**MINI PROJECT REPORT**

**ON**

# FACE MASK AND SOCIAL DISTANCING DETECTION SYSTEM

Submitted in partial fulfilment of the requirement for the award of degree in

## MASTER OF COMPUTER APPLICATIONS

of the

## APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted by

**HARSHIL S (NCE21MCA-2025), SIDHARTHAN V U (NCE21MCA-2044),
ASWATHY VENU (NCE21MCA-2013), FATHIMA THAHASIM U (NCE21MCA-2021),
NASREEN PARAYIL (NCE21MCA-2034)**

Under the guidance of

**Ms. DEEPA A, MCA,** Assistant Professor

Department of MCA, NCERC, Pampady, Thrissur

**DEPARTMENT OF MCA**

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE,**

**(NAAC Re-Accredited with "A" grade) PAMPADY,**

**THIRUVILWAMALA, THRISSUR-680588 NOVEMBER**

**2022**

MINI PROJECT REPORT

ON

# FACE MASK AND SOCIAL DISTANCING DETECTION SYSTEM

Submitted in partial fulfilment for the requirement for the award of degree in

## MASTER OF COMPUTER APPLICATIONS

of the

## APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted by

**HARSHIL S (NCE21MCA-2025), SIDHARTHAN V U (NCE21MCA-2044),**
**ASWATHY VENU (NCE21MCA-2013), FATHIMA THAHASIM U (NCE21MCA-2021),**
**NASREEN PARAYIL (NCE21MCA-2034)**

**Semester 3 MCA (2021-23)**

Under the guidance of

**Ms. DEEPA A, MCA,** Assistant professor

Department of MCA, NCERC, Pampady, Thrissur

**DEPARTMENT OF MCA**

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE,**

**JULY 2022**

# NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE
# DEPARTMENT OF MCA

## COLLEGE VISION

To mould true citizens who are millennium leaders and catalysts of change throughexcellence in education.

## COLLEGE MISSION

NCERC is committed to transform itself into a centre of excellence in Learning and Research in Engineering and Frontier Technology and to impart quality education to mould technically competent citizens with moral integrity, social commitment and ethical values. We intend to facilitate our students to assimilate the latest technological know-how and to imbibe discipline, culture and spiritually, and to mould them in to technological giants, dedicated research scientists and intellectual leaders of the country who can spread the beams of light and happiness among the poor and the underprivileged.

## DEPARTMENT VISION

To create a school of distinction for the PG students, prepare them to be industry - ready, and achieve Academic excellence by continuous endorsement of the faculty team in terms of Academics, Applications & Research.

## DEPARTMENT MISSION

The Department of Computer Applications strives to provide quality and competency-based education and fine-tune the younger generation through Curricular, Co-Curricular and Extra-curricular activities so as to encounter the Professional and Personnel challenges ahead with Pragmatic skills & courage, thereby ‘Creating the True Citizens’.

# DECLARATION

I hereby declare that the project entitled "**FACE MASK AND SOCIAL DISTANCING DETECTION SYSTEM**" submitted to the **Department of MCA at Nehru College of Engineering and Research Centre** in partial fulfilment of the requirement for the award of degree in **MASTER OF COMPUTER APPLICATIONS** from **APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY,** is a record of original work done by us under the guidance of **Ms. DEEPA A ,** Assistant Professor of the Department of MCA, during my Third Semester MCA course period 2022.

**HARSHIL S**<br>
**SIDHARTHAN V U**
**ASWATHY VENU**
**FATHIMA THAHASIM U**
**NASREEN PARAYIL**

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE, PAMPADY**

# CERTIFICATE

This is to certify that, the project work entitled "**FACE MASK AND SOCIAL DISTANCING DETECTION SYSTEM**" has been presented by **HARSHIL S (NCE21MCA-2025), SIDHARTHAN V U (NCE21MCA-2044), ASWATHY VENU (NCE21MCA-2013), FATHIMA THAHASIM U (NCE21MCA-2021), NASREEN PARAYIL (NCE21MCA-2034)** of Third Semester MCA in partial fulfilment of the requirement for the award degree **MASTER OF COMPUTER APPLICATIONS, APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**.

We also certify that the work done is original.

**Project Guide**

**Head of the Department**

**Principal**

**External Examiner**

# ACKNOWLEDGEMENT

First and most, I thank the **God Almighty** for showing me the path to the completion of project work. I thank **Dr. Ambikadevi Amma T**, the principal NCERC, for providing good atmosphere for project completion and presentation. I thank **Dr. Sudheer S Marar**, Head of MCA Department, NCERC for his valuable suggestions and support throughout the project work. I wish to acknowledge my deep sense of gratitude and heartfelt thanks to My project Guide **Ms. Deepa A**, Assistant professor, MCA Department for her valuable suggestions, precious time, kind hearted motivations throughout the project work period. I express my immense gratitude to all my friends, without whom I would have never been able to do my project well.

# ABSTRACT

In addressing the world pandemic situation, one of the most important practices in these outbreaks is to ensure a safe distance between people in public and wear mask properly. For social distancing, people are detected in areas of interest using the Mobilenet Single Shot Multibox Detector (SSD) objecting taking model and OpenCV library for image processing. With the detection of unsafe distance between people, alerts or warnings can be issued to keep the distance safe. In addition to social distance measure another key feature of the system is detecting the presence of people in restricted area, which can also be used to trigger warnings. For face mask detection a Retinal Facemask, a high accuracy and efficient face mask detectors, which consists of a feature pyramid network to fuse high-level semantic information with multiple feature maps and a novel context attention module to focus on the detecting face masks.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background

In addressing the worldwide pandemic situation, one of the most important practices in these outbreaks is to ensure a safe distance between people in public. This presents the detection of people with social distance monitoring as a precautionary measure in reducing physical contact between people. This focuses on detecting people in areas of interest using the Mobilenet Single Shot Multibox Detector (SSD) objecting taking model and OpenCV library for image processing. The distance will be computed between the persons detected in the captured footage and then compared to a fixed pixel values. The distance is measured between the central points and the overlapping boundary between persons in the segmented tracking area. With the detection of unsafe distance between people, alerts or warnings can be issued to keep the distance safe. In addition to social distance measure another key feature of the system is detecting the presence of people in restricted area, which can also be used to trigger warnings. Some analysis has been performed to test the effectiveness of the program for both purposes. From the results obtained, the distance tracking system achieved between 56.5% to 68% accuracy was achieved for the controlled environment on indoor testing. Whereas for the safety violation alert feature based on segmented ROI, it was found to have achieved better accuracy between 95.8% to 100% for all tested input videos. Pandemic has affected the world seriously one major protection method is for people to wear masks in public areas.

### 1.1.1 Motivation

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 abovementioned 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.

## 1.2 Objective

This research paper proposes a novel method of an Artificial Intelligence-based real-time social distancing and face mask detection system. In this research paper, pre-trained deep neural network models such as ResNet Classifier, DSFD, and YOLOv3 are used in the detection of individuals and face masks with the help of bounding boxes. The system also measures the number of people present in the camera footage.

Thus, this paper contributes to the following sections:

- Developing a novel and unique implementation of a real-time based social distancing detection as well as face mask detection system.
- Applying various deep learning models to get a more fast and accurate result.
- Distancing Analyses carried out on a less crowded region such as the Oxford Town Centre data set.

## 1.3 Contribution

The major contributions in this project are:

1. It can show if face mask worn.
2. It can show who are violating social distancing.

## 1.4 Report Organization

The project report is divided into six sections. Section 2 describes literature survey. Section 3 describes the methodology and section 4 describes agile methodology used for implementing the project. Section 5 gives the results and discussions. Finally Section 6 gives the conclusion.

# Chapter 2

# Literature Survey

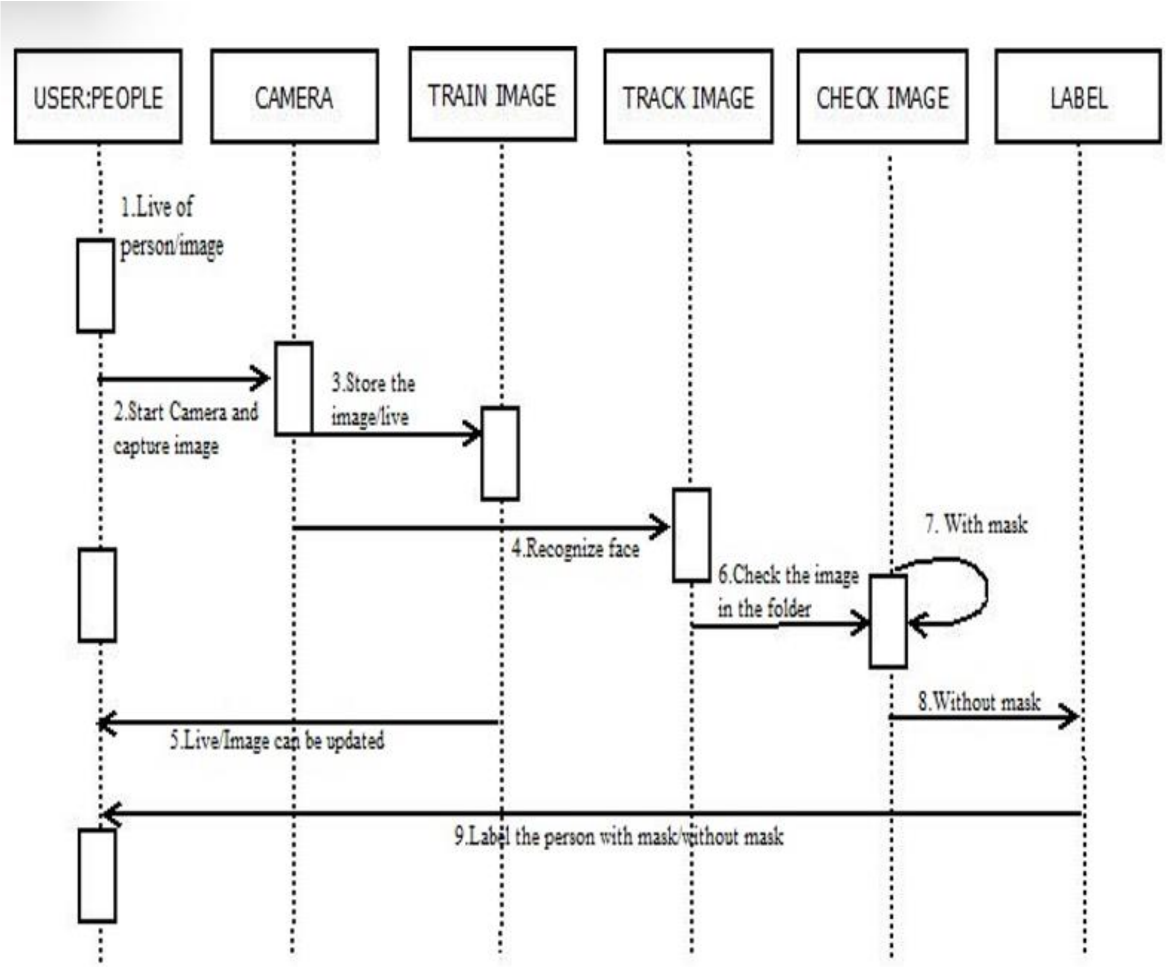
The purpose of the project “Face mask Detection and Social Distancing” is to create a tool that identifies the image of a human that can calculate the probability that he/she is wearing a mask or not and another motive of our project is to detect the people and check whether they are maintaining social distancing norms. Using Machine Learning and Object Detection, this study seeks to detect face masks and social separation. The contribution is a strategy for merging more complex classifiers in a “\cascade\” that allows the image’s background areas to be swiftly rejected while more computation is spent on promising object-like regions.

While deep learning-based methods for general object detection have progressed dramatically in the previous two years, most approaches to face detection still use the CNN framework, which results in low accuracy and processing performance. We examine the use of MobileNet (pretrained model) and feature extraction in this project, which will yield outstanding results on a variety of object detection benchmarks.

YOLO Object Detection was used to detect persons in a frame and calculate the distance between them to check for social distancing. As a result, in a real-time situation, this technology tracks persons wearing or not wearing masks and provides social separation by triggering an alert if there is a violation in the scene or public spaces. This can be combined with current embedded camera infrastructure to allow these analytics, which can be used in a variety of verticals as well as in offices and airport terminals/gates.

# Chapter 3

# Methodology



## 3.1 Data Collection and Preparation

This research paper uses a two pedestrian videos, the first is obtained online from YouTube4 which is provided by BriefCam5 and the other video footage of the Oxford Town Centre6. 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.

## 3.2 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.

# Chapter 4

# Agile Methodology

## 4.1 Introduction

After the initial studies it is found that agile model of software development is suitable and is the best method for the development of this system. Agile methodology mainly focused on the client satisfaction through continuous delivery. Also it sets a minimum number of requirements and turns them in to a deliverable product. As this project has many individual requirements which can be delivered in parts and the user can gradually improve their work efficiency. Agile methodology has a family of methods of which scrum is selected for the development of this project. Scrum is process framework that has been used to manage complex product development. It is not a process or technique for building products rather it is a framework within which various processes can be employed. Also it is suitable method to support the development process. It focuses on lean software development and has in building better software effectively and efficiently.

Agile is one of the most widely used and recognized software development framework. The methodology those experts agreed upon was described as ‘lightweight’ and fast. Agile is also about being the adaptive and continuous improvement, as much as it is about constant feedback and speed of delivery.

Agile is a software development approach where a self-sufficient and cross-functional team works on making continuous deliveries through iterations and evolves throughout the process by gathering feedback from the end users. The major rules in scrum methodology are

1. **The product owner (PO)**: Who represents the stakeholder and the business.
2. **The scrum master**: Ensures the process followed, removes obstructions, and protects the development system.
3. **Development team**: Cross functional, self-organising team who actually do the actual analysis, design implementation and testing process.

They work together in iterative time boxed durations called sprints. The first step is the creation of the product backlog by the PO. It’s a to-do list of stuff to be done by the scrum team. Then the scrum team selects the top priority items and tries to finish them within the time box called a sprint. An easier way to remember all of this is to memorize the 3-3-5 frame work. It means that a scrum project has 3 roles, 3 artifacts, and 5 events.

**These are: -**

1. **Roles**: Product Owner, Scrum Master, and development team.

2. **Artifacts**: Product Backlog, Sprint Backlog and Product Increment.

3. **Events**: Sprint, Sprint planning, Daily Scrum, Sprint review and Sprint retrospective.

The framework begins with a simple premise start with what can be seen or known. After that the progress is tracked and tweak as necessary. The three pillars of scrum are transparency, inspection and adaptation. In scrum everyone has a role.

The Git is used as the version control system for this project. Version control is a system that records changes to a file or set of files over time so that a specific version can be recalled later. Version control systems are a category of software tools that help a software team for managing changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

## 4.2 User Story

A key component of agile software development is putting people first, and user-stories put actual end users at the centre of the conversation. Stories use non-technical language to pro- vide context for the development team and their efforts. After reading a user story, the team knows why they are building what they’re building and what value it creates. A user story is a tool used in agile software development to capture a description of a software feature from an end-user perspective. The user story describes the type of user, what they want and why. A user story helps to create a simplified description of a requirement. User stories are one of the core components of an agile program. They help provide a user-focused framework for daily work — which drives collaboration, creativity, and a better product overall. The user story of system is given in Table 4.1

| User Story ID | As a <type of user> | I want to<br><perform some task> | So that I can<br><achieve some goal> |
|---|---|---|---|
| 1 | User | Detection | Detects objects in the input video |
| 2 | User | Distance calculation | Estimate the person's location in the frame |
| 3 | User | Prediction | Estimate the person's location in the frame |
| 4 | User | Pre-processing | Selecting the useful features and leaving the extra features |
| 5 | User | Recognition | Search faces and localize them in a bounding box. |

Table 4.1: User Story

## 4.3 Product Backlog

A product backlog is a list of the new features, changes to existing features, bug fixes, infrastructure changes or other activities that a team may deliver in order to achieve a specific outcome. The product backlog is the single authoritative source for things that a team works on. That means that nothing gets done that isn't on the product backlog. Conversely, the presence of a product backlog item on a product backlog does not guarantee that it will be delivered. It represents an option the team has for delivering a specific outcome rather than a commitment.

It should be cheap and fast to add a product backlog item to the product backlog, and it should be equally as easy to remove a product backlog item that does not result in direct progress to achieving the desired outcome or enable progress toward the outcome. The Scrum Product Backlog is simply a list of all things that needs to be done within the project. It replaces the traditional requirements specification artifacts. These items can have a technical nature or can be user-centric e.g., in the form of user stories.

The product backlog of the system is given in Table 4.2

| User Story ID | Priority <High / Medium / Low> | Size (Hours) | Sprint <#> | Status <Planned / In progress / Completed> | Release Date | Release Goal |
|---|---|---|---|---|---|---|
| 1 | High | 9 | 1 | Done | 08/08/2022 | Detects objects in the input video |
| 2 | Medium | 8 | | Done | 13/08/2022 | Give response back to user |
| 3 | Medium | 7 | 2 | Done | 29/09/2022 | Extract the content from the user |
| 4 | High | 7 | | Done | 02/10/2022 | Checks social distancing and facemask |

Table 4.2: Product Backlog

## 4.4 Project Plan

A project plan that has a series of tasks laid out for the entire project, listing task durations, responsibility assignments, and dependencies. Plans are developed in this manner based on the assumption that the Project Manager, hopefully along with the team, can predict up front everything that will need to happen in the project, how long it will take, and who will be able to do it. Project plan is given in Table 4.3

| User Story ID | Task Name | Start Date | End Date | Days | Status (to be filled by scrum master) |
|---|---|---|---|---|---|
| 1 | Sprint 1 | 31/03/2022 | 08/09/2022 | 14 | Completed |
| 2 | | 09/04/2022 | 13/09/2022 | | Completed |
| 3 | Sprint 2 | 19/04/2022 | 29/09/2022 | 14 | Completed |
| 4 | | 30/04/2022 | 02/10/2022 | | Completed |

Table 4.3: Project Plan

The Project has two sprints:

### 4.4.1 Sprint 1: Speech Recognition and API calls to user

Two tasks are planned in this sprint. One is to recognize the sound of the user and give response back to the user. The designing of UIS, coding, testing and validation of these functionality is be completed.

### 4.4.2 Sprint 2: Content Extraction and Text to speech

Plan for two major functionalities is Extract the content from the user and convert text to speech these are the two functionalities are made in this sprint. The UI designing, table designing, coding, testing also planned to complete.

## 4.5 Sprint Backlog (Plan)

The sprint backlog is a list of tasks identified by the Scrum team to be completed during the Scrum sprint. During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories, and identifies the tasks necessary to complete each user story. Most teams also estimate how many hours each task will take someone on the team to complete.

**Sprint 1:** This sprint has two goals. One is to recognize the sound of the user and give response back to the user. The designing of UIS, coding, testing and validation of these functionality is be completed. Sprint backlog (planning) for sprint 1 is given Table 4.4

**Sprint 2:** In this sprint two functionalities are performed one is to extract the content from the user and other one is convert the text to speech format. . Sprint backlog (planning) for sprint 1 is given Table 4.5

| Backlog item | Completion date | Original estimate in hrs. | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 11 | Day 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| User story#1hrs. | | | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. |
| UI Design | 30/08/2022 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Coding | 07/09/2022 | 4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Testing | 14/09/2022 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| Total | | 14 | 2 | 2 | 2 | 2 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

Table 4.5.1: Sprint backlog(plan)-sprint1

| Backlog item | Completion date | Original estimate in hrs. | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 11 | Day 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| User story#1hrs. | | | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. |
| Coding | 26/09/2022 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Testing | 09/10/2022 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| Total | | 11 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |

Table 4.5.2: Sprint backlog (plan) - sprint 2

## 4.6 Sprint Backlog (Actual)

Actual sprint backlog is what adequate sprint planning is actually done by project team there may or may not be difference in planned sprint backlog. The detailed sprint backlog (Actual) is given below.

| Backlog item | Completion date | Original estimate in hrs. | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 11 | Day 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| User story#1hrs. | | | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. |
| UI Design | 30/08/2022 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Coding | 07/09/2022 | 4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Testing | 14/09/2022 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| Total | | 14 | 2 | 2 | 2 | 2 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

Table 4.6.1: sprint backlog (actual) – sprint 1

| Backlog item | Completion date | Original estimate in hrs. | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 11 | Day 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| User story#1hrs. | | | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. | Hrs. |
| Coding | 26/09/2022 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Testing | 09/10/2022 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| Total | | 11 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |

Table 4.6.2: Sprint Backlog(actual)-sprint 2

## 4.7 Product Backlog Review

REVIEW FORM

SPRINT 1

| User Story ID | Comments From<br>Scrum master if any | Comments From<br>Product Owner if any |
|---|---|---|
| 1 | Developer should have easily recognize the face mask and social distancing | Face mask and social distance should be easily recognize d |
| 2 | Give response back to user | Easily give response back |

Table 4.8: Product backlog review sprint 1

REVIEW FORM

SPRINT 2

| User Story ID | Comments From<br>Scrum master if any | Comments From<br>Product Owner if any |
|---|---|---|
| 3 | System can identify face mask and social distancing | Easily identify face mask and social distancing |
| 4 | Give the required output to the user | Give required output |

Table 4.9: Product backlog review sprint 2

## 4.8 Sprint Review

At the end of each sprint a Sprint Review meeting is held. During this meeting the Scrum Team shows which Scrum Product Backlog items they completed (according to the Definition of Done) during the sprint. This might take place in the form of a demo of the new features. Backlog items that are not completed shall not be demonstrated. Otherwise this might suggest that these items are finished as well. Instead, incomplete items/remaining activities shall be taken back into the Scrum Product Backlog, re-estimated and completed in one of the following sprints. The

Sprint Review meeting should be kept very informal. No PowerPoint slides should

be used and time for preparation and performing the meeting should be limited. During the meeting the Scrum Product Owner inspects the implemented backlog entries and accepts the solution or adds new stories to the Scrum Product Backlog to adapt the functionality. Participants in the sprint review typically include the Scrum Product Owner, the Scrum Team and the Scrum Master. Additionally, management, customers, and developers from other projects might participate as well.

REVIEW FORM

SPRINT 1

| User Story ID | Comments From Scrum master if any | Comments From Product Owner if any |
|---|---|---|
| 1 | Recognize the social distancing and face mask successfully completed | Should consider validation |
| 2 | System response back to user successfully completed | Satisfying result |

Table 4.10: Sprint review sprint 1

REVIEW FORM

SPRINT 2

| User Story ID | Comments From Scrum master if any | Comments From Product Owner if any |
|---|---|---|
| 3 | Detected social distancing and face mask successfully | Satisfying result |
| 4 | Give required output successfully completed | Satisfying result |

Table 4.11: Sprint review sprint 2

## 4.9 Testing and Validation

According to Agile approaches, a working product is supposed to be delivered at the end of the Sprint, which means that it is tested and accepted by the Product Owner: from a testing perspective this means that all tests (unit test, system test, acceptance test) take place during the Sprint.

Sprint 1

| Test # | Date | Action | Expected Result | Actual Result | Pass? <Yes/ N o> |
|---|---|---|---|---|---|
| 1 | 08/09/22 | Recognize face mask and social distancing | System should recognize the face mask and social distancing | Users distance and face mask should be recognize by the system | Yes |
| 1 | 13/09/22 | Response back to user | System should response back to user | System response back to user | Yes |

Table 4.12: Test Sprint 1

Sprint 2

| Test # | Date | Action | Expected Result | Actual Result | Pass? <Yes/ N o> |
|---|---|---|---|---|---|
| 2 | 29/09/22 | System checks face mask and social distancing | System should check face mask and social distancing | System can successfully detect face mask and social distancing | Yes |
| 2 | 02/10/22 | Give the required output | System should give the required output | System should give the required output | Yes |

Table 4.13: Test Sprint 2

## 4.10 Git

Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. To show the continuous development of the project the Gitlab histories is shown in Appendix
(Fig A.13).

# Chapter 5

# System Design and Implementation

## 5.1 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.

The state-of-the-art approach of this paper is capable of deploying pre-trained models. Pre-trained models can achieve good results to detect 2D object even with the different image qualities, angles, and camera models. Hence, this paper uses state-of-the-art pre-trained deep learning models that are used in detecting pedestrians and their face masks. These models can be used in the monitoring of social distancing as explained in the following sections. A Social Distancing Detection and a Face Mask Detection model is trained and deployed to identify whether people are violating the social distancing measures and wearing a mask or not. YOLOv3 along with DBSCAN clustering is used for recognizing the potential intruders. The face mask classification model uses the ResNet50 to train and deploy the model in order to identify the people whether they are with or without a mask. The project description is given below.

## 5.2. PEDESTRIAN DETECTION

You Only Look Once Version 38 is a real-time, state-of-the-art object detection system which is pre-trained on the COCO9 dataset. It has a resolution of 416*416 and is used in this research for the purpose of obtaining bounding boxes of individuals in the videos.

There are many ways for gaining the position of an individual such as obtaining the centre of the bounding box, that is, the midpoint of the bounding box. Other options include OpenCV’s bird’s eye perspective. This project focuses on the first method to obtain the distances between every individual. First, the model detects individuals in the frames and their faces. It then puts a red or green bounding box around the individual and his/her face to determine if they are safe or not and wearing a mask or not. In order to reduce the complexity of the project, the regions that do not contain the object can be discarded. The process of extraction of such regions is called Region Proposals. These algorithms are proposed so that the Regions of Interest (ROIs) can be selected. Therefore, one of the best approaches to ROI can be found by deploying the Region-based Convolutional Neural Network (R-CNN) (Liao et al., 2019).

## 5.3. 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 (Li et al., 2019). Conventional face detectors such as the MTCNN (Xiang and Zhu, 2017) or the HaarCascades (Tej Chinimilli et al., 2020) 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 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. Figure 2 depicts the basic pipeline behind the working of the Social Distancing and the Face Mask Classification model. 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 YOLOv3. If it reaches to EOF, STOP.
4. Further, get the positions of the people with the help of DBSCAN to detect the clusters are forming.
5. While detecting persons, detect their faces with the help of the DSFD model to detect whether they have masks or not.
6. With the help pf bounding boxes om the person and their faces, measure the distance between them and detect masks on them.
7. Create a results board on top pf the video and display the results.
8. Create an output and then show the results.
9. Do this for every frame till it reaches to end of file.

# Chapter 6 Conclusion

## 6.1 Summary

As we envision the world post pandemic the need of self-responsibility emerges irrefutably. The scenario would mostly focus on accepting and obeying the precautions and rules that WHO has imposed more precisely as responsibility of one will totally embark on themselves and not government. Social Distancing and wearing mask would undoubtedly be the most important factor as virus spreads through close contact with infected ones. In order to supervise large mobs, an effective solution is important and this survey paper focuses on that. The article proposes an efficient real-time deep learning based framework to automate the process of monitoring the social distancing via object detection and tracking approaches, where each individual is identified in the real-time with the help of bounding boxes. The generated bounding boxes aid in identifying the clusters or groups of people satisfying the closeness property computed with the help of pairwise vectorized approach. The number of violations are confirmed by computing the number of groups formed and violation index term computed as the ratio of the number of people to the number of groups. The extensive trials were conducted with popular state-of-the-art object detection models. The emerging trends and the availability of intelligent technologies make us to develop new models that help to satisfy the needs of emerging world. So we have developed a novel social distancing detector which can possibly contribute to public healthcare. The model proposes an efficient real-time deep learning based framework to automate the process of monitoring the social distancing via object detection and tracking approaches, where each individual is identified in the real-time with the help of bounding boxes. Identifying the clusters or groups of people satisfying the closeness property computed with the help of Bird’s eye view approach. The number of violations is confirmed by computing the number of groups formed and violation index term computed as the ratio of the number of people to the number of groups. The extensive trials were conducted with popular state-of-the-art object detection models FasterRCNN, SSD, and YOLO v3, since this approach is highly sensitive to the spatial location of the camera, the same approach can be fine-tuned to better adjust with the corresponding field of view. This work also reviewed the pandemic outbreak. Then, it clarified the basic concepts of deep CNN models. After that, this paper reproduced the training and testing of the most used deep pre-trained-based CNN models on the face mask dataset. Finally and after evaluated the numerical results, best models are tested on an webcam where efficient real-time deep learning-based techniques are implemented with a social distancing task to automate the process of detecting masked faces and violated or maintained distance between peoples. This system works very effectively and efficiently in detection of face mask and identifying the social distancing between the people and generating the alert that can be handled and monitored.

## 6.2 SCOPE FOR FUTUREWORK

- Can use a buzzer to alert people
- Android app can also be developed.
- Make social distancing based on real-time video

# Bibliography

[1] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, “Behavior recognition ´ via sparse spatio- temporal features,” in 2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. IEEE, 2005, pp. 65–72.

[2] M. Piccardi,“Background subtraction techniques: a review,” in 2004 IEEE International Conferenceon Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583), vol. 4. IEEE, 2004, pp. 3099– 3104.

[3] Y. Xu, J. Dong, B. Zhang, and D. Xu, “Background modeling methods in video analysis: A review and comparative evaluation,” CAAI Transactions on Intelligence Technology, vol. 1, no. 1, pp. 43– 60, 2016

[4] H. Tsutsui, J. Miura, and Y. Shirai, “Optical flowbased person tracking by multiple cameras,” in Conference Documentation International Conference on Multisensor Fusion and Integration for Intelligent Systems. MFI 2001 (Cat. No. 01TH8590). IEEE, 2001, pp. 91–96.

[5] A. Agarwal, S. Gupta, and D. K. Singh, “Review of optical flow technique for moving object detection,” in 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I). IEEE, 2016, pp. 409–413.

[6] S. A. Niyogi and E. H. Adelson, “Analyzing gait with spatiotemporal surfaces,” in Proceedings of 1994 IEEE Workshop on Motion of Nonrigid and Articulated Objects. IEEE, 1994, pp. 64–69.

[7] WHO, “Who director-generals opening remarks at the media briefing on covid-19-11 march 2020.”https://www.who.int/dg/speeches/detail/, 2020, [Online; accessed March 12, 2020].

# Appendix

## Source Code

Main.py

```
# -*- coding: utf-8 -*-
import tkinter as tk
from tkinter import Message ,Text
from tkinter import *
import tkinter.messagebox
import PIL
from PIL import ImageTk
from PIL import Image
window = tk.Tk()
window.title("Persons Social Distancing For Safety Violation And Face Mask Detection")
dialog_title = 'QUIT'
dialog_text = 'Are you sure?'
#window.geometry('1280x720')
#window.configure(background='#3b5999')
bg= ImageTk.PhotoImage(file="./bg.jpg")
#Create a canvas
canvas= Canvas(window,width= 400, height= 200)
canvas.pack(expand=True, fill= BOTH)
#Add the image in the canvas
canvas.create_image(0,0,image=bg, anchor="nw")
window.wm_attributes("-transparentcolor", 'grey')
window.attributes('-fullscreen', True)
window.grid_rowconfigure(0, weight=1)
window.grid_columnconfigure(0, weight=1)
message = tk.Label(window, text="Persons Social Distancing For Safety Violation And Face
Mask Detection" ,bg="#3b5999" ,fg="white" ,width=58 ,height=3,font=('times', 30, 'italic
bold
underline'))
message.place(x=10, y=15)
#readdatlbl = tk.Label(window, text="Welcome To Social Distancing Safety Violation And
Face
Mask" ,bg="#0084ff" ,fg="white" ,width=55 ,height=2,font=('times', 15, ' bold '))
#readdatlbl.place(x=500, y=300)
def fm():
#window.destroy()
Importos
```

```
os.system('python facemask.py')
def sd():
#window.destroy()
import os
os.system('python social_distance.py')
socialdistance = tk.Button(window, text="Social Distancing", command=sd ,fg="white"
,bg="#0084ff" ,width=20 ,height=2, activebackground = "#21759b" ,font=('times', 15, '
bold '))
socialdistance.place(x=300, y=350)
facemask = tk.Button(window, text="Face Mask", command=fm ,fg="white"
,bg="#0084ff" ,width=20 ,height=2, activebackground = "#21759b" ,font=('times', 15, '
bold '))
facemask.place(x=800, y=350)
quitWindow = tk.Button(window, text="Quit", command=window.destroy
,fg="white" ,bg="#0084ff" ,width=20 ,height=2, activebackground = "#21759b"
,font=('times', 15, ' bold '))
quitWindow.place(x=1100, y=650)
window.mainloop()
import numpy as np
import keras
import keras.backend as k
from keras.layers import Conv2D,MaxPooling2D,SpatialDropout2D,Flatten,Dropout,Dense
from keras.models import Sequential,load_model
from keras.optimizers import adam
from keras.preprocessing import image
import cv2
import datetime
mymodel=load_model('mymodel.h5')
cap=cv2.VideoCapture(0)
face_cascade=cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
```

```
while cap.isOpened():
_,img=cap.read()
face=face_cascade.detectMultiScale(img,scaleFactor=1.1,minNeighbors=4)
for(x,y,w,h) in face:
face_img = img[y:y+h, x:x+w]
cv2.imwrite('temp.jpg',face_img)
test_image=image.load_img('temp.jpg',target_size=(150,150,3))
est_image=image.img_to_array(test_image)
test_image=np.expand_dims(test_image,axis=0)
pred=mymodel.predict_classes(test_image)[0][0]
if pred==1:
cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255),3)
cv2.putText(img,'NO
MASK',((x+w)//2,y+h+20),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,255),3)
else:
cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),3)
cv2.putText(img,'MASK',((x+w)//2,y+h+20),cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),
3)
datet=str(datetime.datetime.now())
cv2.putText(img,datet,(400,450),cv2.FONT_HERSHEY_SIMPLEX,0.5,(255,255,255),1)
cv2.imshow('img',img)
if cv2.waitKey(1)==ord('q'):
break
cap.release()
cv2.destroyAllWindows()
import cv2
import numpy as np
from math import pow, sqrt
#Constant Values
preprocessing = False
calculateConstant_x = 300
calculateConstant_y = 615
```

```
personLabelID = 15.00
debug = True
accuracyThreshold = 0.4
RED = (0,0,255)
YELLOW = (0,255,255)
GREEN = (0,255,0)
write_video = False
# I used CLAHE preprocessing algorithm for detect humans better.
# HSV (Hue, Saturation, and Value channel). CLAHE uses value channel.
# Value channel refers to the lightness or darkness of a colour. An image without hue or
saturation is a grayscale image.
def CLAHE(bgr_image: np.array) -> np.array:
hsv = cv2.cvtColor(bgr_image, cv2.COLOR_BGR2HSV)
hsv_planes = cv2.split(hsv)
clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))
hsv_planes[2] = clahe.apply(hsv_planes[2])
hsv = cv2.merge(hsv_planes)
return cv2.cvtColor(hsv, cv2.COLOR_HSV2BGR)
def centroid(startX,endX,startY,endY):
centroid_x = round((startX+endX)/2,4)
centroid_y = round((startY+endY)/2,4)
bboxHeight = round(endY-startY,4)
return centroid_x,centroid_y,bboxHeight
def calcDistance(bboxHeight):
distance = (calculateConstant_x * calculateConstant_y) / bboxHeight
return distance
def drawResult(frame,position):
for i in position.keys():
if i in highRisk:
rectangleColor = RED
elif i in mediumRisk:
rectangleColor = YELLOW
```

```
else:
rectangleColor = GREEN
(startX, startY, endX, endY) = detectionCoordinates[i]
cv2.rectangle(frame, (startX, startY), (endX, endY), rectangleColor, 2)
if __name__== "__main__":
caffeNetwork =
cv2.dnn.readNetFromCaffe("./SSD_MobileNet_prototxt.txt",
"./SSD_MobileNet.caffemodel")
cap = cv2.VideoCapture("./pedestrians.mp4")
fourcc = cv2.VideoWriter_fourcc(*"XVID")
output_movie = cv2.VideoWriter("./result.avi",
fourcc, 24,
(int(cap.get(cv2.CAP_PROP_FRAME_WIDTH)),
int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))))
while cap.isOpened():
debug_frame, frame = cap.read()
highRisk = set()
mediumRisk = set()
position = dict()
detectionCoordinates = dict()
if not debug_frame:
print("Video cannot opened or finished!")
break
if preprocessing:
frame = CLAHE(frame)
(imageHeight, imageWidth) = frame.shape[:2]
pDetection = cv2.dnn.blobFromImage(cv2.resize(frame, (imageWidth, imageHeight)),
0.007843, (imageWidth, imageHeight), 127.5)
caffeNetwork.setInput(pDetection)
detections = caffeNetwork.forward()
for i in range(detections.shape[2]):
accuracy = detections[0, 0, i, 2]
```

```
if accuracy > accuracyThreshold:
# Detection class and detection box coordinates.
idOfClasses = int(detections[0, 0, i, 1])
box = detections[0, 0, i, 3:7] * np.array([imageWidth, imageHeight, imageWidth,
imageHeight])
(startX, startY, endX, endY) = box.astype('int')
if idOfClasses == personLabelID:
#Default drawing bounding box.
bboxDefaultColor = (255,255,255)
cv2.rectangle(frame, (startX, startY), (endX, endY), bboxDefaultColor, 2)
detectionCoordinates[i] = (startX, startY, endX, endY)
# Centroid of bounding boxes
centroid_x, centroid_y, bboxHeight = centroid(startX,endX,startY,endY)
distance = calcDistance(bboxHeight)
# Centroid in centimeter distance
centroid_x_centimeters = (centroid_x * distance) / calculateConstant_y
centroid_y_centimeters = (centroid_y * distance) / calculateConstant_y
position[i] = (centroid_x_centimeters, centroid_y_centimeters, distance)
#Risk Counter Using Distance of Positions
for i in position.keys():
for j in position.keys():
if i < j:
distanceOfBboxes = sqrt(pow(position[i][0]-position[j][0],2) + pow(position[i][1]-
position[j][1],2) +
pow(position[i][2]-position[j][2],2))
if distanceOfBboxes < 150: # 150cm or lower
highRisk.add(i),highRisk.add(j)
elif distanceOfBboxes < 200 > 150: # between 150 and 200
mediumRisk.add(i),mediumRisk.add(j)
cv2.putText(frame, "Person in High Risk : " + str(len(highRisk)) , (20,
20),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 255), 2)
```

```
cv2.putText(frame, "Person in Medium Risk : " + str(len(mediumRisk)) , (20, 40),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 255), 2)
cv2.putText(frame, "Detected Person : " +
str(len(detectionCoordinates)), (20, 60),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)
drawResult(frame, position)
if write_video:
output_movie.write(frame)
cv2.imshow('Result', frame)
waitkey = cv2.waitKey(1)
if waitkey == ord("q"):
break
cap.release()
cv2.destroyAllWindows()
```

```
<?xml version="1.0"?>
<!--
Stump-based 24x24 discrete(?) adaboost frontal face detector.
Created by Rainer Lienhart.
-->
<opencv_storage>
<cascade type_id="opencv-cascade-classifier"><stageType>BOOST</stageType>
<featureType>HAAR</featureType>
<height>24</height>
<width>24</width>
<stageParams>
<maxWeakCount>211</maxWeakCount></stageParams>
<featureParams>
<maxCatCount>0</maxCatCount></featureParams>
<stageNum>25</stageNum>
<stages>
<_>
<maxWeakCount>9</maxWeakCount>
```

```
<stageThreshold>-5.0425500869750977e+00</stageThreshold>
<weakClassifiers>
<_>
<internalNodes>
0 -1 0 -3.1511999666690826e-02</internalNodes>
<leafValues>
2.0875380039215088e+00 -2.2172100543975830e+00</leafValues></_>
<_>
<internalNodes>
0 -1 1 1.2396000325679779e-02</internalNodes>
<leafValues>
-1.8633940219879150e+00 1.3272049427032471e+00</leafValues></_>
<_>
<internalNodes>
0 -1 2 2.1927999332547188e-02</internalNodes>
<leafValues>
-1.5105249881744385e+00 1.0625729560852051e+00</leafValues></_>
<_>
<internalNodes>
0 -1 3 5.7529998011887074e-03</internalNodes>
<leafValues>
-8.7463897466659546e-01 1.1760339736938477e+00</leafValues></_>
<_>
<internalNodes>
0 -1 4 1.5014000236988068e-02</internalNodes>
<leafValues>
-7.7945697307586670e-01 1.2608419656753540e+00</leafValues></_>
<_>
<internalNodes>
0 -1 5 9.9371001124382019e-02</internalNodes>
<leafValues>
5.5751299858093262e-01 -1.8743000030517578e+00</leafValues></_>
```

```
<_>
<internalNodes>
0 -1 6 2.7340000960975885e-03</internalNodes>
<leafValues>
-1.6911929845809937e+00 4.4009700417518616e-01</leafValues></_>
<_>
<internalNodes>
0 -1 7 -1.8859000876545906e-02</internalNodes>
<leafValues>
-1.4769539833068848e+00 4.4350099563598633e-01</leafValues></_>
<_>
<internalNodes>
0 -1 8 5.9739998541772366e-03</internalNodes>
<leafValues>
-8.5909199714660645e-01                                  8.5255599021911621e-
01</leafValues></_></weakClassifiers></_>
<_>
<maxWeakCount>16</maxWeakCount>
<stageThreshold>-4.9842400550842285e+00</stageThreshold>
<weakClassifiers>
<_>
<internalNodes>
0 -1 9 -2.1110000088810921e-02</internalNodes>
<leafValues>
1.2435649633407593e+00 -1.5713009834289551e+00</leafValues></_>
<_>
<internalNodes>
0 -1 10 2.0355999469757080e-02</internalNodes>
<leafValues>
-1.6204780340194702e+00 1.1817760467529297e+00</leafValues></_>
<_>
<internalNodes>
```

```
0 -1 11 2.1308999508619308e-02</internalNodes>
<leafValues>
-1.9415930509567261e+00 7.0069098472595215e-01</leafValues></_>
<_>

<internalNodes>
0 -1 12 9.1660000383853912e-02</internalNodes>
<leafValues>
-5.5670100450515747e-01 1.7284419536590576e+00</leafValues></_>
<_>
<internalNodes>
0 -1 13 3.6288000643253326e-02</internalNodes>
<leafValues>
2.6763799786567688e-01 -2.1831810474395752e+00</leafValues></_>
<_>
<internalNodes>
0 -1 14 -1.9109999760985374e-02</internalNodes>
<leafValues>
-2.6730210781097412e+00 4.5670801401138306e-01</leafValues></_>
<_>
<internalNodes>
0 -1 15 8.2539999857544899e-03</internalNodes>
<leafValues>
-1.0852910280227661e+00 5.3564202785491943e-01</leafValues></_>
<_>
<internalNodes>
0 -1 16 1.8355000764131546e-02</internalNodes>
<leafValues>
-3.5200199484825134e-01 9.3339198827743530e-01</leafValues></_>
<_>
<internalNodes>
0 -1 17 -7.0569999516010284e-03</internalNodes>
```

```
<leafValues>
9.2782098054885864e-01 -6.6349899768829346e-01</leafValues></_>
<_>
<internalNodes>
0 -1 18 -9.8770000040531158e-03</internalNodes>
<leafValues>
1.1577470302581787e+00 -2.9774799942970276e-01</leafValues></_>
<_>
<internalNodes>
0 -1 19 1.5814000740647316e-02</internalNodes>
<leafValues>
-4.1960600018501282e-01 1.3576040267944336e+00</leafValues></_>
<_>
<internalNodes>
0 -1 20 -2.0700000226497650e-02</internalNodes>
<leafValues>
1.4590020179748535e+00 -1.9739399850368500e-01</leafValues></_>
<_>
<internalNodes>
0 -1 21 -1.3760800659656525e-01</internalNodes>
<leafValues>
1.1186759471893311e+00 -5.2915501594543457e-01</leafValues></_>
<_>
<internalNodes>
0 -1 22 1.4318999834358692e-02</internalNodes>
<leafValues>
-3.5127198696136475e-01 1.1440860033035278e+00</leafValues></_>
<_>
<internalNodes>
0 -1 23 1.0253000073134899e-02</internalNodes>
<leafValues>
-6.0850602388381958e-01 7.7098500728607178e-01</leafValues></_>
```

```
<_>
<internalNodes>
0 -1 24 9.1508001089096069e-02</internalNodes>
<leafValues>
3.8817799091339111e-01                                                          -
1.5122940540313721e+00</leafValues></_></weakClassifiers></_>
<_>
<maxWeakCount>27</maxWeakCount>
<stageThreshold>-4.6551899909973145e+00</stageThreshold>
<weakClassifiers>
<_>
<internalNodes>
0 -1 25 6.9747000932693481e-02</internalNodes>
<leafValues>
-1.0130879878997803e+00 1.4687349796295166e+00</leafValues></_>
<_>
<internalNodes>
0 -1 26 3.1502999365329742e-02</internalNodes>
<leafValues>
-1.6463639736175537e+00 1.0000629425048828e+00</leafValues></_>
<_>
<internalNodes>
0 -1 27 1.4260999858379364e-02</internalNodes>
<leafValues>
4.6480301022529602e-01 -1.5959889888763428e+00</leafValues></_>
<_>
<internalNodes>
0 -1 28 1.4453000389039516e-02</internalNodes>
<leafValues>
-6.5511900186538696e-01 8.3021801710128784e-01</leafValues></_>
<_>
<internalNodes>
```

```
0 -1 29 -3.0509999487549067e-03</internalNodes>
<leafValues>
-1.3982310295104980e+00 4.2550599575042725e-01</leafValues></_>
<_>
<internalNodes>
0 -1 30 3.2722998410463333e-02</internalNodes>
<leafValues>
-5.0702601671218872e-01 1.0526109933853149e+00</leafValues></_>
<_>
<internalNodes>
0 -1 31 -7.2960001416504383e-03</internalNodes>
<leafValues>
3.6356899142265320e-01 -1.3464889526367188e+00</leafValues></_>
<_>
<internalNodes>
0 -1 32 5.0425000488758087e-02</internalNodes>
<leafValues>
-3.0461400747299194e-01 1.4504129886627197e+00</leafValues></_>
<_>
<internalNodes>
0 -1 33 4.6879000961780548e-02</internalNodes>
<leafValues>
-4.0286201238632202e-01 1.2145609855651855e+00</leafValues></_>
<_>
<internalNodes>
0 -1 34 -6.9358997046947479e-02</internalNodes>
<leafValues>
1.0539360046386719e+00 -4.5719701051712036e-01</leafValues></_>
<_>
<internalNodes>
0 -1 35 -4.9033999443054199e-02</internalNodes>
<leafValues>
```

```
-1.6253089904785156e+00 1.5378999710083008e-01</leafValues></_>
<_>
<internalNodes>
0 -1 36 8.4827996790409088e-02</internalNodes>
<leafValues>
2.8402999043464661e-01 -1.5662059783935547e+00</leafValues></_>
<_>
<internalNodes>
0 -1 37 -1.7229999648407102e-03</internalNodes>
<leafValues>
-1.0147459506988525e+00 2.3294800519943237e-01</leafValues></_>
<_>
<internalNodes>
0 -1 38 1.1562199890613556e-01</internalNodes>
<leafValues>
-1.6732899844646454e-01 1.2804069519042969e+00</leafValues></_>
<_>
<internalNodes>
0 -1 39 -5.1279999315738678e-02</internalNodes>
<leafValues>
1.5162390470504761e+00 -3.0271100997924805e-01</leafValues></_>
<_>
<internalNodes>
0 -1 40 -4.2706999927759171e-02</internalNodes>
<leafValues>

1.7631920576095581e+00 -5.1832001656293869e-02</leafValues></_>
<_>
<internalNodes>
0 -1 41 3.7178099155426025e-01</internalNodes>
<leafValues>
-3.1389200687408447e-01 1.5357979536056519e+00</leafValues></_>
```

```
<_>
<internalNodes>
0 -1 42 1.9412999972701073e-02</internalNodes>
<leafValues>
-1.0017599910497665e-01 9.3655401468276978e-01</leafValues></_>
<_>
<internalNodes>
0 -1 43 1.7439000308513641e-02</internalNodes>
<leafValues>
-4.0379899740219116e-01 9.6293002367019653e-01</leafValues></_>
<_>
<internalNodes>
0 -1 44 3.9638999849557877e-02</internalNodes>
<leafValues>
1.7039099335670471e-01 -2.9602990150451660e+00</leafValues></_>
<_>
<internalNodes>
0 -1 45 -9.1469995677471161e-03</internalNodes>
<leafValues>
8.8786798715591431e-01 -4.3818700313568115e-01</leafValues></_>
<_>
<internalNodes>
0 -1 46 1.7219999572262168e-03</internalNodes>
<leafValues>
-3.7218600511550903e-01 4.0018901228904724e-01</leafValues></_>
<_>
<internalNodes>
0 -1 47 3.0231000855565071e-02</internalNodes>
<leafValues>
6.5924003720283508e-02 -2.6469180583953857e+00</leafValues></_>
<_>
```

```
<internalNodes>
0 -1 48 -7.8795999288558960e-02</internalNodes>
<leafValues>
-1.7491459846496582e+00 2.8475299477577209e-01</leafValues></_>
<_>
<internalNodes>
0 -1 49 2.1110000088810921e-03</internalNodes>
<leafValues>
-9.3908101320266724e-01 2.3205199837684631e-01</leafValues></_>
<_>
<internalNodes>
0 -1 50 2.7091000229120255e-02</internalNodes>
<leafValues>
-5.2664000540971756e-02 1.0756820440292358e+00</leafValues></_>
<_>
<internalNodes>
0 -1 51 -4.4964998960494995e-02</internalNodes>
<leafValues>
-1.8294479846954346e+00                                  9.9561996757984161e-
02</leafValues></_></weakClassifiers></_>
<_>
<maxWeakCount>32</maxWeakCount>
<stageThreshold>-4.4531588554382324e+00</stageThreshold>
<weakClassifiers>
<_>
<internalNodes>
0 -1 52 -6.5701000392436981e-02</internalNodes>
<leafValues>
1.1558510065078735e+00 -1.0716359615325928e+00</leafValues></_>
<_>
<internalNodes>
0 -1 53 1.5839999541640282e-02</internalNodes>
```

```
<leafValues>
-1.5634720325469971e+00 7.6877099275588989e-01</leafValues></_>
<_>
<internalNodes>
0 -1 54 1.4570899307727814e-01</internalNodes>
<leafValues>

-5.7450097799301147e-01 1.3808720111846924e+00</leafValues></_>
<_>
<internalNodes>
0 -1 55 6.1389999464154243e-03</internalNodes>
<leafValues>
-1.4570560455322266e+00 5.1610302925109863e-01</leafValues></_>
<_>
<internalNodes>
0 -1 56 6.7179999314248562e-03</internalNodes>
<leafValues>
-8.3533602952957153e-01 5.8522200584411621e-01</leafValues></_>
<_>
<internalNodes>
0 -1 57 1.8518000841140747e-02</internalNodes>
<leafValues>
-3.1312099099159241e-01 1.1696679592132568e+00</leafValues></_>
<_>
<internalNodes>
0 -1 58 1.9958000630140305e-02</internalNodes>
<leafValues>
-4.3442600965499878e-01 9.5446902513504028e-01</leafValues></_>
<_>
<internalNodes>
0 -1 59 -2.7755001187324524e-01</internalNodes>
<leafValues>
```

1.4906179904937744e+00 -1.3815900683403015e-01</leafValues></_>
<_>
<internalNodes>
0 -1 60 9.1859996318817139e-03</internalNodes>
<leafValues>
-9.6361500024795532e-01 2.7665498852729797e-01</leafValues></_>
<_>
<internalNodes>
0 -1 61 -3.7737999111413956e-02</internalNodes>
<leafValues>
-2.4464108943939209e+00 2.3619599640369415e-01</leafValues></_>
<_>
<internalNodes>
0 -1 62 1.8463000655174255e-02</internalNodes>
<leafValues>
1.7539200186729431e-01 -1.3423130512237549e+00</leafValues></_>
<_>
<internalNodes>
0 -1 63 -1.1114999651908875e-02</internalNodes>
<leafValues>
4.8710799217224121e-01 -8.9851897954940796e-01</leafValues></_>
<_>
<internalNodes>
0 -1 64 3.3927999436855316e-02</internalNodes>
<leafValues>
1.7874200642108917e-01 -1.6342279911041260e+00</leafValues></_>
<_>
<internalNodes>
0 -1 65 -3.5649001598358154e-02</internalNodes>
<leafValues>
-1.9607399702072144e+00 1.8102499842643738e-01</leafValues></_>
<_>

```
<internalNodes>
0 -1 66 -1.1438000015914440e-02</internalNodes>
<leafValues>
9.9010699987411499e-01 -3.8103199005126953e-01</leafValues></_>
<_>
<internalNodes>
0 -1 67 -6.5236002206802368e-02</internalNodes>
<leafValues>
-2.5794160366058350e+00 2.4753600358963013e-01</leafValues></_>
<_>
<internalNodes>
0 -1 68 -4.2272001504898071e-02</internalNodes>
<leafValues>
1.4411840438842773e+00 -2.9508298635482788e-01</leafValues></_>
<_>
<internalNodes>
0 -1 69 1.9219999667257071e-03</internalNodes>
<leafValues>
-4.9608600139617920e-01 6.3173598051071167e-01</leafValues></_>
<_>
<internalNodes>
0 -1 70 -1.2921799719333649e-01</internalNodes>
<leafValues>
-2.3314270973205566e+00 5.4496999830007553e-02</leafValues></_>
<_>
<internalNodes>
0 -1 71 2.2931000217795372e-02</internalNodes>
<leafValues>
-8.4447097778320312e-01 3.8738098740577698e-01</leafValues></_>
<_>
<internalNodes>
0 -1 72 -3.4120000898838043e-02</internalNodes>
```

```
<leafValues>
-1.4431500434875488e+00 9.8422996699810028e-02</leafValues></_>
<_>
<internalNodes>
0 -1 73 2.6223000138998032e-02</internalNodes>
<leafValues>
1.8223099410533905e-01 -1.2586519718170166e+00</leafValues></_>
<_>
<internalNodes>
0 -1 74 2.2236999124288559e-02</internalNodes>
<leafValues>
6.9807998836040497e-02 -2.3820950984954834e+00</leafValues></_>
<_>
<internalNodes>
0 -1 75 -5.8240001089870930e-03</internalNodes>
<leafValues>
3.9332500100135803e-01 -2.7542799711227417e-01</leafValues></_>
<_>
<internalNodes>
0 -1 76 4.3653000146150589e-02</internalNodes>
<leafValues>
1.4832699298858643e-01 -1.1368780136108398e+00</leafValues></_>

<_>
<internalNodes>
0 -1 77 5.7266999036073685e-02</internalNodes>
<leafValues>
2.4628099799156189e-01 -1.2687400579452515e+00</leafValues></_>
<_>
<internalNodes>
0 -1 78 2.3409998975694180e-03</internalNodes>
<leafValues>
```

```
-7.5448900461196899e-01 2.7163800597190857e-01</leafValues></_>
<_>
<internalNodes>
0 -1 79 1.2996000237762928e-02</internalNodes>
<leafValues>
-3.6394900083541870e-01 7.0959198474884033e-01</leafValues></_>
<_>
<internalNodes>
0 -1 80 -2.6517000049352646e-02</internalNodes>
<leafValues>
-2.3221859931945801e+00 3.5744000226259232e-02</leafValues></_>
<_>
<internalNodes>
0 -1 81 -5.8400002308189869e-03</internalNodes>
<leafValues>
4.2194300889968872e-01 -4.8184998333454132e-02</leafValues></_>
<_>
<internalNodes>
0 -1 82 -1.6568999737501144e-02</internalNodes>
<leafValues>
1.1099940538406372e+00 -3.4849700331687927e-01</leafValues></_>
<_>
<internalNodes>
0 -1 83 -6.8157002329826355e-02</internalNodes>
<leafValues>
-3.3269989490509033e+00 2.1299000084400177e-01</leafValues></_></weakClassifiers></_>
<_>
<maxWeakCount>52</maxWeakCount>

<stageThreshold>-4.3864588737487793e+00</stageThreshold>
<weakClassifiers>
```

```
<_>
<internalNodes>
0 -1 84 3.9974000304937363e-02</internalNodes>
<leafValues>
-1.2173449993133545e+00 1.0826710462570190e+00</leafValues></_>
<_>
<internalNodes>
0 -1 85 1.8819500505924225e-01</internalNodes>
<leafValues>
-4.8289400339126587e-01 1.4045250415802002e+00</leafValues></_>
<_>
<internalNodes>
0 -1 86 7.8027002513408661e-02</internalNodes>
<leafValues>
-1.0782150030136108e+00 7.4040299654006958e-01</leafValues></_>
<_>
<internalNodes>
0 -1 87 1.1899999663000926e-04</internalNodes>
<leafValues>
-1.2019979953765869e+00 3.7749201059341431e-01</leafValues></_>
<_>
<internalNodes>
0 -1 88 8.5056997835636139e-02</internalNodes>
<leafValues>
-4.3939098715782166e-01 1.2647340297698975e+00</leafValues></_>
<_>
<internalNodes>
0 -1 89 8.9720003306865692e-03</internalNodes>
<leafValues>
-1.8440499901771545e-01 4.5726400613784790e-01</leafValues></_>
<_>
<internalNodes>
```

```
0 -1 90 8.8120000436902046e-03</internalNodes>
<leafValues>
3.0396699905395508e-01-9.5991098880767822e-01</leafValues></_>

<_>
<internalNodes>
0 -1 91 -2.3507999256253242e-02</internalNodes>
<leafValues>
1.2487529516220093e+00 4.6227999031543732e-02</leafValues></_>
<_>
<internalNodes>
0 -1 92 7.0039997808635235e-03</internalNodes>
<leafValues>
-5.9442102909088135e-01 5.3963297605514526e-01</leafValues></_>
<_>
<internalNodes>
0 -1 93 3.3851999789476395e-02</internalNodes>
<leafValues>
2.8496098518371582e-01 -1.4895249605178833e+00</leafValues></_>
<_>
<internalNodes>
0 -1 94 -3.2530000898987055e-03</internalNodes>
<leafValues>
4.8120799660682678e-01 -5.2712398767471313e-01</leafValues></_>
<_>
<internalNodes>
0 -1 95 2.9097000136971474e-02</internalNodes>
<leafValues>
2.6743900775909424e-01 -1.6007850170135498e+00</leafValues></_>
<_>
<internalNodes>
0 -1 96 -8.4790000692009926e-03</internalNodes>
```

```
<leafValues>
-1.3107639551162720e+00 1.5243099629878998e-01</leafValues></_>
<_>
<internalNodes>
0 -1 97 -1.0795000009238720e-02</internalNodes>
<leafValues>
4.5613598823547363e-01 -7.2050899267196655e-01</leafValues></_>
<_>
<internalNodes>

0 -1 98 -2.4620000272989273e-02</internalNodes>
<leafValues>
-1.7320619821548462e+00 6.8363003432750702e-02</leafValues></_>
<_>
<internalNodes>
0 -1 99 3.7380000576376915e-03</internalNodes>
<leafValues>
-1.9303299486637115e-01 6.8243497610092163e-01</leafValues></_>
<_>
<internalNodes>
0 -1 100 -1.2264000251889229e-02</internalNodes>
<leafValues>
-1.6095290184020996e+00 7.5268000364303589e-02</leafValues></_>
<_>
<internalNodes>
0 -1 101 -4.8670000396668911e-03</internalNodes>
<leafValues>
7.4286502599716187e-01 -2.1510200202465057e-01</leafValues></_>
<_>
<internalNodes>
0 -1 102 7.6725997030735016e-02</internalNodes>
<leafValues>
```

```
-2.6835098862648010e-01 1.3094140291213989e+00</leafValues></_>
<_>
<internalNodes>
0 -1 103 2.8578000143170357e-02</internalNodes>
<leafValues>
-5.8793000876903534e-02 1.2196329832077026e+00</leafValues></_>
<_>
<internalNodes>
0 -1 104 1.9694000482559204e-02</internalNodes>
<leafValues>
-3.5142898559570312e-01 8.4926998615264893e-01</leafValues></_>
<_>
<internalNodes>
0 -1 105 -2.9093999415636063e-02</internalNodes>
<leafValues>
-1.0507299900054932e+00 2.9806300997734070e-01</leafValues></_>
<_>
<internalNodes>
0 -1 106 -2.9144000262022018e-02</internalNodes>
<leafValues>
8.2547801733016968e-01 -3.2687199115753174e-01</leafValues></_>
<_>
<internalNodes>
0 -1 107 1.9741000607609749e-02</internalNodes>
<leafValues>
2.0452600717544556e-01 -8.3760201930999756e-01</leafValues></_>
<_>
<internalNodes>
0 -1 108 4.3299999088048935e-03</internalNodes>
<leafValues>
2.0577900111675262e-01 -6.6829800605773926e-01</leafValues></_>
<_>
```

```
<internalNodes>
0 -1 109 -3.5500999540090561e-02</internalNodes>
<leafValues>
-1.2969900369644165e+00 1.3897499442100525e-01</leafValues></_>
<_>
<internalNodes>
0 -1 110 -1.6172999516129494e-02</internalNodes>
<leafValues>
-1.3110569715499878e+00 7.5751997530460358e-02</leafValues></_>
<_>
<internalNodes>
0 -1 111 -2.2151000797748566e-02</internalNodes>
<leafValues>
-1.0524389743804932e+00 1.9241100549697876e-01</leafValues></_>
<_>
<internalNodes>
0 -1 112 -2.2707000374794006e-02</internalNodes>
<leafValues>
-1.3735309839248657e+00 6.6780999302864075e-02</leafValues></_>
<_>
<internalNodes>
0 -1 113 1.6607999801635742e-02</internalNodes>
<leafValues>
-3.7135999649763107e-02 7.7846401929855347e-01</leafValues></_>
<_>
<internalNodes>
0 -1 114 -1.3309000059962273e-02</internalNodes>
<leafValues>
-9.9850702285766602e-01 1.2248100340366364e-01</leafValues></_>
<_>
<internalNodes>
0 -1 115 -3.3732000738382339e-02</internalNodes>
```

```
<leafValues>
1.4461359977722168e+00 1.3151999562978745e-02</leafValues></_>
<_>
<internalNodes>
0 -1 116 1.6935000196099281e-02</internalNodes>
<leafValues>
-3.7121298909187317e-01 5.2842199802398682e-01</leafValues></_>
<_>
<internalNodes>
0 -1 117 3.3259999472647905e-03</internalNodes>
<leafValues>
-5.7568502426147461e-01 3.9261901378631592e-01</leafValues></_>
<_>
<internalNodes>
0 -1 118 8.3644002676010132e-02</internalNodes>
<leafValues>
1.6116000711917877e-02 -2.1173279285430908e+00</leafValues></_>
<_>
<internalNodes>
0 -1 119 2.5785198807716370e-01</internalNodes>
<leafValues>
-8.1609003245830536e-02 9.8782497644424438e-01</leafValues></_>
<_>
<internalNodes>
0 -1 120 -3.6566998809576035e-02</internalNodes>
<leafValues>
-1.1512110233306885e+00 9.6459001302719116e-02</leafValues></_>
<_>
<internalNodes>
0 -1 121 -1.6445999965071678e-02</internalNodes>
<leafValues>
3.7315499782562256e-01 -1.4585399627685547e-01</leafValues></_>
```

```
<_>
<internalNodes>
0 -1 122 -3.7519999314099550e-03</internalNodes>
<leafValues>
2.6179298758506775e-01 -5.8156698942184448e-01</leafValues></_>
<_>
<internalNodes>
0 -1 123 -6.3660000450909138e-03</internalNodes>
<leafValues>
7.5477397441864014e-01 -1.7055200040340424e-01</leafValues></_>
<_>
<internalNodes>
0 -1 124 -3.8499999791383743e-03</internalNodes>
<leafValues>
2.2653999924659729e-01 -6.3876402378082275e-01</leafValues></_>
<_>
<internalNodes>
0 -1 125 -4.5494001358747482e-02</internalNodes>
<leafValues>
-1.2640299797058105e+00 2.5260698795318604e-01</leafValues></_>
<_>
<internalNodes>
0 -1 126 -2.3941000923514366e-02</internalNodes>
<leafValues>
8.7068402767181396e-01 -2.7104699611663818e-01</leafValues></_>
<_>
<internalNodes>
0 -1 127 -7.7558003365993500e-02</internalNodes>
<leafValues>
-1.3901610374450684e+00 2.3612299561500549e-01</leafValues></_>
<_>
<internalNodes>
```

```
0 -1 128 2.3614000529050827e-02</internalNodes>
<leafValues>
6.6140003502368927e-02 -1.2645419836044312e+00</leafValues></_>
<_>
<internalNodes>
0 -1 129 -2.5750000495463610e-03</internalNodes>
<leafValues>
-5.3841698169708252e-01 3.0379098653793335e-01</leafValues></_>
<_>
<internalNodes>
0 -1 130 1.2010800093412399e-01</internalNodes>
<leafValues>
-3.5343000292778015e-01 5.2866202592849731e-01</leafValues></_>
<_>
<internalNodes>
0 -1 131 2.2899999748915434e-03</internalNodes>
<leafValues>
-5.8701997995376587e-01 2.4061000347137451e-01</leafValues></_>
<_>
<internalNodes>
0 -1 132 6.9716997444629669e-02</internalNodes>
<leafValues>
-3.3348900079727173e-01 5.1916301250457764e-01</leafValues></_>
<_>
<internalNodes>
0 -1 133 -4.6670001000165939e-02</internalNodes>
<leafValues>
6.9795399904251099e-01 -1.4895999804139137e-02</leafValues></_>
<_>
<internalNodes>
0 -1 134 -5.0129000097513199e-02</internalNodes>
<leafValues>
```

```
8.6146199703216553e-01 -2.5986000895500183e-01</leafValues></_>
<_>
<internalNodes>
0 -1 135 3.0147999525070190e-02</internalNodes>
<leafValues>
1.9332799315452576e-01 -5.9131097793579102e-
01</leafValues></_></weakClassifiers></_>
<_>
<maxWeakCount>53</maxWeakCount>
<stageThreshold>-4.1299300193786621e+00</stageThreshold>
<weakClassifiers>
<_>
```

# Appendix

## Output

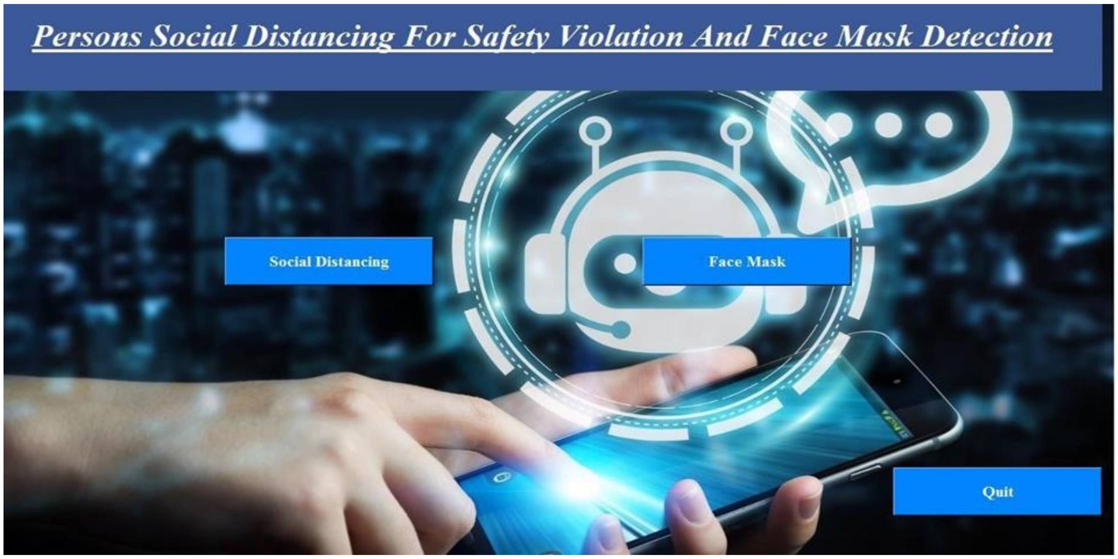
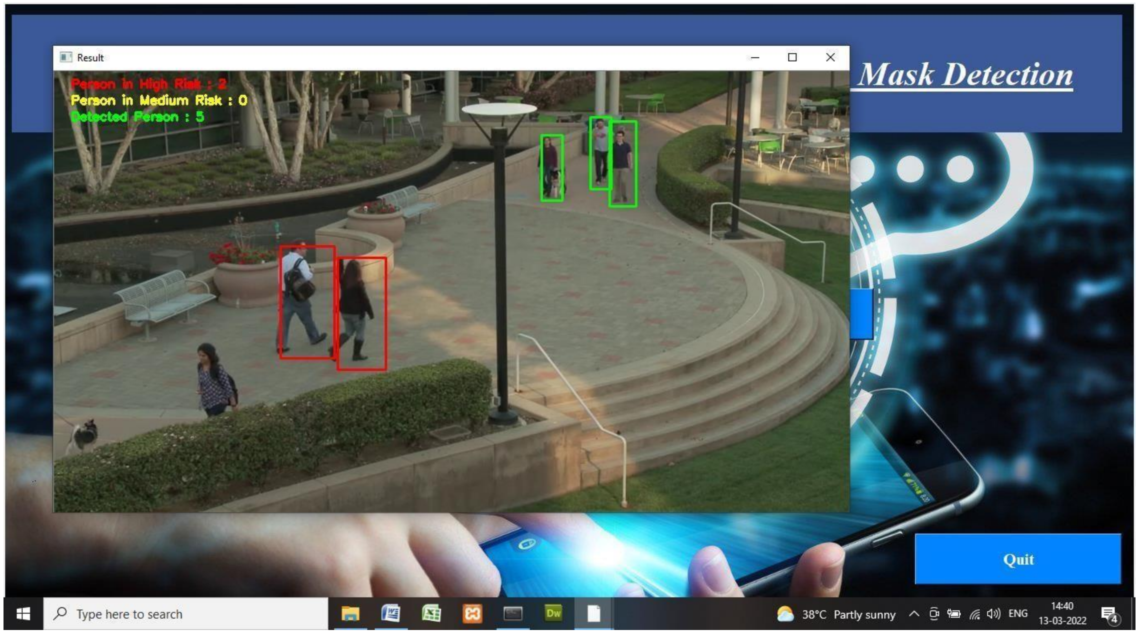

Persons Social Distancing For Safety Violation And Face Mask Detection
Social Distancing
Face Mask
Quit

Fig: 8.3.1 Splash Screen



```
Person in High Risk : 2
Person in Medium Risk : 0
Detected Person : 5
```

Fig 8.3.2 Social distancing

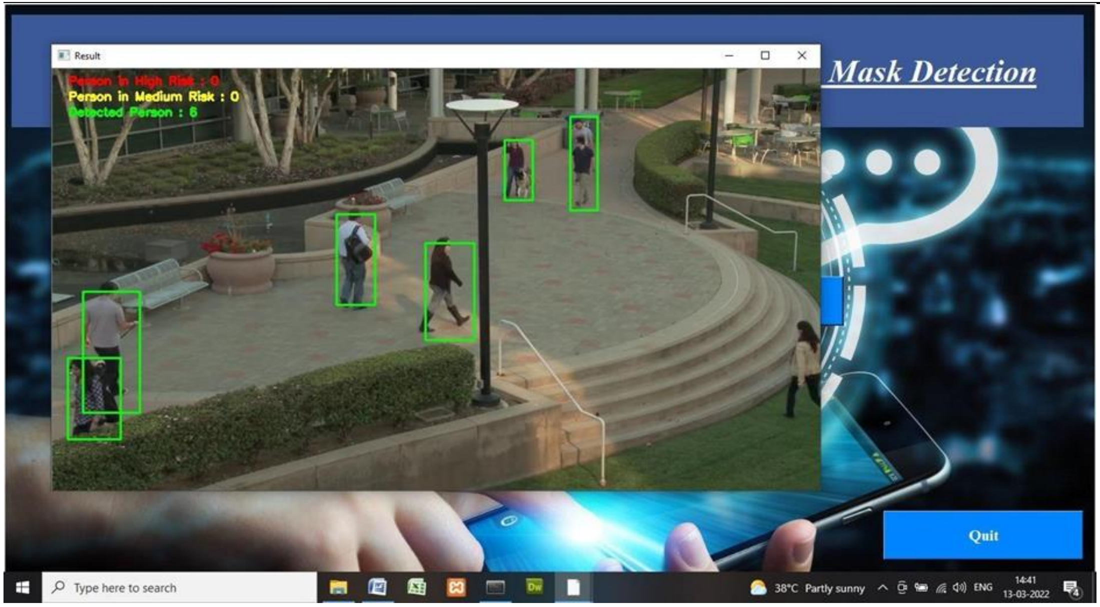


Fig 8.3.3 Social distancing

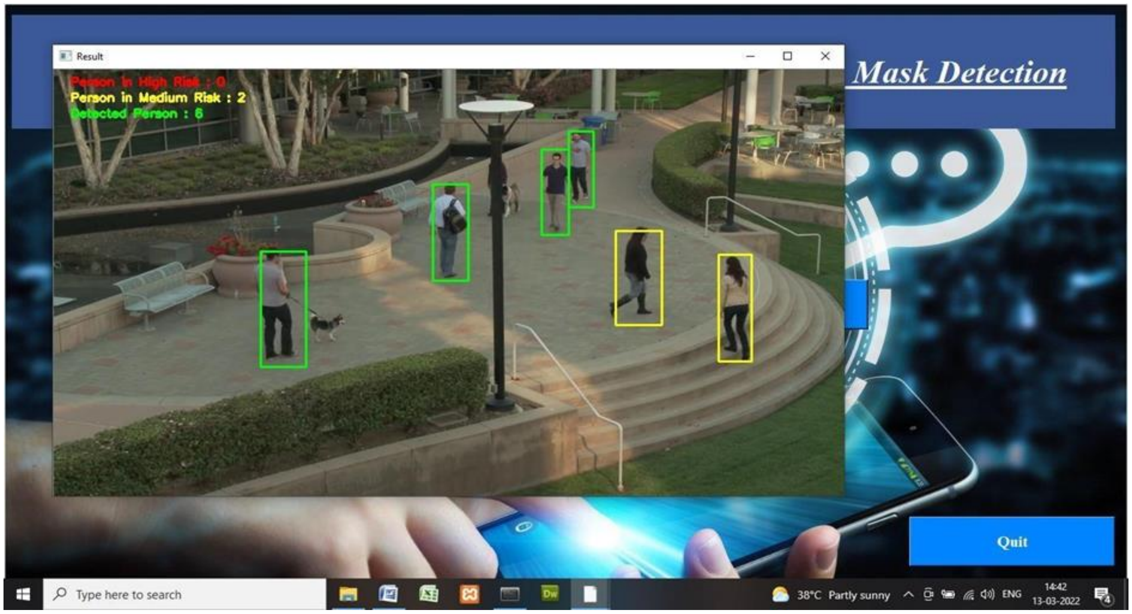


Fig 8.3.4 Social distancing

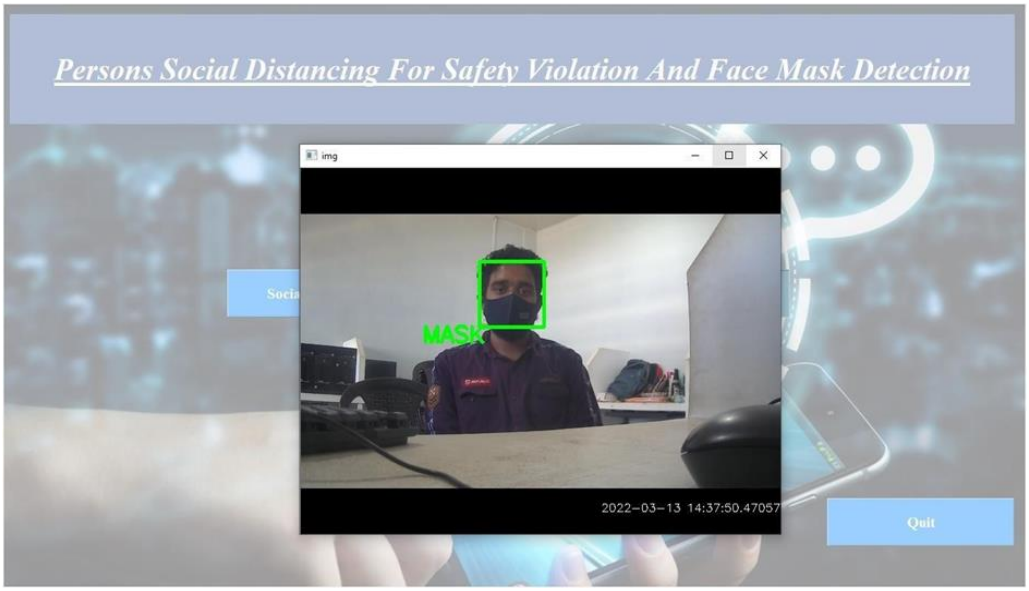

Persons Social Distancing For Safety Violation And Face Mask Detection
img
Socia
MASK
2022-03-13 14:37:50.47057
Quit

Fig 8.3.5 Face with mask

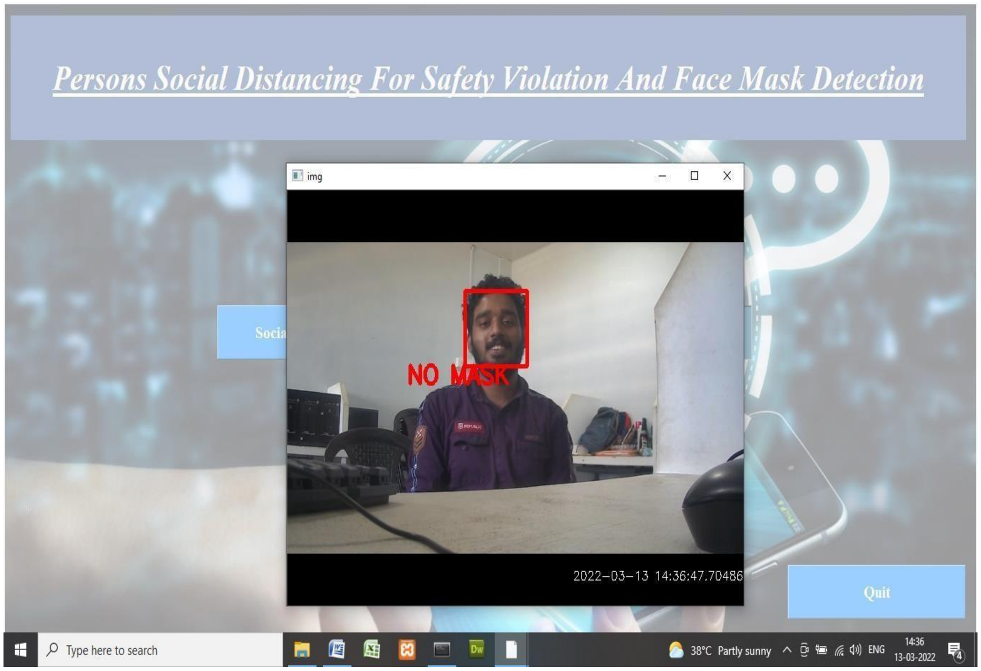


Fig 8.3.6 Face with no mask