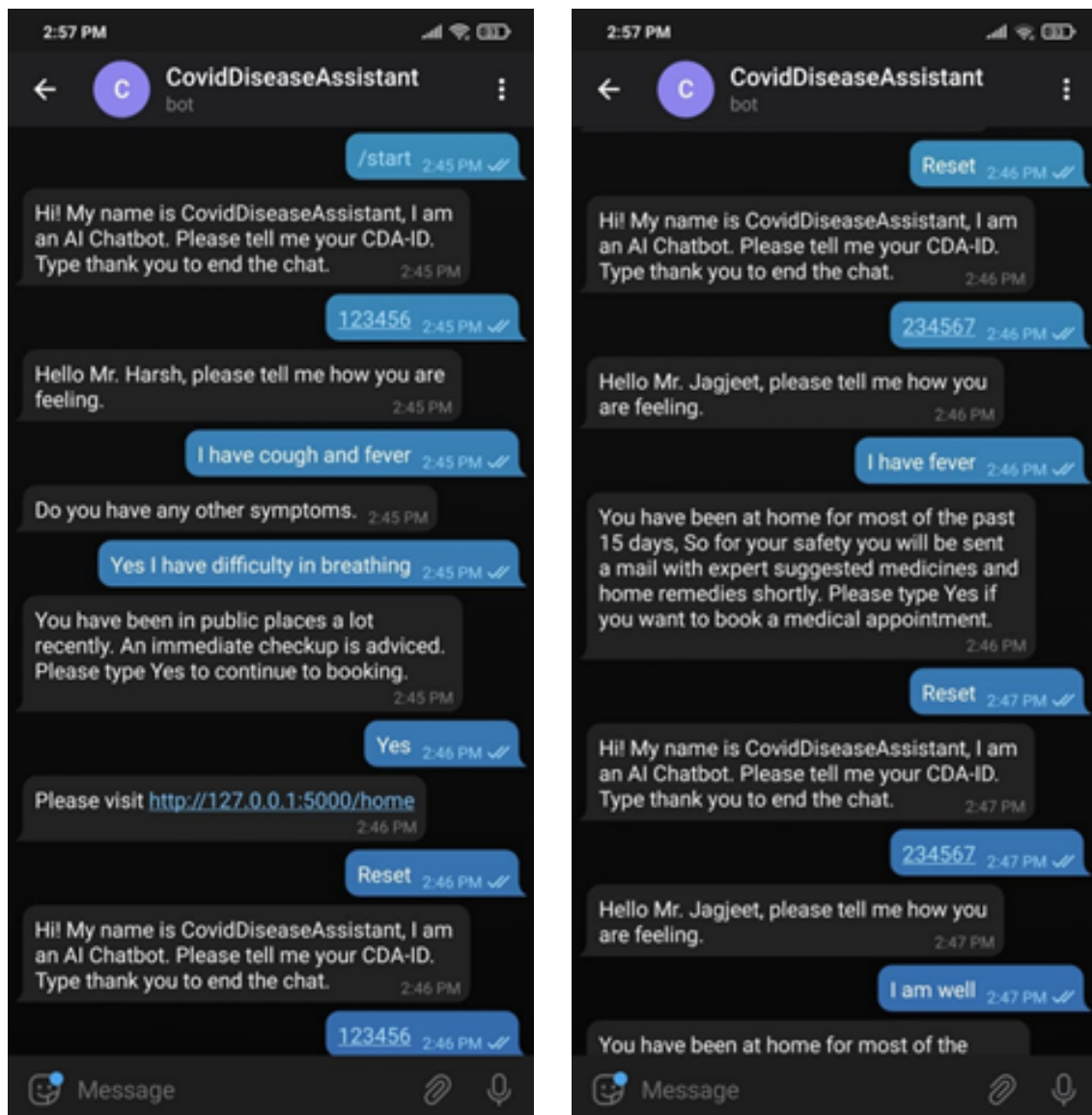


Fig 5: CovidDiseaseAssistant ChatBot

The Disease Assistant Booking Slot Website Page is shown in Fig 6. The website has a tab for “Register” and “Login”, and below we can see various days to book the appointment and time slots for individual days. Green slots indicate that they are free, while red slots indicate that they have already been booked. As shown in the figure, Day 2, 11:00 is booked by CDA-ID “12345” and hence no patient can book on that day. If the patient is new he is required to register else login by using his credentials. Fig 7 is the register page wherein the patient is required to add username and confirm password. The patient can now login with these respective username and password to track their account by clicking the “Login” tab henceforth.

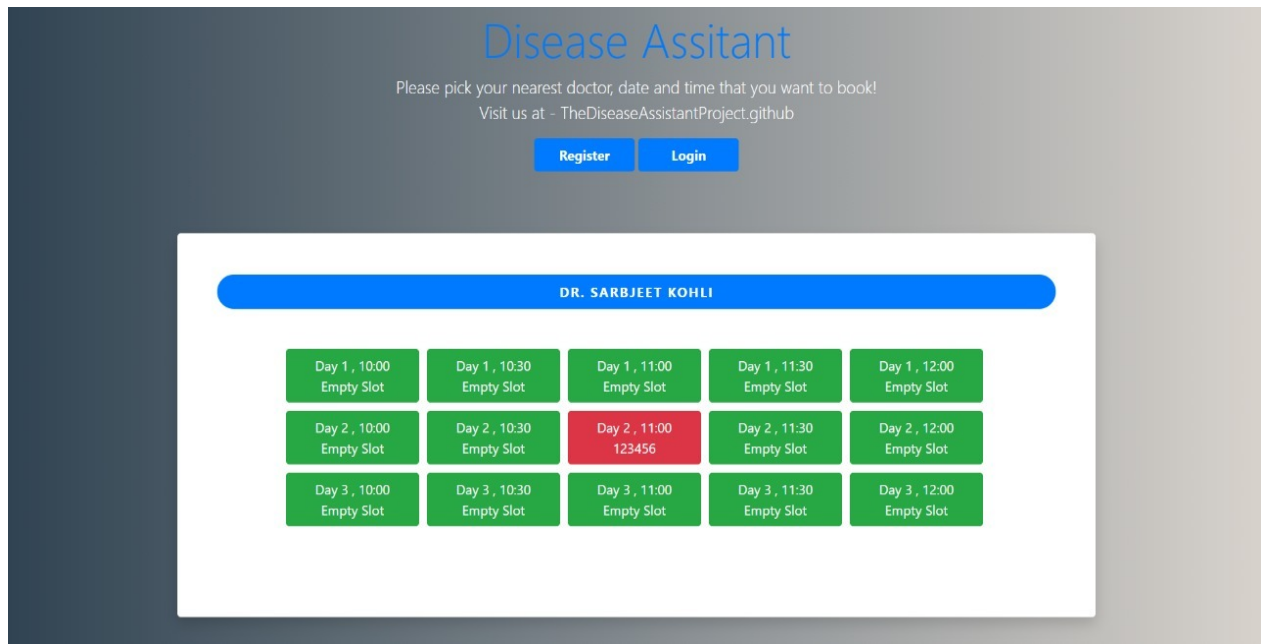


Fig 6: Disease Assistant Booking Slot Website Page



Fig 7: Register Page of Disease Assistant Booking Slot Website

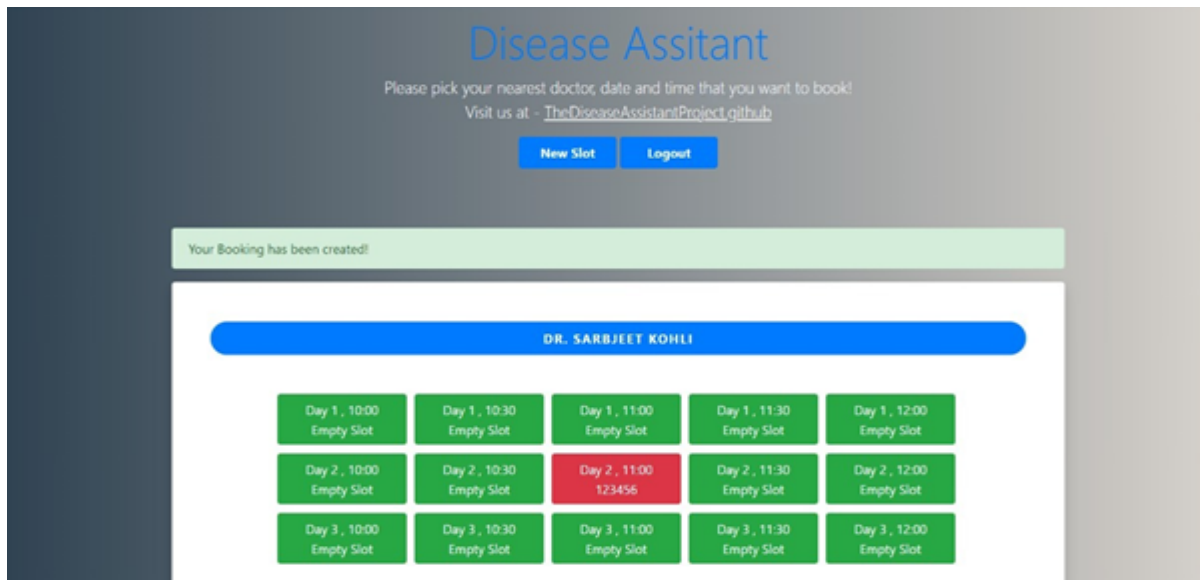


Fig 8: After Register page of Disease Assistant Booking Slot Website

Fig 8 shows the output of the after registered page. The green slots indicate that they are empty and can be booked with the desired timings whereas the red slots indicate that they have been already booked by CDA-ID '123456' on Day 2, 11:00. A particular account needs to simply click on the desired slot, as shown in Fig 9 and press the Book button. It also shows the desired slot to be booked, that is Day 1, 10:00 to be booked, and confirm by pressing Book button.

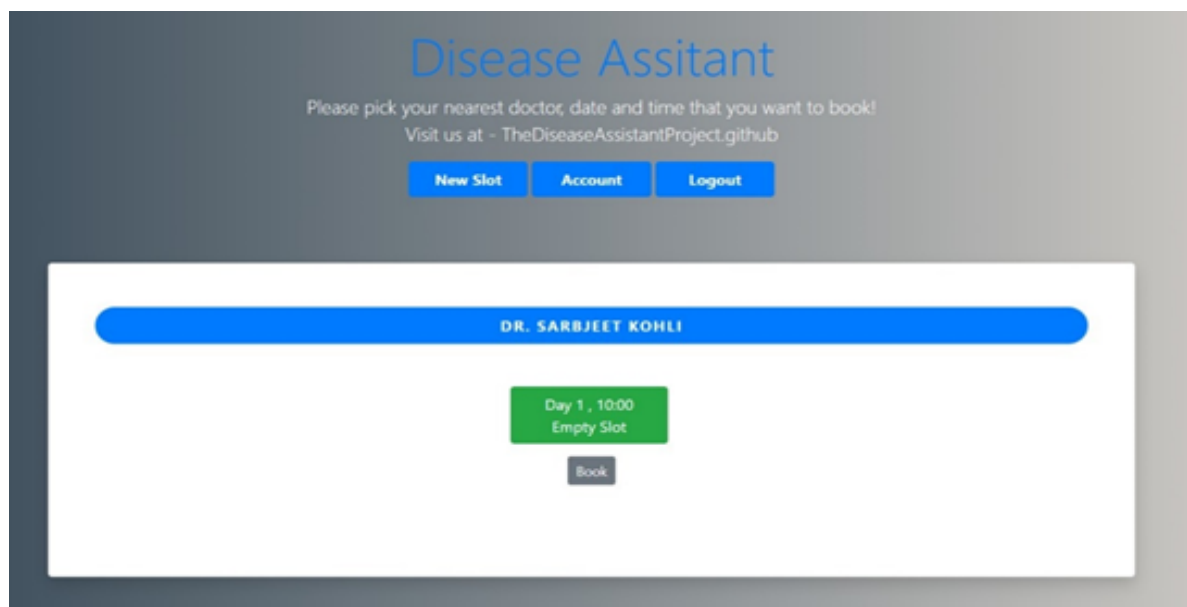


Fig 9: Booking slot Day 1 Page

When the “Book” button is pressed Fig 10 shows the New booking page. The patient needs to add the Name or CDA-ID and click on Done. Fig 11 shows the output after booking the Day 1 10:00 slot is red indicating that it is booked by CDA-ID ‘234567’. Hence the slot was successfully booked.

Disease Assitant
Please pick your nearest doctor, date and time that you want to book!
Visit us at - TheDiseaseAssistantProject.github

New Booking

Day
1

Time
10:00

Name
234567

Done

Fig 10: New Booking Page

Disease Assitant
Please pick your nearest doctor, date and time that you want to book!
Visit us at - TheDiseaseAssistantProject.github

New Slot Logout

DR. SARBJEET KOHLI

Day 1 , 10:00 234567	Day 1 , 10:30 Empty Slot	Day 1 , 11:00 Empty Slot	Day 1 , 11:30 Empty Slot	Day 1 , 12:00 Empty Slot
Day 2 , 10:00 Empty Slot	Day 2 , 10:30 Empty Slot	Day 2 , 11:00 123456	Day 2 , 11:30 Empty Slot	Day 2 , 12:00 Empty Slot
Day 3 , 10:00 Empty Slot	Day 3 , 10:30 Empty Slot	Day 3 , 11:00 Empty Slot	Day 3 , 11:30 Empty Slot	Day 3 , 12:00 Empty Slot

Fig 11: After Booking Day1 slot by CDA-ID 234567

Chapter 5 - Conclusion and Future Scope

The proposed disease assistant model is successful in eliminating social distancing as well as facial detection problems. It successfully recognized objects using the Haar Cascade trained model and gave an accurate proximity measure. The MobileNetV2 and Transfer learning used gives the best results for feature extraction. The accuracy of the system combining both the facial detection and social distancing model comes out to be high. The bird's eye view model is successfully implemented using Detectron 2.0. It also consists of a passive approach used to calculate the risk factor based on various risk parameters. The Telegram Chat-bot used makes the interfacing extremely user friendly. It also directs the user to an easy appointment booking software. This model will help in preventing infectious diseases, increasing general hygiene as well as prove to be extremely helpful in pandemic scenarios.

Disease Assistant algorithms will be applied parallelly to multiple images of the same place and the results will be combined to get the most precise output as well as parallel analysis ensures no disagreement is caused between models. The system uses very less database power, as it clears memory once the object is out of scope. Day to day functions are handled by the processor on a local computation environment (EDGE technology) which helps speed up the system. The system can be controlled and edited using a very interactive UI which is simple to understand by the end user. The feedback of the effectiveness of the solution will update the database, so that our technology adapts to the changes in the pattern of spreading of diseases under different conditions.

The system can be used for targeting the common communicable ailments and ensuring general hygiene is maintained in public places where social interaction is at a peak. It can be further modified to successfully target a specific disease very easily and ensure maximum accuracy and quick adaptation since all the major parameters are accounted for in the base system. A disease becomes a pandemic upon negligence and high infectivity. This system can be successfully implemented as safety and an effective preventive system.













References

- [1] M. S. Ejaz and M. R. Islam, "Masked Face Recognition Using Convolutional Neural Network," 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, 2019, pp. 1-6, doi: 10.1109/STI47673.2019.9068044.
- [2] Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques N. S. Punna, S. K. Sonbhadra, S. Agarwal, Indian Institute of Information Technology Allahabad, Jhalwa, Prayagraj, Uttar Pradesh, India; emails: {pse2017002, rsi2017502,sonali} @iiita.ac.in.
- [3] Ghorai, Arnab and Gawde, Sarah and Kalbande, Dhananjay, Digital Solution for Enforcing Social Distancing (May 31, 2020). Proceedings of the International Conference on Innovative Computing & Communications (ICICC) 2020, Available at SSRN: <https://ssrn.com/abstract=3614898> or <http://dx.doi.org/10.2139/ssrn.3614898>
- [4] Liu Y, Wang Z, Tian Y, Zhou M, Zhou T, Ye K, Zhao Y, Qiu Y, Li J, Ren JA COVID-19 Risk Assessment Decision Support System for General Practitioners: Design and Development Study J Med Internet Res 2020;22(6):e19786.
- [5] Viola, Paul & Jones, Michael. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE Conf Comput Vis Pattern Recognit. 1. I-511. 10.1109/CVPR.2001.990517.
- [6] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, 2001, pp. I-I, doi: 10.1109/CVPR.2001.99051.

Document Information

Analyzed document	G1 Blackbook.pdf (D104490539)
Submitted	5/11/2021 7:18:00 AM
Submitted by	Pushkar Sathe
Submitter email	pushkar.sathe@siesgst.ac.in
Similarity	6%
Analysis address	pranavi.mhatre.sies@analysis.urkund.com

Sources included in the report

W	URL: http://blog.entegraty.com/b4g9eu/face-detection-matlab.html Fetched: 12/22/2020 9:26:28 AM		2
W	URL: http://amicusdh.org/shopify-themes/face-mask-detection-tensorflow.html Fetched: 12/3/2020 7:13:45 AM		1
W	URL: http://ijraset.com/files/serve.php%3FFID%3D30560 Fetched: 12/2/2020 8:42:29 AM		2
W	URL: http://norforgo.com/mlive-unlimited-wlkro/best-object-detection-algorithm.html Fetched: 1/7/2021 5:42:47 AM		5
W	URL: http://halfspoons.com/x32-midi/face-mask-opencv.html Fetched: 11/17/2020 10:47:06 AM		2
W	URL: https://ssrn.com/abstract=3614898 Fetched: 5/11/2021 7:19:00 AM		2
W	URL: http://ipiex.cl/gmail-not-jeodt/object-detection-library.html Fetched: 1/4/2021 9:52:47 AM		1
W	URL: https://www.analyticsvidhya.com/blog/2020/05/social-distancing-detection-tool-deep ... Fetched: 5/11/2021 7:19:00 AM		2
W	URL: https://thesai.org/Downloads/Volume11No11/Paper_6-Definition_of_Unique_Objects.pdf Fetched: 5/11/2021 7:19:00 AM		1
W	URL: https://www.researchgate.net/publication/344173985_Face_Mask_Detector Fetched: 12/26/2020 3:50:47 PM		1
W	URL: https://arxiv.org/pdf/2005.05287 Fetched: 11/27/2020 7:01:33 AM		1
W	URL: https://journals.plos.org/plosone/article%25Fid%25D10.1371/journal.pone.0247440 Fetched: 5/11/2021 7:19:00 AM		1