**Social Distance Detector**

**(AY20BECSP70520)**

A **Major Project Report** Submitted in partial fulfilment of the requirements

of the degree of

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER ENGINEERING**

BY

Isha Dharmraj Gupta (Roll No. 12 )

Shivang Rajiv Srivastava (Roll No. 55)

Gaurav Dharmendra Upadhyay (Roll No. 58)

Supervisor

Prof. Uma Goradiya

****

**DEPARTMENT OF COMPUTER ENGINEERING**

**SHREE L. R. TIWARI COLLEGE OF ENGINEERING**

**KANAKIA PARK, MIRA ROAD (E), THANE -401 107, MAHARASHTRA.**

**University of Mumbai**

**(AY 2020-21)**

# Declaration by the Candidate

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We 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. We 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.

Date: 27th November, 2020

**(Isha Dharmraj Gupta)**

Roll No.:12 Exam. Seat No.:

**(Shivang Rajiv Srivastava)**

Roll No.:55 Exam. Seat No.:

**(Gaurav Dharmendra Upadhyay)**

Roll No.:58 Exam. Seat No.:

|  |
| --- |
|  |
| DEPARTMENT OF COMPUTER ENGINEERING |
| CSP705 Major Project \_I |
| Seventh Semester, 2020-2021 (Odd Semester) |

CERTIFICATE

This is to certify that the **Major Project** entitled**“SOCIAL DISTANCE DETECTOR”** is a bonafide work of

**Isha Dharmraj Gupta (Roll No. 12)**

**Shivang Rajiv Srivastava (Roll No. 55)**

**Gaurav Dharmendra Upadhyay (Roll No. 58)**

submitted to the University of Mumbai in partial fulfilment of the requirement of course name “**Major Project – I**” having course code **CSP705** for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering”**.

**Signature of Supervisor/Guide**

**Name: Prof. Uma Goradiya**

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of the H.O.D. Signature of the Principal**

**Name: Mrs. Neelam Phadnis Name: Dr. S. Ram Reddy**

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |
| --- |
|  |
| DEPARTMENT OF COMPUTER ENGINEERING |
| CSP705 Major Project \_I |
| Seventh Semester, 2020-2021 (Odd Semester) |

# Major Project Report Approval

This major project report entitled “**SOCIAL DISTANCE DETECTOR*”*** by

**Isha Dharmraj Gupta (Roll No. 12)**

**Shivang Rajiv Srivastava (Roll No. 55)**

**Gaurav Dharmendra Upadhyay (Roll No. 58)**

is belonging to the course name “**Major Project – I”** having course code **CSP705** submitted as a Term work and approved for the degree of Bachelor of Engineering in Computer Engineering.

**Examiners**

1. Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(Internal)

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(External)

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Date:**

**Place:**

# Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

We are deeply thankful to our Guide Prof. Uma Goradiya , Dr. Vinayak D. Shinde for their patient guidance, enthusiastic encouragement and useful critiques of this research work and for helping us throughout the course in accomplishing our final project. Their guidance, support and motivation enabled us in achieving the objectives of the project.Our grateful thanks are also extended to Mr Adrian Rosebrock for his insightful blog about the use of object detection and use of YOLO object detection with Open CV.

We heartily thank our internal project guide Prof. Uma Goradiya Department of Computer Science, for her guidance and suggestions during this project work. We are thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staff of the Department of computer science which helped us in completing our project work. Also, we would like to extend our sincere regard to all the non teaching staff of the department of computer science for their timely support.

We would also like to extend my thanks to the technicians of the laboratory of the computer science department for their help in offering us the resources in running the program. Finally, we wish to thank our parents for their support and encouragement throughout our study.

**Isha Dharmraj Gupta**

Roll No.:12 Exam. Seat No.:

**Shivang Rajiv Srivastava**

Roll No.:55 Exam. Seat No.:

**Gaurav Dharendra Upadhyay**

Roll No.:58 Exam. Seat No.:

# Abstract

The rampant corona virus disease 2019 (COVID-19) has brought global crisis with its deadly spread to more than 180 countries, and about 3,519,901 confirmed cases along with 247,630 deaths globally as on May 4, 2020.

The absence of any active therapeutic agents and the lack of immunity against COVID-19 increases the vulnerability of the population.

Since there are no vaccines available, social distancing is the only feasible approach to  fight  against  this  pandemic.

Motivated  by  this  notion,  we propose a deep learning based framework for automating the task of monitoring social distancing using surveillance video.

The proposed framework utilizes the YOLO v3 object detection model to segregate humans from the  background and deep-sort approach to track the identified people with the help of bounding boxes and assigned IDs.

The results of the YOLO v3 model are further compared with other popular state-of-the-art models, e.g. faster region-based CNN  (convolution neural network) and single shot detector (SSD) in terms of mean  average precision (mAP), frames per second (FPS) and loss values defined by object classification and localization.

Later, the pairwise vectorized L2 norm is computed based on the three-dimensional feature space obtained by using the centroid  coordinates and dimensions of the bounding box.

The violation index term is  proposed to limit the non-adoption of social distancing protocol. From the experimental analysis, it is observed that the YOLO v3 with deep-sort tracking  scheme displays best results with balanced mAP and FPS score to monitor the social distancing in real-time.

# 

# Table of Contents

Social Distance Detector [i](#_30j0zll)

[(AY](#_1fob9te)20BECSP70520[) i](#_1fob9te)

[Declaration by the Candidate ii](#_3znysh7)

[Major Project Report Approval iv](#_tyjcwt)

[Acknowledgement v](#_3dy6vkm)

[Abstract vi](#_1t3h5sf)

[Table of Contents vii](#_4d34og8)

[List of Figures ix](#_2s8eyo1)

[List of Tables x](#_3rdcrjn)

[List of Abbreviations xi](#_26in1rg)

[1](#_35nkun2) Introduction 1

[1.1](#_1ksv4uv) Introduction 1

[1.2](#_44sinio) Background and Motivation 2

[1.3](#_2jxsxqh) Problem statement 3

[1.4](#_z337ya) Project Objectives 3

[1.5](#_1y810tw) Project Importance 4

[1.6](#_2xcytpi) Scope of Project Work 5

[1.7](#_1ci93xb) Organization of the Report 5

[2](#_3whwml4) Literature Review 6

[2.1](#_2bn6wsx) Survey of Existing System 6

[2.2](#_3as4poj) Problems with Present System 8

[2.3](#_1pxezwc) Limitation existing system or research gap 8

[2.4](#_49x2ik5) Major project Contribution 9

[3](#_147n2zr) Proposed System 10

[3.1](#_3o7alnk) Introduction 10

[3.2](#_23ckvvd) Requirement Analysis 11

3.2.1 Functional Requirements 11

3.2.2 Non- Functional Requirements 11

3.2.2.1 Performance 11

3.2.2.2 Reliability 11

3.2.2.3 Availability 11

[3.3](#_ihv636) Architecture 12

[3.4](#_1hmsyys) Framework 13

[3.5](#_41mghml) Algorithm 13

3.5.1 YOLO Object detection with Open CV 13

3.5.2 Deep Sort 14

[3.6](#_vx1227) Details of Hardware & Software 15

3.6.1 Hardware Requirements 15

3.6.2 Software Requirements 15

[3.7](#_1v1yuxt) Design details 16

[3.7.1](#_4f1mdlm) Design Detail Introduction 16

[3.7.2](#_2u6wntf) Sequence Diagram 18

[3.7.3](#_19c6y18) Activity Diagram 19

[3.7.4](#_3tbugp1) State Chart Diagram 19

[3.7.5](#_28h4qwu) Data Flow Diagram 20

[3.8](#_37m2jsg) Methodology 21

[3.9](#_1mrcu09) Implementation Plan for next semester 22

[3.10 Conclusion and Future work. 2](#_2lwamvv)2

[4](#_111kx3o) References 23

# List of Figures

|  |  |
| --- | --- |
| Fig. 1: An outcome of social distancing as the reduced peak of the epidemic and matching with available health care capacity. | 1 |
| Fig 2.1: Computer Vision technology based on Open CV and YOLO-Stage 1 | 4 |
| Fig 2.2: Computer Vision technology based on Open CV and YOLO-Stage 2 | 4 |
| Fig 3: Open CV Social Distancing Detector Steps | 12 |
| Fig 4: All File Snippet | 16 |
| Fig 5 : Sequence Diagram | 18 |
| Fig 6: Activity Diagram Of Social Distance Detector | 19 |
| Fig 7: State Chart Diagram | 19 |
| Fig 8: Data Flow Diagram | 20 |

# 

# List of Tables

|  |  |
| --- | --- |
| Table 1-1: Literature Review | 6 |

# List of Abbreviations

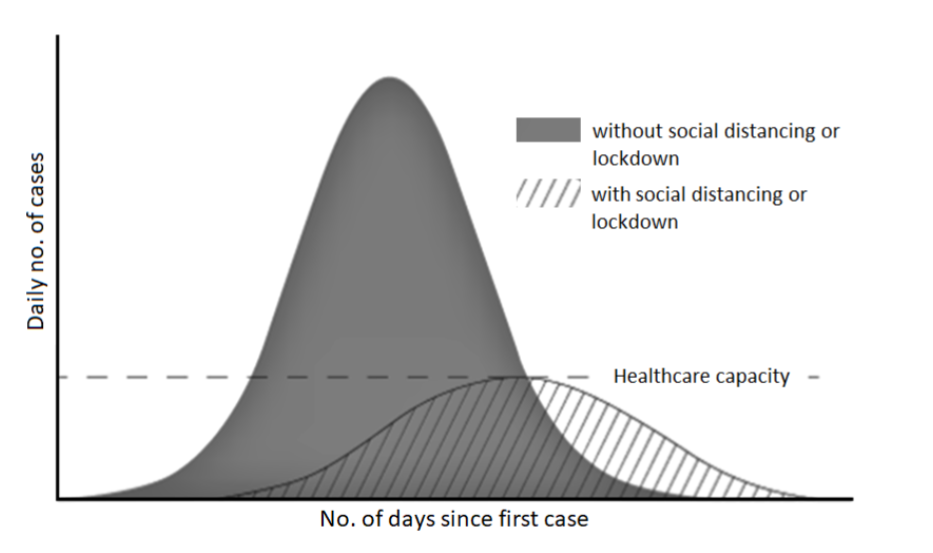
|  |  |
| --- | --- |
|  |  |
| AICTE | All India Council of Technical Education |
| AINA | Advanced Information Networking and Applications |
| ANOVA | Analysis of Variance |
| API | Application programming interfaces |
| ATD | Advanced Technology and Development |
| AWS | Amazon Web services |
| BPM | Business Process Management |
| CCT | Cloud Computing Technology |
| CIO | Chief Information Officer |
| COMPSACW | Computer Software and Applications Conference Workshops |
| CRM | Customer Relationship Management |
| CSA | Cloud Security Alliance |
| CSEE&T | Conference on Software Engineering Education and Training |
| Covid19 | Corona Virus Disease 2019 |
| EaaS | Education as a Service |
| EC2 | Elastic Computing Cloud |
| ECM | Engineering colleges in Mumbai |
| EDF | Empirical distribution function |
| ERP | Enterprise Resource Planning |
| GER | Gross Enrolment Ratio |
| Gmail | Google mail |
| HES | Higher Education Sector |
| IaaS | Infrastructure as a Service |
| ICCCA | International Conference on Computing, Communication and Applications |
| ICUFN | International Conference on Ubiquitous and Future Networks |
| IIS | Internet-based information systems |
| YOLO | You Only Look Once |
| WHO | World Health Organization |

# Introduction

Social distancing is not a new concept. Social distancing is a method used to control the spread of contagious diseases. As the name suggests, social distancing implies that people should physically distance themselves from one another, reducing close contact, and thereby reducing the spread of a contagious disease (such as corona virus).

## Introduction

COVID-19 belongs to the family of corona virus caused diseases, initially reported at Wuhan, China, during late December 2019. Several health care organizations, medical experts and scientists are trying to develop proper medicines and vaccines for this deadly virus, but till date, no success is reported. This situation forces the global community to look for alternate ways to stop the spread of this infectious virus. Social distancing is claimed as the best spread stopper in the present scenario, and all affected countries are locked-down to implement social distancing. This research is aimed to support and mitigate the corona virus pandemic along with minimum loss of economic endeavors, and propose a solution to detect the social distancing among people gathered at any public place.

Fig. 1: An outcome of social distancing as the reduced peak of

the epidemic and matching with available health care capacity.

Social distancing aims at reducing the physical contact between possibly infected individuals and healthy persons. As per the WHO norms it is prescribed that people should maintain at least 6 feet of distance among each other in order to follow social distancing. Fig. 1 indicates that proper social distancing is the best way to reduce infectious physical contact, hence reduces the infection rate. This reduced peak may surely match with the available health care infrastructure and help to offer better facilities to the patients battling against the corona virus pandemic. To study epidemiological phenomena, mathematical models are always the most preferred choice.

## Background and Motivation

Social distancing is surely the most trustworthy technique to stop the spreading of infectious disease, with this belief, in the background of December 2019, when COVID-19 emerged in Wuhan, China, it was opted as an unprecedented measure on January 23, 2020. Within one month, the outbreak in China gained a peak in the first week of February with 2,000 to 4,000 new confirmed cases per day. Later, for the first time after this outbreak, there have been a sign of relief with no new confirmed cases for five consecutive days up to 23 March 2020. This is evident that social distancing measures enacted in China initially, adopted worldwide later to control COVID-19. Many countries, including India and South Korea, for instance, utilizing GPS to track the movements of the suspected or infected persons to monitor any possibility of their exposure among healthy people. In India, the government is using the Arogya Setu App, which worked with the help of GPS and Bluetooth to locate the presence of COVID-19 patients in the vicinity area. It also helps others to keep a safe distance from the infected person. On the other hand, some law enforcement departments have been using drones and other surveillance cameras to detect mass gatherings of people, and taking regulatory actions to disperse the crowd. Such manual intervention in these critical situations might help flatten the curve, but it also brings a unique set of threats to the public and is challenging to the

workforce. Many research findings have been reported in the last few years. Crowd counting emerged as a promising area of research, with many societal applications. Esheletal., focused on crowd detection and person count by proposing multiple height tomographic for head top detection and solved the occlusions problem associated with video surveillance related applications. Chenetal.

developed an electronic advertising application based on the concept of crowd counting. In similar application, Chih-Wen et al. proposed a vision-based people counting model. Following this, Yao et al. generated inputs from stationary cameras to perform background subtraction to train the model for the appearance and the foreground shape of the crowd in videos.

## Problem statement

Design a program to implement a social distancing detector using Open CV, Deep Learning, and Computer Vision with the use of webcam to detect people gathering and not following social distancing norms using square boxes. Display the count of violation in the bottom of the screen.Social distancing is not a new concept. Social distancing is a method used to control the spread of contagious diseases. As the name suggests, social distancing implies that people should physically distance themselves from one another, reducing close contact, and thereby reducing the spread of a contagious disease (such as corona virus).Our YOLO object detection files including the CNN architecture definition, pre-trained weights, and class names are housed in the yolo-coco/directory. This YOLO model is compatible with OpenCV’s DNN module.Our social distance detector application logic resides in the social\_distance\_detector.py script. This file is responsible for looping over frames of a video stream and ensuring that people are maintaining a healthy distance from one another during a pandemic.It is compatible with both video files and webcam streams.

## Project Objectives

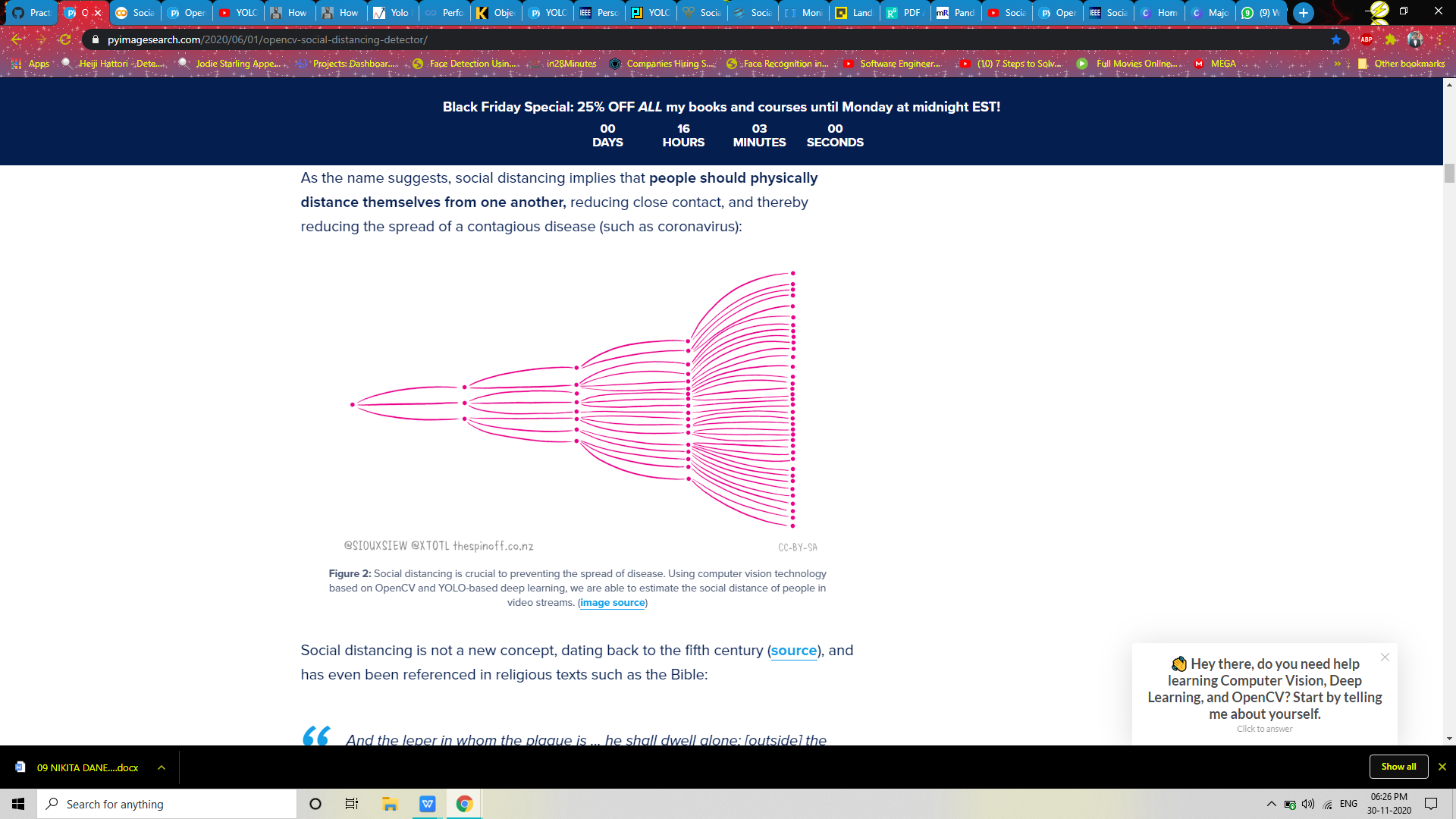
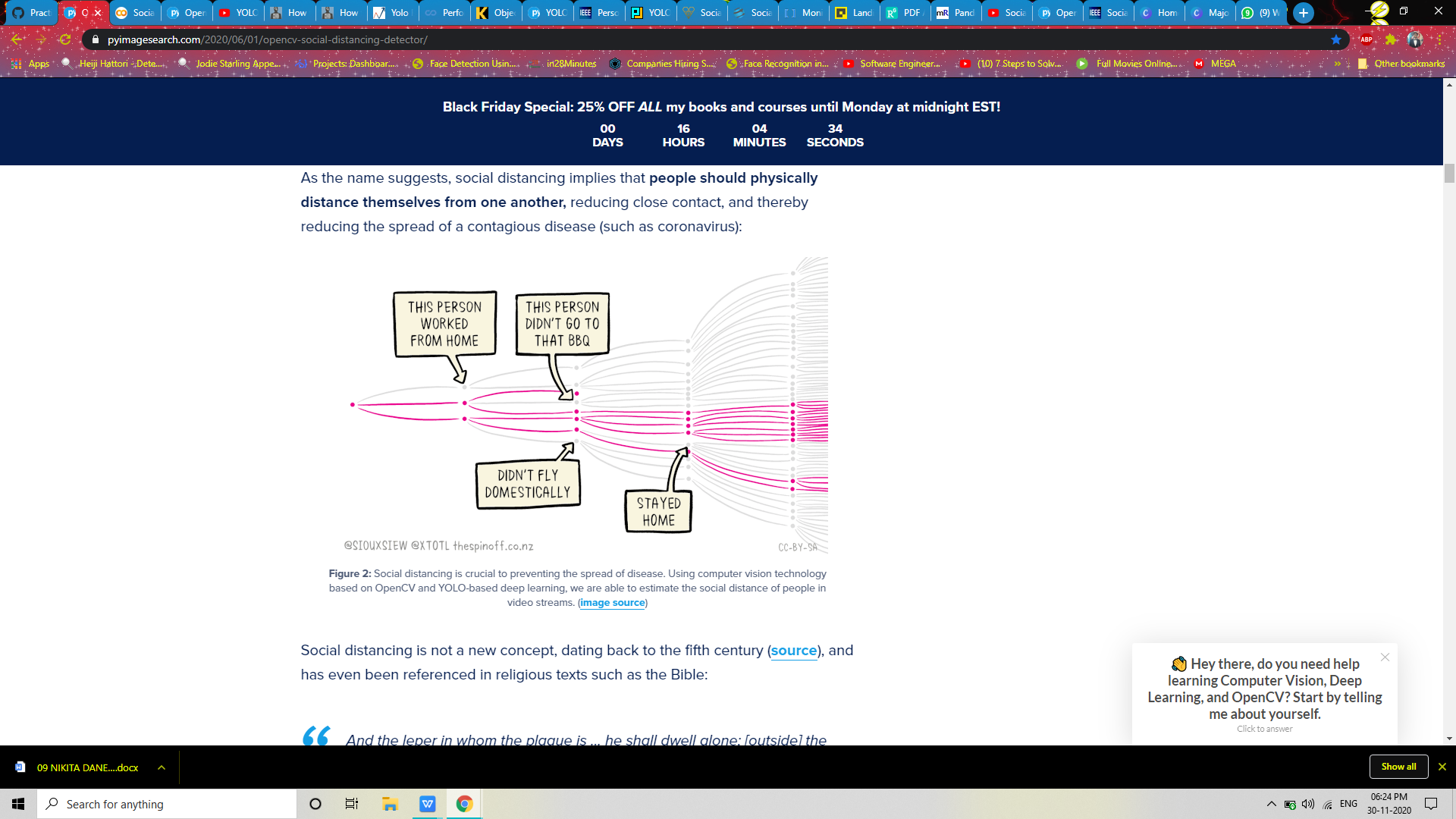
Social distancing is an effective measure against the novel corona virus Disease 2019 (COVID-19) pandemic. However, the general public is not used to keep an imaginary safety bubble around themselves. An automatic warning system can help and augment the perceptive capabilities of individuals. Deploying such an active surveillance system requires serious ethical considerations and smart system design. The first challenge is privacy. If data is recorded and stored, the privacy of individuals may be violated intentionally or unintentionally. Recording data and labeling individuals who do not follow the measures will breach individuals’ rights in free-societies. These concerns can addressed with some additional measures such as prior consents for such working environments, hiding a person’s identity in general, and maintaining transparency about its fair uses within limited stakeholders. Second, the detector must not discriminate. The safest way to achieve this is by building an AI-based detection system. Removing the human out of the detection loop may not be enough – the detector must also be design free. Higher number of false positives may raise discomfort and panic situations among people being observed. Since this application is intended to be used in any working environment; accuracy and precision are highly desired to serve the purpose.

## Project Importance

Several health-care organizations, medical experts and scientists are trying to develop proper medicines and vaccines for this deadly virus, but till date, no success is reported. This situation forces the global community to look for alternate ways to stop the spread of this infectious virus. Social distancing is claimed as

the best spread stopper in the present scenario, and all affected countries are locked-down to implement social distancing.

1. A recent study indicates that social distancing is an important containment measure and essential to prevent SARS-CoV-2, because people with mild or no symptoms may fortuitously carry corona infection and can infect others.
2. Fig. 1.1 indicates that proper social distancing is the best way to reduce infectious physical contact, hence reduces the infection rate.
3. When COVID-19 emerged in Wuhan, China, it was opted as an unprecedented measure on January 23, 2020. Within one month, the outbreak in China gained a peak in the first week of February with 2,000 to 4,000 new confirmed cases per day. Later, for the first time after this outbreak, there was a sign of relief with no new confirmed cases for five consecutive days up to 23 March 2020. This is evidence that social distancing measures enacted in China.
4. Later, many countries implemented a lock-down to ensure social distancing was carried out imperatively to curb the spread of the pandemic.

## Scope of Project Work

Human detection using visual surveillance system is an established area of research which is relying upon manual methods of identifying unusual activities, however, it has limited capabilities. Although human detection is an ambitious goal, due to a variety of constraints such as low-resolution video, varying articulated pose, clothing, lighting and background complexities and limited machine vision capabilities, wherein prior knowledge on these challenges can improve the detection performance. Detecting an object which is in motion, incorporates two stages: object detection and object classification. The primary stage of object detection could be achieved by using background subtraction, optical flow and spatio-temporal filtering techniques. In the background subtraction method, the difference between the current frame and a background frame (first frame), at pixel or block level is computed. Once an object is detected, classification techniques can be applied to identify a human on the basis of shape, texture or motion-based features. In shape-based methods, the shape related information of moving regions such as points, boxes and blobs are determined to identify the human.

## Organization of the Report

One of the initial steps of developing a social distance detecting system was to study the existing technologies and works that are in the direction of object detection using YOLO and deep-sort. Emerging technologies have developed a social distance monitoring approach but it does not contain any statistical analysis or implementation of privacy-related discussion other than the violation index. Very recently, several prototypes utilizing machine learning and sensing technologies have been developed to help social distancing monitoring. These prototypes helped us understand the techniques and data flow of the required system. The system required person detection using a trained neural network. The YOLO object detector files include the CNN architecture definition, pre-trained weights, and class names in the directory. This YOLO model is compatible with OpenCV’s DNN module. Spatial detection to determine social distance violations was needed. The detector creates detection based output as:

* Identify a human on the basis of shape, texture or motion-based features.
* Display the violation with red bounding boxes on the screen.

Count and display the total number of violations in the frame and display it at the bottom of the screen. Rest of the report is organized as follows: in section 2 literature review of project is discussed, section 3 social distancing proposed system is described followed by section for that is reference.

# Literature Review

Here we will elaborate the aspects like the literature survey of the project and what all projects are existing and have actually been used in the market which the makers of this project took the inspiration from and thus decided to go ahead with the project covering with the problem statement.

## Survey of Existing System

|  |  |  |  |
| --- | --- | --- | --- |
| Reference Paper | Journal / Conference and Year | Algorithm and tools used | Observation & Result |
| A Vision-based Social Distancing and Critical Density Detection System for COVID-19 | 8 Jul 2020. The Ohio State University, Columbus, OH 43210, USA | Object detection with deep learning and deep convolutional neural networks | Two different deep CNN based object detectors: Faster R-CNN and YOLOv4 and both detectors achieved real-time performance. |
| Pandemic Politics:  Timing State-Level Social Distancing  Responses to COVID-19 | On 28 March 2020, this paper was made available under a CC-BY-NC-ND 4.0 International license . | Data Collection and Graph Reading | Different states had different pace in adopting social distancing policies |
| Social Distancing Detection with Deep Learning Model | Date of Conference: 24-26 Aug. 2020. Published in: 2020 8th (ICIMU) | Object detection using the CNN model such as R-CNN and YOLO | pedestrians walking within the specified space will be counted for people density measurement |

Many authors used synthetic location-specific contact patterns to simulate the ongoing trajectory of the outbreak using susceptible-exposed-infected-removed (SEIR) models. It was also suggested that premature and sudden lifting of social distancing could lead to an earlier secondary peak, which could be flattened by relaxing the interventions gradually. As we all understand, social distancing though essential but economically painful measures to flatten the infection curve. Adolph et al. highlighted the situation of the United States of America, where due to lack of common consent among all policymakers it could not be adopted at an early stage, which is resulting into on-going harm to public health. Although social distancing impacted economic productivity, many researchers are trying hard to overcome the loss. Following from this context, Kylie et al. studied the correlation between the strictness of social distancing and the economic status of the region. The study indicated that intermediate levels of activities could be permitted while avoiding a massive outbreak. Since the novel corona virus pandemic began, many countries have been taking the help of technology based solutions in different capacities to contain the outbreak. Pedestrian detection can be regarded as either a part of a general object detection problem or as a specific task of detecting pedestrians only. A detailed survey of 2D object detectors, as well as datasets, metrics, and fundamentals, can be found. Another survey focuses on deep learning approaches for both generic object detection and pedestrian detection. State-of-the-art object detectors use deep learning approaches, which are usually divided into two categories. The first one is called two-stage detectors, mostly based on R-CNN, which starts with region proposals and then performs the classification and bounding box regression. The second one is called one-stage detectors, of which the famous models are YOLOv1-v4, SSD, RetinaNet, and EfficientDet. In addition to these anchor-based approaches, there are also some anchor-free detectors: CornerNet, CenterNet, FCOS, and RepPoints. These models were usually evaluated on datasets of Pascal VOC and MS COCO. The accuracy and real-time performance of these approaches are good enough for deploying pre-trained models for social distancing detection. Social distancing monitoring. Emerging technologies can assist in the practice of social distancing. A recent work has identified how emerging technologies like wireless, networking, and artificial intelligence (AI) can enable or even enforce social distancing. The work discussed possible basic concepts, measurements, models, and practical scenarios for social distancing. Another work has classified various emerging techniques as either anthropocentric or smart-space categories, along with the SWOT analysis of the discussed techniques. A specific social distancing monitoring approach that utilizes YOLOv3 and deep-sort was proposed to detect and track pedestrians followed by calculating a violation index for non-social-distancing behaviors. The approach is interesting but results do not contain any statistical analysis. Furthermore, there is no implementation or privacy-related discussion other than the violation index. Social distancing monitoring is also defined as a visual social distancing (VSD) problem in. The work introduced a skeleton detection based approach for inter-personal distance measuring. It also discussed the effect of social context on people’s social distancing and raised the concern of privacy. The discussions are inspirational but again it does not generate solid results for social distancing monitoring and leaves the question open. Very recently, several prototypes utilizing machine learning and sensing technologies have been developed to help social distancing monitoring. Landing AI has proposed a social distancing detector using a surveillance camera to highlight people whose physical distance is below the recommended value. A similar system was deployed to monitor worker activity and send real-time voice alerts in a manufacturing plant. In addition to surveillance cameras, LiDAR based and stereo camera based systems were also proposed, which demonstrated that different types of sensors besides surveillance cameras can also help. The above systems are interesting, but recording data and sending intrusive alerts might be unacceptable by some people. On the contrary, we propose a non-intrusive warning system with softer omnidirectional audio-visual cues. In addition, our system evaluates critical social density and modulates inflow into a region-of-interest.

## Problems with Present System

Emerging technologies can assist in the practice of social distancing. A recent

work has identified how emerging technologies like wireless, networking, and artificial intelligence (AI) can enable or even enforce social distancing. The work discussed possible basic concepts, measurements, models, and practical scenarios for social distancing. Another work has classified various emerging techniques as either anthropocentric or smart-space categories, along with the SWOT analysis of the discussed techniques. A specific social distancing monitoring approach that utilizes YOLOv3 and deep-sort was proposed to detect and track pedestrians followed by calculating a violation index for non-social distancing behaviors. The approach is interesting but results do not contain any statistical analysis. Furthermore, there is no implementation or privacy-related discussion other than the violation index. The discussions are inspirational but again it does not generate solid results for social distancing monitoring and leaves the question open. Against this backdrop, we propose a non-intrusive augmentative AI-based active surveillance system for sending omnidirectional visual/audio cues when a social distancing breach is detected. The proposed system uses a pre-trained deep convolutional neural network (CNN) to detect individuals with bounding boxes in a given video frame. Our system is real-time and does not record data. To help increase the speed of deep learning-based object detectors, both Single Shot Detectors (SSDs) and YOLO use a **one-stage detector strategy**. These algorithms treat object detection as a regression problem, taking a given input image and simultaneously learning bounding box coordinates and corresponding class label probabilities.

## Limitation of existing system or research gap

Object detection in the image domain is a fundamental computer vision problem. The goal is to detect instances of semantic objects that belong to certain classes, e.g., humans, cars, buildings. Recently, object detection benchmarks have been dominated by deep convolutional neural networks (CNNs) models. Very recently, several prototypes utilizing machine learning and sensing technologies have been developed to help social distancing monitoring. Landing AI has proposed a social distancing detector using a surveillance camera to highlight people whose physical distance is below the recommended value. A similar system was deployed to monitor worker activity and send real-time voice alerts in a manufacturing plant. The above systems are interesting, but recording data and sending intrusive alerts might be unacceptable by some people. On the contrary, we propose a non-intrusive warning system with softer omnidirectional visual cues.

## Major project Contribution

Social distancing is not a new concept. Social distancing is a method used to control the spread of contagious diseases. As the name suggests, social distancing implies that people should physically distance themselves from one another, reducing close contact, and thereby reducing the spread of a contagious disease (such as corona virus)

Our YOLO object detection files including the CNN architecture definition, pre-trained weights, and class names are housed in the yolo-coco/directory. This YOLO model is compatible with OpenCV’s DNN module.This project will play an important role in the area where large amount of people can be expected such as a mall or movie theatre or airport.With the help of this project we can make sure the people are following social distancing norm.

# Proposed System

## Introduction

Object detection in the image domain is a fundamental computer vision problem. The goal is to detect instances of semantic objects that belong to certain classes, e.g., humans, cars, buildings. Recently, object detection benchmarks have been dominated by deep convolutional neural networks (CNNs) models. These models are usually trained by supervised learning, with techniques like data augmentation to increase the variety of data. he generalization capability of the state-of-the-art is good enough for deploying pre-trained models to new environments. For 2D object detection, even with different camera models, angles, and illumination conditions, pre-trained models can still achieve good performance.

Therefore, a pre-trained state-of-the-art deep learning based pedestrian detector can be directly utilized for the task of social distancing monitoring. We propose to use a fixed camera to detect individuals in a region of interest (ROI) and measure the interpersonal distances in real time without data recording. The proposed system sends a non-intrusive visual cue to the screen warn the crowd if any social distancing breach is detected. We conducted 3 case studies to evaluate the proposed method. Each case utilizes a different pedestrian crowd dataset. In the present article, a deep learning based framework is proposed that utilizes object detection and tracking models to aid in the social distancing remedy for dealing with the escalation of COVID-19 cases. In order to maintain the balance of speed and accuracy, YOLO v3 alongside the deep-sort are utilized as object detection and tracking approaches while surrounding each detected object with the bounding boxes. Later, these bounding boxes are utilized to

compute the pairwise L2 norm with computationally efficient vectorized representation for identifying the clusters of people not obeying the order of social distancing. Furthermore, to visualize the clusters in the live stream, each bounding box is color-coded based on its association with the group where people belonging to the same group are represented with the same color. Each surveillance frame is also accompanied with the streamline plot depicting the statistical count of the number of social groups and an index term (violation index) representing the ratio of the number of people to the number of groups. Furthermore, estimated violations can be computed by multiplying the violation index with the total number of social groups.

## Requirement Analysis

In requirements analysis encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.Project planning is part which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment. Initially, the project scope is defined and the appropriate methods for completing the project are determined.

**3.2.1 Functional Requirement**

The functional requirements for a system describe what the system does.

1. The developed system should be able to detect the distance between two or more people who are or are not following social distancing.
2. System shall show the the total count message to the user.
3. System must provide the quality of service to the user that is not distort the output.
4. System must provide accuracy for people detection and people count.

**3.2.2 Nonfunctional requirements**

**3.2.2.1 Performance**

Social distance detector will take access of the web camera and take the video as input. Then apply social distancing on that video and provide an accurate result.

**3.2.2.2 Reliability**

This software will deliver on the functional requirements and provides the users web cam shows the video correctly. It's also very easy to use and reduces the work of the security guard and other surveillance authority.

**3.2.2.3 Availability**

This system currently will be available to use on a laptop and pc provided to got the specific system requirements.

## Architecture

Deep learning has brought techniques for a variety of tasks and challenges including medical diagnosis, machine translation, speech recognition, and much more.Most of these tasks are centred around object classification, detection, segmentation, tracking, and recognition. The convolution neural network (CNN) based architectures have shown significant performance improvements that are leading towards the high quality of object detection.In order to maintain the balance of speed and accuracy, YOLO v3 alongside deep-sort are being used as object detection and tracking approaches while surrounding each detected object with the bounded boxes.Using Open CV, computer vision, and deep learning for implementing a social distancing detector.YOLO object detector. This YOLO model is compatible with OpenCV’s DNN module.Using YOLO with Open CV requires a bit more output processing than other object detection methods (such as Single Shot Detectors or Faster R-CNN), so in order to keep our code tidy, we will implement a detect\_people function that encapsulates any YOLO object detection logic.

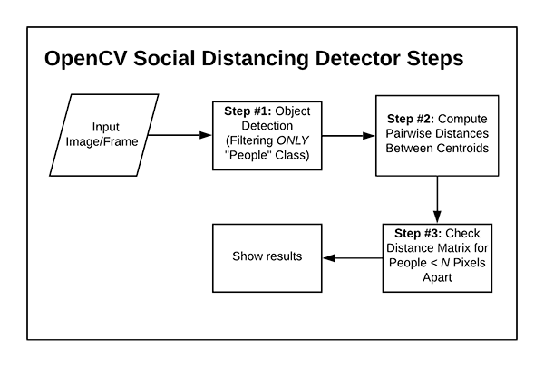


Fig-3: Open CV Social Distancing Detector Steps

* If no video file path is provided, your computer’s first webcam will be used by default.
* Apply object detection to detect all people (and only people) in a video stream.
* Compute the pairwise distances between all detected people
* Based on these distances, check to see if any two people are less than N pixels apart.

## Framework

The implementation consists of 4 modules:

1. social\_distancing\_config.py:The whole configuration that is the distance in pixel which needs to be considered for following social distancing norm, setting up yolo model path , initializing minimum probability to filter weak detection along with the threshold when applying non-maxima suppression.

2. detection.py: This file consist of the detect\_people function which uses the configuration from earlier to detect the people in the video clip and segregate them.

3. social\_distance\_detector1.py: This can be consider as the first model putting together everything from earlier and implementing our project in which webcam is used to access the video to detect the video and show the output of the people following the social distancing norm.

4. social\_distance\_detector.py: This file is responsible for looping over frames of a video stream and ensuring that people are maintaining a healthy distance from one another during a pandemic. It is compatible with both video files and webcam streams..

## Algorithm

**3.5.1.YOLO object detection with Open CV:**

For object detection, another competitor of SSD is YOLO. This method can predict the type and location of an object by looking only once at the image. YOLO considers the object detection problem as a regression task instead of classification to assign class probabilities to the anchor boxes. A single convolutional network simultaneously predicts multiple bounding boxes and class probabilities. Majorly, there are three versions of YOLO: v1, v2 and v3. In contrast, YOLO v3 performs multi-label classification with the help of logistic classifiers instead of using softmax as in case of YOLO v1 and v2. In YOLO v3, Darknet-53 as a backbone architecture that extracts features maps for classification was proposed. In contrast to Darknet-19, Darknet-53 consists of residual blocks (short connections) along with the up-sampling layers for concatenation and added depth to the network. YOLO v3 generates three predictions for each spatial location at different scales in an image, which eliminates the problem of not being able to detect small objects efficiently. Each prediction is monitored by computing objectiveness, boundary box regressor and classification scores.The YOLO is a great example of a single stage detector. First introduced in 2015 by Redmon et al., their paper, [You Only Look Once: Unified, Real-Time Object Detection](https://arxiv.org/abs/1506.02640" \t "https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/_blank), details an object detector capable of super real-time object detection, obtaining 45 FPS on a GPU.YOLO has gone through a number of different iterations, including [YOLO9000: Better, Faster, Stronger](https://arxiv.org/abs/1612.08242" \t "https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/_blank) (i.e., YOLOv2), capable of detecting over 9,000 object detectors.Our YOLO object detector files including the CNN architecture definition, pre-trained weights, and class names are housed in the

yolo-coco/ directory. This YOLO model is compatible with OpenCV’s DNN module.

**3.5.2. Deep-sort:**

deep-sort is a deep learning based approach to track custom objects in a video. In the present research, deep-sort is utilized to track individuals present in the surveillance footage. It makes use of patterns learned via detected objects in the images which is later combined with the temporal information for predicting associated trajectories of the objects of interest. It keeps track of each object under consideration by mapping unique identifiers for further statistical analysis. deep-sort is also useful to handle associated challenges such as occlusion, multiple viewpoints, non-stationary cameras and annotating training data. For effective tracking, the Kalman filter and the Hungarian algorithm are used. Kalman filter is recursively used for better association, and it can predict future positions based on the current position. Hungarian algorithm is used for association and id attribution that identifies if an object in the current frame is the same as the one in the previous frame.

## Details of Hardware & Software

**3.6.1 Hardware requirements**

1- Windows 10 (64-bit), Windows 8.1 (64-bit), Windows 7 (64-bit) \*32-bit versions are supported but not recommended

2- RAM 4GB memory or higher

3- Multi high-speed hard disks or SSDs, 8 GB free on installation drive + free space for images (20 GB+)

4- 1.9 GHz CPU Intel quad-core processor or better

5- Webcam HD video capture: Up to 1280 X 720 pixels

**3.6.2 Software requirements**

1. Anaconda Software 4.0 or above
2. Python3 or above version

3- Open CV 3.0

4- Different modules in Python such as CV2, Numpy, Scipy, Imutilis, OS

## Design details

3.7.1. Design Detail Introduction

In this project, we are using a YOLO-COCO Detector for our task.We have created 2 different folders one for keeping the detection and configuration and other consisting of the pre trained model. Yolo-coc file consists of coco.names, yolov3.cfg, yolov3.weights.We are using the pre trained models for object detection. Other than the two folders we have kept in the main folder in the main folder we have the main python file which we are going to run in order to check if the program is running successfully or not.

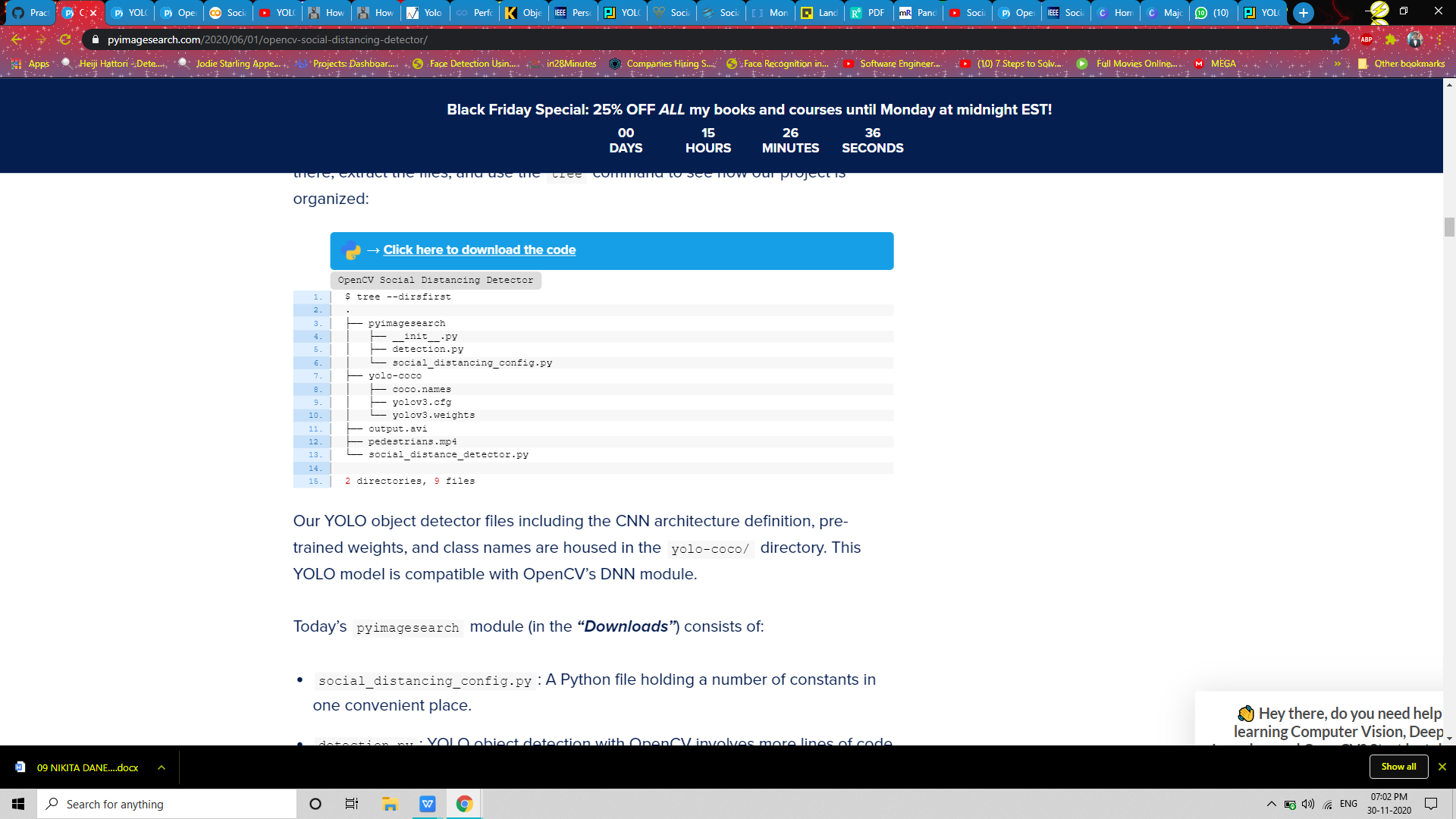


Fig 4: All File Snippet

To help keep our code tidy and organized, we’ll be using a configuration file to store important variables.Let’s take a look at the  social\_distancing\_config.py file inside the pyimagesearch module:

Here, we have the path to the YOLO object detection model. We also define the minimum object detection confidence and [non-maxima suppression](https://www.pyimagesearch.com/2014/11/17/non-maximum-suppression-object-detection-python/" \t "https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/_blank) threshold.We have two more configuration constants to define:The USE\_GPU Boolean on  indicates whether your NVIDIA CUDA-capable GPU will be used to speed up inference (requires that [OpenCV’s “dnn” module be installed with NVIDIA GPU support](https://www.pyimagesearch.com/2020/02/03/how-to-use-opencvs-dnn-module-with-nvidia-gpus-cuda-and-cudnn/" \t "https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/_blank)). MIN\_DISTANCE = 50 Defines the minimum distance (in pixels) that people must stay from each other in order to adhere to social distancing protocols.

We’ll be using the [YOLO object detector](https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/" \t "https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/_blank) to detect people in our video stream.

Using YOLO with Open CV requires a bit more output processing than other object detection methods (such as [Single Shot Detectors](https://www.pyimagesearch.com/2017/09/18/real-time-object-detection-with-deep-learning-and-opencv/" \t "https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/_blank) or Faster R-CNN), so in order to keep our code tidy, let’s implement a detect\_people function that encapsulates any YOLO object detection logic. Open up the detection.py file inside the pyimagesearch module, and let’s get started:We begin with imports, including those needed from our configuration file on Lines 2 and 3 — the  NMS\_THRESH and MIN\_CONF (refer to the previous section as needed). We’ll also take advantage of NumPy and Open CV in this script (Lines 4 and 5).Our script consists of a single function definition for detecting people — let’s define that function now: Beginning on Line 7, we define detect\_people; the function accepts four parameters:

frame: The frame from your video file or directly from your webcam

net: The pre-initialized and pre-trained YOLO object detection model

ln: The YOLO CNN output layer names

personIdx: The YOLO model can detect many types of objects; this index is specifically for the person class, as we won’t be considering other objects

Line 10 grabs the frame dimensions for scaling purposes.We then initialize our results list, which the function ultimately returns. The results consist of (1) the person prediction probability, (2) bounding box coordinates for the detection, and (3) the centroid of the object.

Given our frame, now it is time to perform inference with YOLO:

Pre-processing our frame requires that we construct a blob (Lines 16 and 17). From there, we are able to perform object detection with YOLO and Open CV (Lines 18 and 19).Lines 23-25 initialize lists that will soon hold our bounding boxes, object centroids, and object detection confidences. Now that we’ve performed inference, let’s process the results:

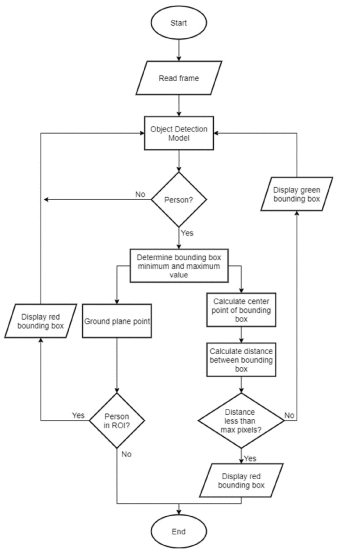
Looping over each of the layerOutputs and detections (Lines 28-30), we first extract the classID and confidence (i.e., probability) of the current detected object (Lines 33-35).From there, we verify that (1) the current detection is a person and (2) the minimum confidence is met or exceeded (Line 40).Assuming so, we compute bounding box coordinates and then derive the center (i.e., centroid) of the bounding box (Lines 46 and 47). Notice how we scale (i.e., multiply) our detection by the frame dimensions we gathered earlier.Using the bounding box coordinates, Lines 51 and 52 then derive the top-left coordinates for the object.We then update each of our lists (boxes, centroids, and confidences) via Lines 56-58.Next, we apply non-maxima suppression:

The purpose of non-maxima suppression is to suppress weak, overlapping bounding boxes. Line 62 applies this method (it is built-in to Open CV) and results in the idxs of the detections.Assuming the result of NMS yields at least one detection (Line 65), we loop over them, extract bounding box coordinates, and update our results list consisting of the:

Confidence of each person detection, Bounding box of each person, Centroid of each person and finally, we return the results to the calling function.

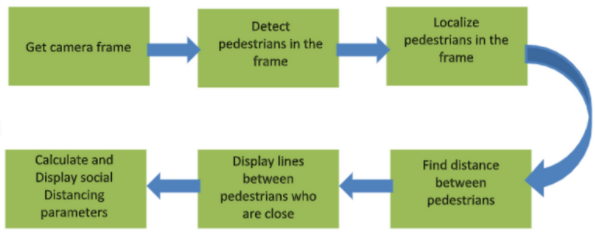
**3.7.2 Sequence Diagram**

A sequence diagram is a graphical view of a scenario that shows object interaction in a time based sequence: what happens first, what happens next. Sequence diagrams establish the role of objects and help provide essential information to determine class responsibilities and interfaces. This type of diagram is best used during the early analysis phase in design because they are simple and easy to comprehend. Sequence diagrams are normally associated with use cases.



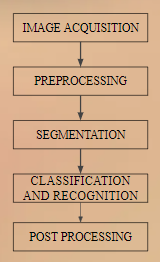
**Fig 5 : Sequence Diagram**

**3.7.3 Activity Diagram**



**Fig 6: Activity Diagram Of Social Distance Detector**

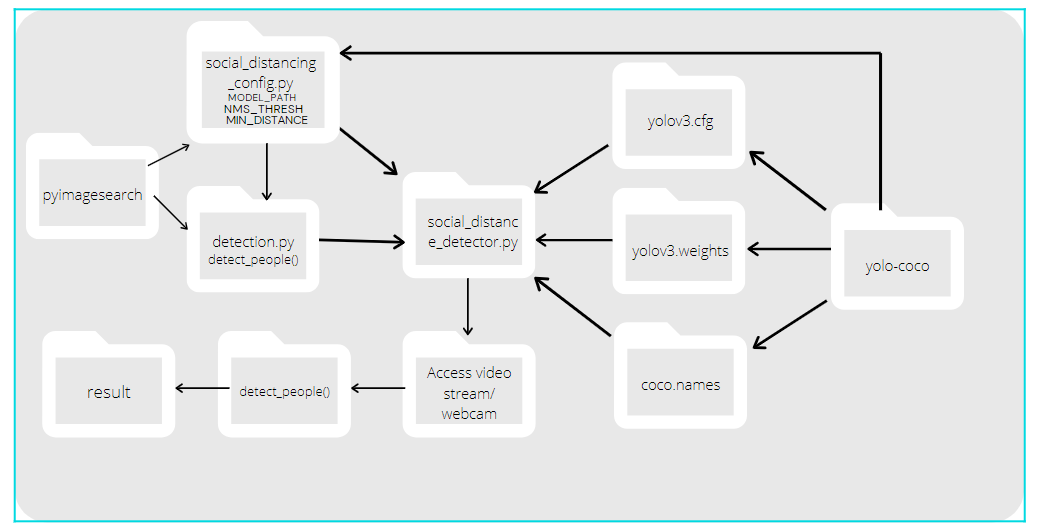
**3.7.4 State Chart Diagram**

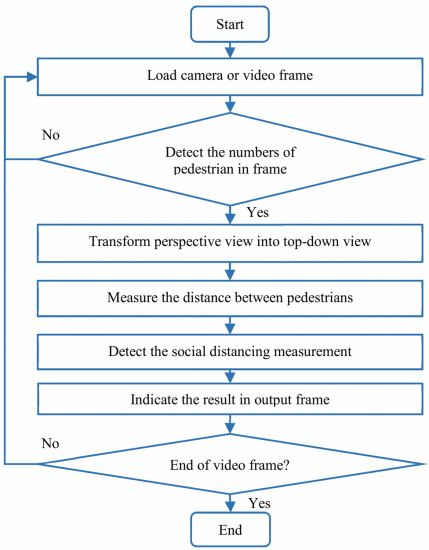


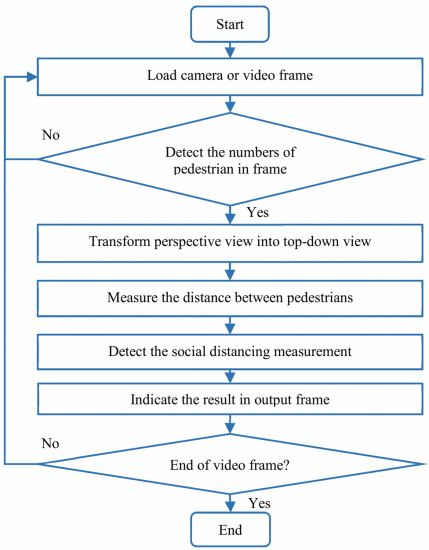
**Fig 7: State Chart Diagram**

**3.7.5 Data Flow Diagram**

Data flow diagram (DFD) is also called as Bubble Chart is a graphical technique, which is used to represent information flow, and transformers those are applied when data moves from input to output. DFD represents system requirements clearly and identify transformers those become programs in design. DFD may further partitioned into different levels to show detailed information flow e.g. level 0, level 1 etc.







**Fig 8: Data Flow Diagram**

## Methodology

The emergence of deep learning has brought the best performing techniques for a wide variety of tasks and challenges including medical diagnosis, machine translation, speech recognition, and a lot more. Most of these tasks are centred around object classification, detection, segmentation, tracking, and recognition. In recent years, the convolution neural network (CNN) based architectures have shown significant performance improvements that are leading towards the high quality of object detection, which presents the performance of such models in terms of mAP and FPS on standard benchmark datasets. In the present article, a deep learning based framework is proposed that utilizes object detection and tracking models to aid in the social distancing remedy for dealing with the escalation of COVID-19 cases. In order to maintain the balance of speed and accuracy, YOLO v3 alongside the deep-sort are utilized as object detection and tracking approaches while surrounding each detected object with the bounding boxes. Later, these bounding boxes are utilized to compute the pairwise L2 norm with computationally efficient vectorized representation for identifying the clusters of people not obeying the order of social distancing. Furthermore, to visualize the clusters in the live stream, each bounding box is color-coded based on its association with the group where people belonging to the same group are represented with the same color. Each surveillance frame is also accompanied with the streamline plot depicting the statistical count of the number of social groups and an index term (violation index) representing the ratio of the number of people to the number of groups. Furthermore, estimated violations can be computed by multiplying the violation index with the total number of social groups.

## Implementation Plan for next semester

In continuation, next semester we will be dealing with the implementation of code of our project i.e. object detection for people part using yolo and after that implementing social distance detector. After implementing the first part of our code we will try to upgrade the detection part , We can obtain a higher frame processing rate by (1) utilizing an NVIDIA CUDA-capable GPU and (2) compiling/installing OpenCV’s “dnn” module with NVIDIA GPU support.Our social distancing detector did not leverage a proper camera calibration, meaning that we could not (easily) map distances in pixels to actual measurable units (i.e., meters, feet, etc.).We will try to use the bird eye view implementation in the next semester if that works properly our efficiency will increase by a huge percentage.

## Conclusion and Future work

A methodology of social distancing detection tool using a deep learning model is proposed. By using computer vision, the distance between people can be estimated and any noncompliant pair of people will be indicated with a red frame.We can have live video surveillance to fight against covid-19 spread. Tracking the crowd movement across the day time. Hot-spot area can be monitored by security forces from central station.Furthermore, the work can be further improved by optimizing the pedestrian detection algorithm, integrating other detection algorithms such as mask detection and human body temperature detection, improving the computing power of the hardware, and calibrating the camera perspective view.

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: Faster RCNN, SSD, and YOLO v3, where YOLO v3 illustrated the efficient performance with balanced FPS and mAP score. 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.

# 4 References

|  |  |
| --- | --- |
| [1] | https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/ |
| [2] | https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/ |
| [3] | https://www.pyimagesearch.com/2020/02/10/opencv-dnn-with-nvidia-gpus-1549-faster-yolo-ssd-and-mask-r-cnn/ |
| [4] | https://www.youtube.com/watch?v=4eIBisqx9\_g |
| [5] | https://machinelearningmastery.com/how-to-perform-object-detection-with-yolov3-in-keras/ |
| [6] | https://www.analyticsvidhya.com/blog/2018/12/practical-guide-object-detection-yolo-framewor-python/ |
| [7] | https://www.coursera.org/projects/real-time-object-detection-yolo |
| [8] | https://www.kdnuggets.com/2018/09/object-detection-image-classification-yolo.html |
| [9] | https://ieeexplore.ieee.org/document/9204934 |
| [10] | https://pjreddie.com/darknet/yolo/ |
| [11] | https://heartbeat.fritz.ai/social-distance-detector-with-python-yolov4-darknet-and-opencv-62e66c15c2a4 |
| [12] | https://axiscades.com/social-distancing-detector.html |
| [13] | https://deepai.org/publication/monitoring-covid-19-social-distancing-with-person-detection-and-tracking-via-fine-tuned-yolo-v3-and-deepsort-techniques |
| [14] | https://analyticsindiamag.com/landing-ais-ai-enabled-social-distancing-detection-tool/ |
| [15] | https://www.researchgate.net/publication/342763051\_A\_Vision-based\_Social\_Distancing\_and\_Critical\_Density\_Detection\_System\_for\_COVID-19#pf7 |
| [16] | https://www.medrxiv.org/content/10.1101/2020.03.30.20046326v1 |
| [17] | https://www.youtube.com/watch?v=ZKJwJv7iVRE |
| [18] | https://ieeexplore.ieee.org/abstract/document/9243478 |
| [19] | https://www.canva.com/design/DAEOn6hS9VQ/lSIZtUC1uI3PcDLG\_IRYTw/edit |
| [20] | https://pjreddie.com/darknet/yolo/ |
| [21] | https://towardsdatascience.com/covid-19-ai-enabled-social-distancing-detector-using-opencv-ea2abd827d34 |
| [22] | https://www.analyticsvidhya.com/blog/2020/05/social-distancing-detection-tool-deep-learning/ |
| [23] | https://www.irjet.net/archives/V7/i8/IRJET-V7I8698.pdf |
| [24] | https://towardsdatascience.com/monitoring-social-distancing-using-ai-c5b81da44c9f |