A Mini Project Report

Social Distance Analyser

Submitted in partial fulfilment for the degree of Bachelor of Technology in Information Technology

Submitted by

Snehal Salunke

Vaishnavi Sawalkar

Sakshi Sawaant

Under the guidance of

Prof. Kumud Wasnik

Usha Mittal Institute of Technology

S.N.D.T Women's University

Mumbai - 400049

2020 - 2021

Approval Sheet

This is to certify that Snehal Salunke, Vaishnavi Sawalkar and Sakshi Sawant has completed the seminar report on the topic "Social Distancing Analyser" satisfactorily in partial fulfillment for the Bachelor's Degree in Computer Science and Technology under the guidance of Prof.Kumud Wasnik during the year 2020-2021 as prescribed by SNDT Women's University.

Guide	Head Of Department
Prof.Kumud Wasnik	Dr.Sanjay Pawar

Principal
Dr.Sanjay Pawar

Examiner 1	Examiner 2

Declaration

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

(Signature)

Snehal Salunke (39)
Vaishnavi Sawalkar (40)
Sakshi Sawant (41)

Date: 08/06/2021

Abstract

Abstract

COVID-19 is a disease caused by a severe respiratory syndrome coronavirus. It was identified in December 2019 in Wuhan, China, and soon spread globally resulting in an ongoing pandemic that caused infected cases including many deaths. CoronaVirus is spread between people during close contact. This virus spreads from person to person through their physical contact or through the droplets which fall while people cough or sneeze, hence it has become necessary for the nations to call for the lockdown or being home quarantined in order to prevent the spread of the virus as it is highly contagious.But it is not always possible to stay indoors and people have to go out for their daily basic necessities and therefore the World Health Organization recommended some guidelines for the prevention of the virus. Among these guidelines, the most important guideline is to maintain a safe distance of minimum of 6 feet from each other while people are in crowded places.

Motivating to this notion, this paper presents a methodology for a Deep Learning Model for the Social Distancing Classification of people. By exploiting the YOLOv4 (You Only Look Once) approach, a deep learning detection technique is developed for detecting and tracking people. An algorithm is also implemented for measuring and classifying the distance between persons and to automatically check if social distancing rules are respected or not.

Hence, this work aims at minimizing the spread of the COVID-19 virus by evaluating if and how persons comply with social distancing rules. The proposed approach is applied to images or video stream input to the system, to establish a Deep Learning system for people tracking and social distancing classification and accordingly give alerts.

The open-source object detection pre-trained model based on the YOLOv4 algorithm is employed for pedestrian detection. Later, the video frame is transformed into a top-down view for distance measurement from the 2D plane. The distance between people can be estimated and any non-compliant pair of people in the display will be indicated with a red frame and red line. The proposed method is validated on a pre-recorded video of pedestrians walking on the street. The proposed technique can be further developed as a detection tool in real-time applications.

Keywords: *Covid-19, Social Distancing, Deep Learning, CNN, YOLOv4, Transfer Learning*

Contents

List of Figures

Nomenclature

CNN	Convolutional Neural Network
DNN	Deep Neural Network
GPU	Graphical Processing Unit
R-CNN	Region Based Convolutional Neural Networks
SSD	Single Shot Detection
YOLO	You Only Look Once

Chapter 1

Introduction

Corona (COVID-19) is a highly infectious disease caused by coronavirus discovered a year ago which affects different people in different ways. This virus was firstly found in Wuhan City of China in December 2019 and then slowly contributed to spreading worldwide. The common symptoms are fever, dry cough, and tiredness. The virus that causes COVID-19 is mainly transmitted through droplets generated when an infected person coughs, sneezes, or exhales. These droplets are too heavy to hang in the air and quickly fall on floors or surfaces. You can be infected by breathing in the virus if you are within close proximity of someone who has COVID-19. Most people who are infected with the SARS-CoV-2 virus have respiratory symptoms. They start to feel a little bit unwell, they will have a fever, they may have a cough or a sore throat or sneeze. People with mild symptoms who are otherwise healthy should manage their symptoms at home. On average it takes 5–6 days from when someone is infected with the virus for symptoms to show, however it can take up to 14 days. Due to the increasing number of cases, many of the countries declared total lockdown to avoid physical contact. Individuals, communities, businesses, and healthcare organizations are all part of a community with their responsibility to mitigate the spread of the Covid-19 disease. In reducing the impact of this coronavirus pandemic, practicing social distancing, keeping a safe distance and self-isolation have been deemed as the most effective ways to break the chain of infections after restarting the economic activities. The world has not yet fully recovered from this pandemic and But it is not always possible to stay indoors and people have to go out for their daily basic necessity therefore the world health organization recommended some guidelines for the prevention of the virus and hence maintaining the norm of social distancing is a necessity and also in our benefit to living a safer and healthier life. To implement social distancing the distance between two individuals must be 6-feet or 2-meters. But it's not possible always in crowded places to maintain a perfect distance. So we will be proposing a model to determine whether or not an individual is following the rule of social distancing. The findings are verified using both live streams as well as video feed by alerting people through mobile notification using a deep learning model.

The following points will summarize the model in brief:

- Developing a deep learning model for detecting if social distance with the minimum distance between persons is being followed or not.
- Using YOLO V4 algorithm for the object detection.

Chapter 2

Problem Statement

Everyone has been affected by the COVID-19 coronavirus pandemic on a global scale. It crippled the economic growth of the entire nation around the world. As of April 2021, the virus reached nearly 152 million infected patients and 3 million died from the virus. To combat the transmission of the virus, there are enforced protocols set by the World Health Organization (WHO) like observing strict social distancing in public places.

Social distancing also called as “physical distancing”, means keeping a safe space between yourself and other people who are not from your household. As the country has started to unlock amid surging COVID-19 cases, maintaining social distancing has become a key issue. The biggest concern revolving around the COVID-19 situation is how quickly the infection spreads from one person to another through contact or even being within proximity of an infected person. Social Distancing is here to stay longer than expected to fight Covid-19.

Nevertheless, the efficiency of this rule in preventing disease transmission in the public has been lessened because of not following the proper protocols. Hence, It is essential to develop a social distancing analyzer model which will provide individual protection and prevent the local epidemic.

Deep learning advancement with the integration of computer vision offers the breakthrough in development in countless fields of technology. The existing work in the field has been implemented by using AI and deep learning technologies. Deep neural networks (DNNs) include object detection, image classification, and image segmentation abilities. Convolutional neural networks (CNNs) are one of the principal models of DNN that is generally used in computer vision tasks. After training the model, CNNs can identify and classify facial images even with minor differences using feature extraction ability and store image pattern details. But to improve the performance of the system, YOLO pre-trained model with transfer learning methodology needs to be implemented.

The primary algorithm, SSD (Single Shot Detection) gives lower accuracy than the YOLO algorithm. The different versions of the YOLO algorithm, v2, v3, v4 subsequently constitute in improving accuracy and efficiency.YOLOv4 algorithm gives better accuracy as compared to other versions. Hence, the proposed system aims at using the latest version, YOLOv4 as an object detection algorithm. Further accuracy can also be improved by calibrating the input image or video frame into a Top-down view instead of processing it directly as fed to the system.

Chapter 3

Review of Literature

Since the rise of COVID-19, various research has been done to control the pandemic, among the various researches social distancing and wearing masks proved to be an efficient way to control the transmission of the virus and these methods have also been suggested by various health organizations like World Health Organization WHO and so every nation opted for this method to prevent the spread of this deadly virus. Even after getting vaccinated against COVID-19, it is mandatory to follow all the safety measures according to WHO. So there has to be some system to instruct and keep a check on the people about following various rules and regulations that are necessary to prevent the spread of the virus. In the past years, there has been significant work done in object detection techniques using deep learning models which are efficient in doing complex tasks and have achieved outstanding performance in computer vision ZHAO [2017]. Deep learning models for detecting a person focus on feature extraction and transfer learning techniques MathWorks, in transfer learning there is a process that involves fine-tuning the pre-trained model and is started with some existing networks like AlexNet, GoogleNet, ResNet, or VGG and new dataset with unknown classes are been feed into those models. In feature Extraction, all the layers are tasked for learning some specific features from images so that these features can be taken at any time from the network during the training process and these features can be given as input to other algorithms. Hou et al. [2020]Human detection can be considered as object detection which is a part of computer vision for the classification and localization of the object in the video. Nguyen et al. [2016]There has been huge growth in research in object recognition and detection in artificial intelligence which has also achieved outstanding results. Nguyen et al. presented a brief analysis of state-of-the-art, recent development and challenges of human detection. The survey mainly focuses on human descriptors, machine learning algorithms, and real-time human class detection. Various techniques using deep convolutional neural networks(CNN) have achieved superior performance on many image recognition algorithms. Hou et al. [2020]A deep convolutional neural network(CNN) is a deep learning algorithm with multilayer perceptron neural networks which contain several convolutional layers,sub-sampling layers, and fully connected layers. Then the weights in the whole layers in the networks are trained for each object classification and detection based on the dataset. CNN has achieved great success in large-scale image classification due to the recent development in high-performance computing systems and the availability of large datasets. Different CNN models for object detection had been proposed in terms of network architecture and algorithms. CNN models such as AlexNet, VGG16, Inception V3, and ResNet-50 are trained to achieve outstanding results in object recognition.

Various research work has been done on social distancing using many different techniques. Berglund designed an app named 'Healthy Together' which is deployed in Utah for tracking the COVID-19 which uses GPS, Bluetooth, and built-in applications in smartphones to track the infected people. In this app, the GPS generates the location maps with a user's time-stamped path and it also includes symptom tracking, resources for nearby COVID-19 testing sites. The user has to download the app and if any user came across the infected person then the app notifies or alerts the user about it. But this technology also had some disadvantages as it won't work if there is no WiFi or signal in their phones.

Some researchers also used drones to capture the video of people to ensure if there have been any violations, that is if there have been any gatherings of people in public areas. Robakowska et al. [2017] used drones with cameras to capture real-time videos and images to track the gatherings of people in public places. Yadav [2020]proposed a system that used raspberry pi4 along with a camera to track public places in real-time to prevent the transmission of the virus. The model was trained with the custom dataset and this dataset was installed in the raspberry pi4 along with the camera. The camera captures the real-time videos or images of the public places to the model which has been fed into the raspberry pi4 which continuously monitors the public places and checks if all the proper rules like wearing masks and keeping a safe distance are followed or not. In this method if a person identified without a mask is detected then his photo is taken and sent to the State Police Headquarters and if a social distance violation is detected an alarm is made to ring which alerts the people to maintain a safe distance.

Hou et al. [2020]proposed a methodology for social distancing detection using deep learning to calculate the distance between people to control the impact of the pandemic. A detection tool was developed to alert people to maintain a safe distance from each other by evaluating a video that is captured in real-time. Object detection pre-trained model based on the YOLOv3 algorithm was used for detecting the pedestrian. After capturing the video frame, the video frame was transformed into a top-down view for calculating the distance from the 2D plane. If the distance between the people is less than the specified distance then it is indicated by red bounding boxes else it is indicated by green bounding boxes if people are at safe distance. This model is slower than other models and the accuracy of calculating the distance between the people can be improved by changing the camera perspective view.

Sahraoui et al. [2020]designed a framework based on deep learning and the internet of vehicles(IoV) to track objects using the Faster R-CNN algorithm by using a switching camera system and detecting the people as well as the vehicles violating the physical distancing. In this thermal cameras are used as they are even visible during nights or unusual weather conditions. When any violations are noticed by any vehicle it sends an alert to the nearby roadside units, that is, advertisement boards or directly by vehicle to vehicle communication.

In a recent paper published by Vinitha.V and Velantina which highlights the use of AI using computer vision and deep learning for social distancing detection systems, they have used OpenCV, Tensorflow library, and YOLOv3 algorithm for detecting the people and showing how many people are at risk by calculating the distance between them. The groups of people who are not maintaining proper distance are computed by Bird's eye view approach. In this approach for converting to Bird's eye view that is top view, Region of Interest(ROI) and distance scale are computed after loading the video frame into the model. Then the pedestrians are detected and bounding boxes are drawn and then results are shown by different colors of bounding boxes.

Ahmed et al. [2021]presented a deep learning-based social distance monitoring framework using overhead perspective using a pre-trained YOLOv3 algorithm for human detection. In this framework, the transfer learning method was adopted to improve the pre-trained model's performance. The model was trained on the overhead dataset and the newly trained layer was appended with the existing model. Then the bounding boxes were used to display the human objects and calculate the distance between the humans using the centroid of the bounding boxes. The distance was calculated using the Euclidean distance formula. The object tracking accuracy of the model was 95%. However different algorithms to track the people can be used to improve the accuracy and the model can be improved for different indoor and outdoor environments.

Elhanashi and Gagliardi [2021] presented an intelligent surveillance system for people tracking and social distancing classification based on thermal images.YOLOv2 algorithm was used to detect and track people from the video frame and the model was tested on Jetson nano (NVIDIA device) which is suitable for real-time applications in different scenarios and is capable to process multiple high-definition video streams. The proposed approach can also be implemented in a distributed video surveillance system.

Keniya and Mehendale [2020]proposed a system that distinguishes the social distancing pattern and classifies them as a violation of social distancing or maintaining the proper distance or not. In this system, they loaded about 295 images from the dataset in which every image had single or multiple labels, and these images dataset was used for training the model. The SocialdistancingNet-19 architecture was used for training the model and the object was marked by a rectangular box. This network consisted of 2 different subnetworks which are feature extraction and feature detection, in this model feature extraction was carried out using a pre-trained convolutional neural network (CNN) model and a YOLO detection model. In this model first, the frame was detected and then the distance was calculated using centroids of the bounding boxes. Then the comparison of the distance between the centroids of two different individuals was done and the safety distance was calculated accordingly. NVIDIA GPU - 1660,1408 Cuda core with 6GB DDR5 RAM and 192 bits memory bus was used to train the network. The accuracy of the SocialdistancingNet-19 was 92.8% and in the results, the bounding boxes were shown along with the labels if the person is following proper distance or not.

Since most of the work was done on either YOLOv2 or YOLOv3 pre-trained model for detecting and tracking the person in the video frame, we will be using YOLOv4 pre-trained model for detecting and tracking the person object into the video frame which is faster and efficient as compared to other models.

Chapter 4

Existing System

Elhanashi and Gagliardi [2021] presented a methodology to develop a workflow for monitoring social distancing using thermal images. Thermal imaging is the process of converting infrared radiation into visible images which depict the spatial distribution of temperature differences in a scene that is captured by a thermal camera. In this method firstly, the thermal images or video which are captured using a thermal camera is prepared for applying to the model. Then the algorithm for object detection to detect the people is applied to these thermal images or video frames and it checks the number of people who are present in the images or video frame and the distance between the centroid of the bounding boxes of the detected people is calculated and the algorithm then decides whether the people are following the social distance and whether they are safe or not based on the number of persons and the measured distance between the centroid of bounding boxes. Elhanashi and Gagliardi [2021] The steps involved for people detection and social distancing classification on thermal images is as shown below,

In this method, Deep Neural Network(DNN) application is used in MATLAB to construct YOLOv2 neural network layers and then the DNN is applied to NVIDIA Jetson Nano. The CNN is built with 29 layers, it includes the input layer, middle layers, and subnetwork of YOLOv2 layers. The algorithm was trained on two different datasets of thermal images, Of this dataset, 70% was used for training, 20% for validation, and 10% for testing. As a result, the trained algorithm showed great performance for people detection as compared to other algorithms like R-CNN and

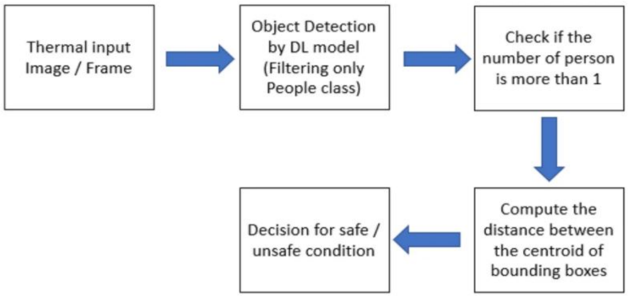

Figure 4.1: Steps for people detection and social distancing detection Elhanashi and Gagliardi [2021]

Fast R-CNN. This approach also used comparatively less memory size when compared to other algorithms for object detection. But the speed and accuracy can further be increased by using other algorithms like YOLOv3 or YOLOv4.

Keniya and Mehendale [2020] proposed a framework to distinguish the social distancing pattern and classify them if they are violating the social distancing rules or not. In this approach, they used 295 images from a dataset that had labels in order to use them for training the model, 60% dataset was used for training, 10% for validation, and 30% for testing. This method used the SocialdistancingNet-19 architecture for training the model and a rectangular box was used to highlight the person object. This network had 2 subnetworks which are feature extraction and feature detection. The feature extraction was done on a pre-trained convolutional neural network(CNN) model and the detection of sub-networks of small CNN is compared to feature extraction, composed of a few convolutional layers which are specific to the YOLO object detection model. Once the video frame is loaded into the YOLO algorithm the euclidean distance is calculated and is then used to calculate the distance between the centroids of the object which are detected into the frame. In this approach, data augmentation was done in order to improve the accuracy by randomly transforming the data while training, it also added a variety of data during training the model. In order to train the network, the NVIDIA GPU-1660,1408 Cuda core with 6GB DDR5 Ram and 192 bits memory bus was used. The accuracy of the model SocialdistancingNet-19 was 92.8% and it also achieved the proper detection of people according to the distance between the pair of the people. However, when capturing the video through the webcam it has to be necessary that the people keep on moving continuously in order to detect the distance and people correctly. The accuracy of the model can also be improved by using some other algorithm or by changing the camera perspective.

Chapter 5

Proposed System

The proposed system focuses on identifying the persons from the image or video frame who have been given as an input to the model to check whether the people are following proper distance between them or not and check how many people are at risk if they are not following the proper distance with the help of deep learning algorithm by using OpenCV library and YOLOv4 algorithm.

Steps for checking the social distancing among people are as follows,

1. Input the image/video frame
2. Detecting the people(humans) in the frame using YOLOv4
3. Calculating the distance between every person who has been detected into the frame
4. Displaying the results by using different color bounding boxes among the people.

Object detection simply refers to identifying objects in digital photographs and object localization means identifying the location of one or more objects in an image or video frame and drawing the bounding boxes around that particular object. In this method, both the methods, object detection, and object localization will be used. Since there are various models available for detecting and tracking the object front of the video frame like R-CNN, Fast R-NN, Faster R-CNN but YOLO model family is considered to be much faster than the R-CNN.

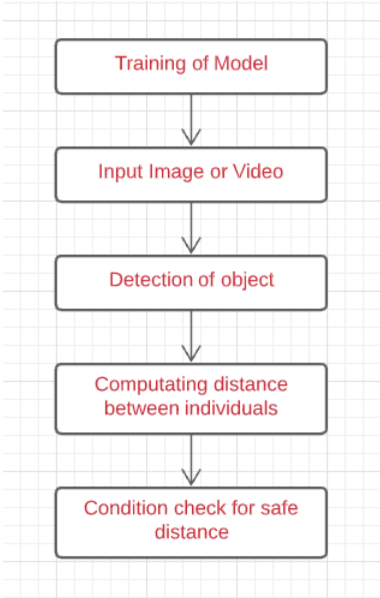

Figure 5.1: Data Flow Diagram for Social Distance analyzer model

5.0.1 Object detection and tracking using YOLO algorithm

The YOLO(You Only Look Once) is the most effective real-time object detection algorithm based on regression and it predicts the classes and bounding boxes for the whole image in one run of the algorithm. The algorithm will first decide to predict the person's class and will show the bounding boxes specifying the object location. The bounding boxes are described using 4 features which are,

1. Center of the box(x,y)
2. Width(w)
3. Height(h)
4. Value c corresponding to that particular class of the object.

Since the YOLO algorithm is trained on the COCO dataset[15] the value of c will be zero. Since the class of the person object in the COCO dataset is at the 0th position. There are different versions of YOLO algorithms like YOLO, Fast YOLO, YOLOv2, Tiny YOLO, YOLOv3, YOLOv3, YOLOv3-tiny, YOLOv4. We will be using YOLOv4 for detecting the people and tracking them. Why

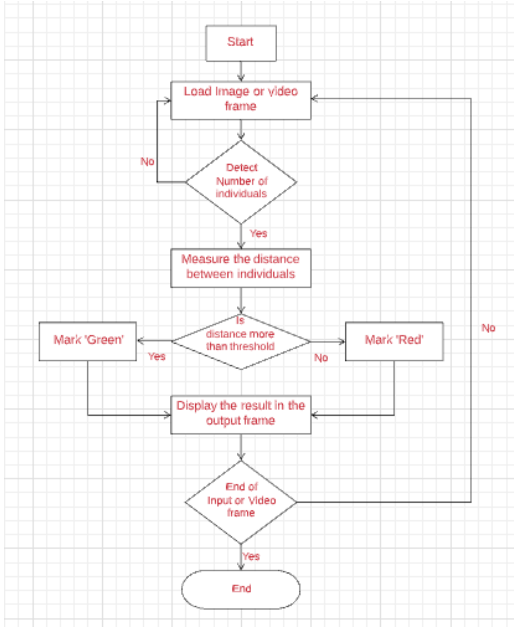

Figure 5.2: Flow Chart for the social distance analyzer

YOLOv4? YOLOv4 is built on YOLOv3.[16]The architecture of the neural network of YOLOv4 has increased the mean Average Precision(mAP) by 10% and the number of Frame per Second(FPS) by 12% and it has also become easier to train this neural network on a single GPU.

5.0.2 Distance Calculation

To calculate the distance we use the Euclidean distance which will calculate the distance between the centroids of the two bounding boxes. The centroid, C(x,y) of the bounding boxes is calculated as follows,

$C(X,Y) = \frac{X_{\min}+X_{\max}}{2}, \frac{Y_{\min}+Y_{\max}}{2}$

Xmin and Xmax are the minimum and maximum values for the corresponding width of the bounding box and Ymin and Ymax are the minimum and maximum value of the corresponding height of the bounding box To calculate the distance between the two centroids(bounding boxes) C1 and C2 we use the Euclidean distance which is calculated using the following formula,

$D(C_1, C_2) = \sqrt{(X_{\max} - X_{\min})^2 + (Y_{\max} - Y_{\min})^2}$

5.0.3 Dispalying Results

After getting the distance we will be comparing the distance between every bounding box and if the distance is more than the threshold value then the bounding box will be of green color and if it is less than the threshold value then the bounding box will be in red color.

Chapter 6

Architectural Overview

In this paper, a deep learning model is introduced for social distance monitoring. The general overview of the work is divided into two modules, person detection, and social distance monitoring.

1. Person Detection Module: The input images are processed to the human detection module; the human detection module utilizes deep learning YOLOv4 architecture and detects human bounding boxes. The bounding box information is further utilized by a social distance monitoring module that identifies the violations and further processes them to the surveillance unit.

   The architecture is additionally trained for top view human detection. The recorded human data set is divided into training and testing samples. The detection algorithm detects humans in the image and initializes its color to green.

2. Social Distance Monitoring: In this step, after human detection, the center point, also called the centroid, is calculated utilizing the information of detected bounding box coordinates. The centroid, C(x,y) of the bounding boxes is calculated as follows,

$$C(X,Y) = \frac{X_{\min} + X_{\max}}{2}, \frac{Y_{\min} + Y_{\max}}{2} \tag{6.1}$$

   Xmin and Xmax are the minimum and maximum values for the corresponding width of the bounding box and Ymin and Ymax are the minimum and

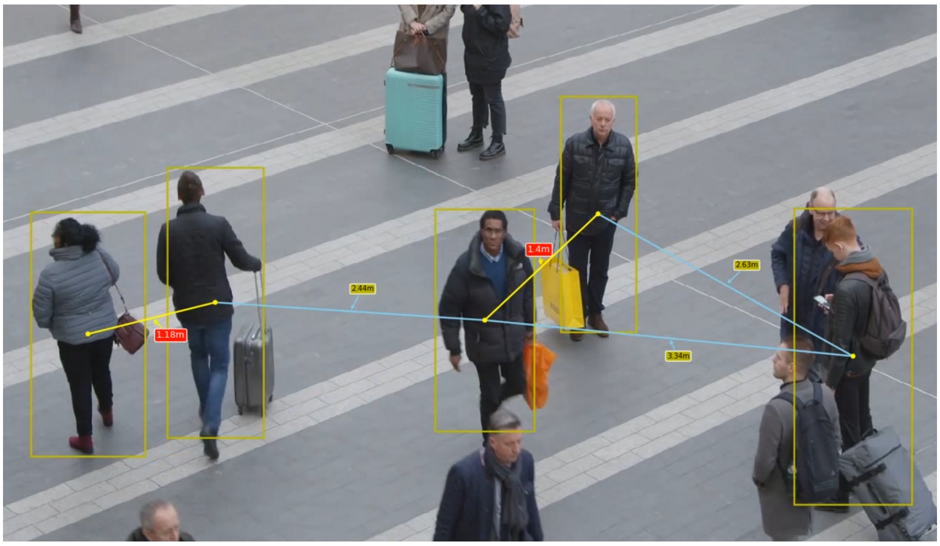

Figure 6.1: Functioning of Person Detection and Social Distance Monitoring Module Diaz [July 20, 2020]

maximum value of the corresponding height of the bounding box.

The distance between each center point of the bounding box is computed using Euclidean distance. To calculate the distance between the two centroids(bounding boxes) C1 and C2 we use the Euclidean distance which is calculated using the following formula,

$$D(C_1, C_2) = \sqrt{(X_{\max} - X_{\min})^2 + (Y_{\max} - Y_{\min})^2}, \tag{6.2}$$

A distance-based threshold is defined to identify whether people in the top view scene make a social distance violation or not, utilizing the estimated centroid distance information. This value is used to check whether any two or more people are at smaller distances than the defined T pixels threshold. The threshold T is applied using the below equation:

$$V = 1, T \geq D; V = 0, otherwise. \tag{6.3}$$

The threshold value T estimates the social distance violation V, in terms of the image pixels. The people do not maintain or violate the social distance if the estimated distance between people is less than the defined threshold (number of pixels); in such a scenario, the information of the bounding box is collected in the violation list. The detected bounding box is initialized as green color by detection architecture. The stored information is verified with a violation list; if the information exists and the detected bounding box is too close, the color is changed to red. The developed framework displays the total number of violations caused by people and detects people bounding boxes along with centroid at the output.

The Transfer Learning approach is adopted to increase the performance and efficiency of the human detection module. For training and testing purposes, a sufficient number of sample images are utilized. The training and testing samples are randomly split at a predefined ratio. The detection architecture is further trained using images of the top view data set. An additional layer is produced after training and is combined with a pre-trained architecture. The detection architecture is again tested for similar test images, as presented in the previous section.

6.0.1 Overview of YOLOv4

YOLO v4 Overview:
For human detection, a deep learning paradigm is applied, as different kinds of object detection algorithms are available. Due to the best performance, in this project, we use YOLOv4. The architecture applied here is a one-stage detector network to estimate object class probabilities and bounding box. The architecture is previously trained on the COCO dataset.

There are two types of object detection models, one stage or two-stage models. A one-stage model is capable of detecting objects without the need for a preliminary step. On the contrary, a two-stage detector uses a preliminary stage where regions of importance are detected and then classified to see if an object has been detected in these areas. The advantage of a one-stage detector is the speed it is able to make predictions quickly allowing real-time use. YOLOv4 is an important improvement of YOLOv3, the implementation of new architecture in the Backbone and the modifications in the Neck have improved the mAP(mean

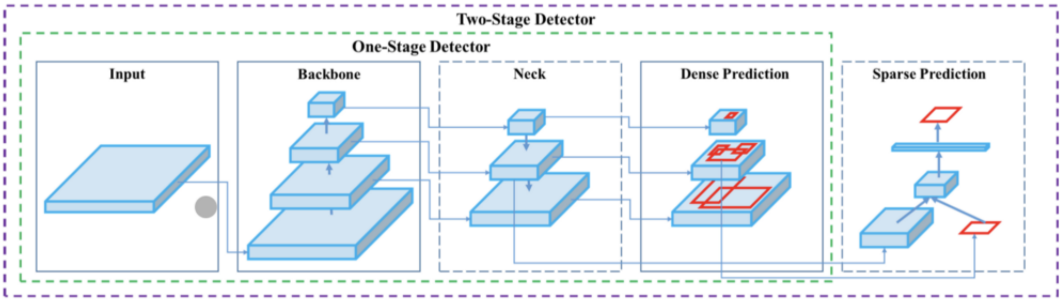

Figure 6.2: Object Detection Architecture RUGERY [Sep 7, 2020]

Average Precision) by 10% and the number of FPS(Frame per Second) by 12%. In addition, it has become easier to train this neural network on a single GPU.

1. Backbone
   The main objective of the backbone is to extract the essential features. Often pre-trained neural networks are used to train the backbone. The YOLOv4 backbone architecture is composed of three parts:
   Bag of freebies: Bag of freebies methods is the set of methods that only increase the cost of training or change the training strategy while leaving the cost of inference low.
   Bag of specials: A bag of special methods is the set of methods that increase inference cost by a small amount but can significantly improve the accuracy of object detection.
   CSPDarknet53: The Cross Stage Partial architecture is derived from the DenseNet architecture which uses the previous input and concatenates it with the current input before moving into the dense layer.

2. Neck
   The essential role of the neck is to collect feature maps from different stages. Usually, a neck is composed of several bottom-up paths and several top-down paths.

3. Head
   The role of the head in the case of a one-stage detector is to perform dense prediction. The dense prediction is the final prediction which is composed of

a vector containing the coordinates of the predicted bounding box (center, height, width), the confidence score of the prediction, and the label.

YOLO v4 consists of:

- Backbone: CSPDarknet53
- Neck: SPP, PAN
- Head: YOLOv3

Chapter 7

Implementation Details

The implementation of the proposed system focuses on developing a Social Distancing Analyser Model which aims at monitoring if social distancing protocols are respected and accordingly show the violations. Implementation of the model is done by using Deep Learning Approach. OpenCV library in python and YOLOv4 algorithms are the base technologies used for implementing object detection.

Steps involved in the Implementation of the Proposed System includes the following 6 steps,

1. Environment Setup
2. Taking the Input
3. Detecting and Tracking the objects
4. Calculating the distance
5. Displaying the violating cases
6. Deploying the model

7.0.1 Environment Setup

Since the YOLOv4 requires GPU, we will be using The Google Colaboratory.

The Following things are required to set up before training the model.

- Enabling the GPU runtime

  It is done by navigating to the Edit section of Colab. In the Notebook Settings, Set Hardware Accelerator to ”GPU”

- Connecting Google Drive with Google Colab

- Cloning the official Darknet(Neural network framework) repository

- Verifying Cuda Version

- Building darknet folder

7.0.2 Taking the Input

Inorder to take the inputs, the following steps are followed

- Downloading the YOLO v4 weights

- Creating user defined functions to upload and download image

7.0.3 Detecting and Tracking the objects

After taking the inputs we need to test the image to check if its detecting the objects or not.

- Testing model on the sample data

- Testing model on the user uploaded image and videos

7.0.4 Distance Calculation

- For every person detected in the input frame, centroid is calculated by using Centroid formula

$$C(X,Y) = \frac{X_{\min} + X_{\max}}{2}, \frac{Y_{\min} + Y_{\max}}{2} \tag{7.1}$$

Figure 7.1: User Specified video(screenshot) input to Model

Figure 7.2: Output video(screenshot) obtained after Processing

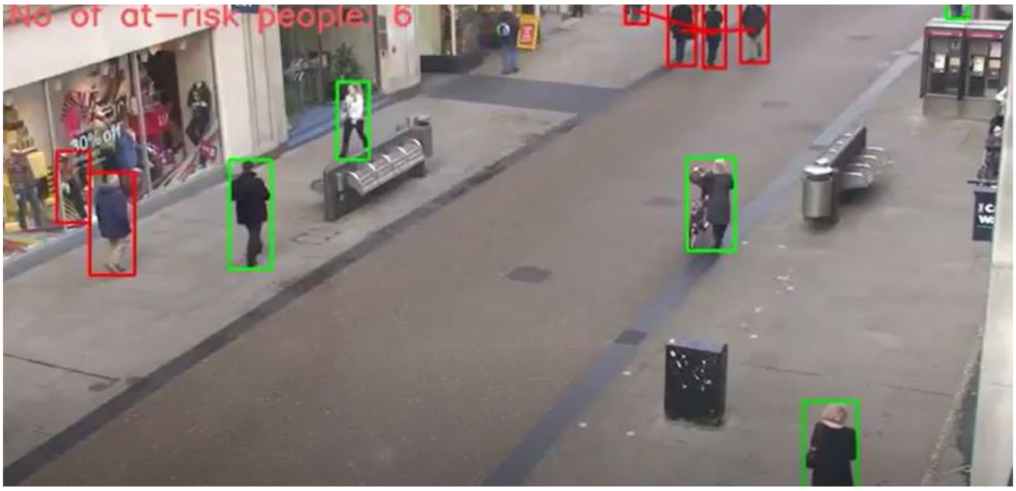

Figure 7.3: Detecting the violating cases

- Euclidean Distance formula is applied to calculate distance between all the possible pairs of centroids

$$D(C_1, C_2) = \sqrt{(X_{\max} - X_{\min})^2 + (Y_{\max} - Y_{\min})^2}, \tag{7.2}$$

- Calculated distance for pairs is stored in the list
- A minimum threshold distance is set(6 feet or 2 meter)
- Each element in the list(distance) is checked against threshold value

7.0.5 Output

Checking for Violating Cases:

- Initially, every person detected is represented by GREEN bounding box.
- Color of bounding box is updated to RED if the distance is less than threshold value
- Thus, Green and red color boxes indicates safe distance and violation of social distancing respectively.

7.0.6 Deployment

- Well trained model is deployed using ngrok application
- ngrok is cross-platform application that will expose a local development server to the Internet
- Model is executed in local colab server
- Local port will be tunneled with public port using ngrok and anyone with the port number(HTTP link) can access the application.

ngrok

ngrok is a free tool that allows us to tunnel from a public URL to our application running locally. ngrok is built in Go, so it is packaged as binaries for each major platform.

To install ngrok do the following:

- Download the package for the system
- Unzip the package
- (Only for windows) configure ngrok

Once ngrok is installed,we can use it to tunnel to an application running on, say, port 3000 is as easy as: http://3000.ngrok.io/

When we start up ngrok it shows the URL which we are tunnelling through and another local URL, eg. http://127.0.0.1:4040. By opening that url one can access the localhost from other device as well.

The dashboard allows us to see the status of the tunnel, the requests made through it and the responses that came back from your application. Great for debugging misfiring webhook endpoints.These tunnels can also be secured with password.

The final deployed website will look as shown below,

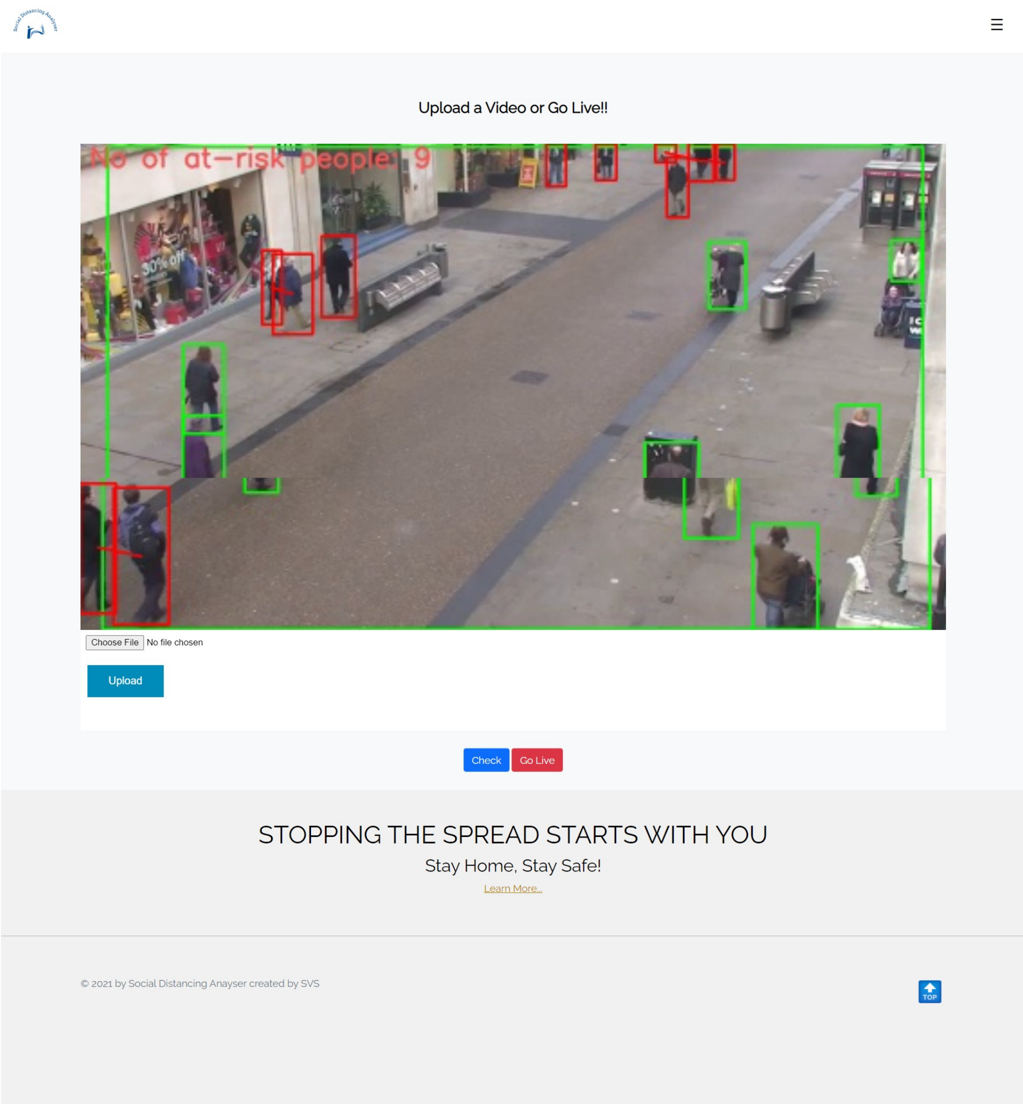

Figure 7.4: Deployed Website

Chapter 8

Testing and Practical Difficulties

8.0.1 Testing Difficulties

Difficulties faced during testing of model

1. YOLOv4 requires GPU, lack of GPU based system so need to switch to online GPU available which is Google Colaboratory
2. Limited(Usuage Limits) resource access in google colab
3. Internet issues
4. Training model required much time in spite of strong network connection. Hence, turning off device or any internet fluctuations result in process abortion.
5. Mounting drive failed when tried to access in more than one devices at same time

8.0.2 Practical Difficulties

Practical difficulties faced during implementation of model

1. Converting to tflite models in order to switch to android device
2. Need for camera based device to start live stream

3. While testing for the Social distancing on input video or live stream frame, people must be moving to detect violations. If people are at steady state then objects will be detected but model will have less accuracy for evaluating violations.

4. Need to have Android system or for PC needs most recent OS

5. Needs access to camera and internal issues

6. Strong network connection recommended

Chapter 9

Future Scope

This method works efficiently for detecting the proper distance among the people and changes the color of the bounding boxes as a green color for people who are safe and red color for people who are at risk. Hence it can help to control the crowd in public places, eventually controlling the spread of the virus.
The work can further be extended to detect if people are also wearing masks along with maintaining the proper distance. This model can also be used in real-time by capturing the video through the CCTV cameras and can be used to alert the people on the spot by displaying it to the nearby units like hoardings or bulletin boards or by announcing on the speaker. Further work can be done if the vehicles are following the proper rules like wearing the masks inside the cars or if their windows are properly closed or not which will help to stop the transmission of the virus.

Chapter 10

Conclusion

In this method, a solution based on a deep learning approach to check the distance between two individuals is presented using the front view of the camera. The pre-trained YOLOv4 algorithm is used for detecting the persons into the video frame and the distance between the people is calculated using the Euclidean formula by taking into consideration the centroid of the bounding boxes around the person object. The speed and accuracy are improved by using the YOLOv4 as compared to the YOLOv3 algorithm for object detection and tracking. Since the increasing spread of the COVID-19 virus, this method can be adopted by the government officials as well as by some private sectors to stop the transmission by avoiding the overcrowded areas.

Chapter 11

Application

This work may be improved in the future for indoor as well as outdoor activities. Various tracking, as well as detection algorithms, might be used to track people violating social distancing. Also, The algorithm was then implemented for live video streams and images also. This system can be used in CCTV for surveillance of people during pandemics. Mass screening is possible and hence can be used in crowded places like railway stations, bus stops, markets, streets, mall entrances, schools, colleges, etc.

By monitoring the distance between two individuals, we can make sure that an individual is maintaining social distancing in the right way which will enable us to curb the virus. For example in most crowded places like Vegetable Market, people who are crowding around specific vendors and not following the protocol of social distancing the vendor's license can be canceled in the charge of breaking the protocol and those people can also be fined for the same by the government.

Appendix A

Formulae

Centroid : $C(X,Y) = \frac{X_{\min}+X_{\max}}{2}, \frac{Y_{\min}+Y_{\max}}{2}$
Euclidean distance: $D(C_1,C_2) = \sqrt{(X_{\max}-X_{\min})^2 + (Y_{\max}-Y_{\min})^2}$

References

Imran Ahmed, Misbah Ahmad, Joel J.P.C. Rodrigues, Gwanggil Jeon, and Sadia Din. A deep learning-based social distance monitoring framework for covid-19. *Sustainable Cities and Society*, 65:102571, 2021. ISSN 2210-6707. doi: https://doi.org/10.1016/j.scs.2020.102571. URL `https://www.sciencedirect.com/science/article/pii/S2210670720307897`.

Jennifer Berglund. Tracking covid-19: There's an app for that. *IEEE Pulse*. URL `https://www.embs.org/pulse/articles/tracking-covid-19-theres-an-app-for-that/`.

Alex Diaz. Measuring social distance in the time of covid-19. *Towards Data Science*, July 20, 2020.

Saponara Elhanashi and Gagliardi. A. implementing a real-time, ai-based, people detection and social distancing measuring system for covid-19. *J Real-Time Image Proc*, 2021. doi: https://doi.org/10.1007/s11554-021-01070-6. URL `https://link.springer.com/article/10.1007%2Fs11554-021-01070-6`.

Yew Cheong Hou, Mohd Zafri Baharuddin, Salman Yussof, and Sumayyah Dzulkifly. Social distancing detection with deep learning model. In *2020 8th International Conference on Information Technology and Multimedia (ICIMU)*, pages 334–338, 2020. doi: 10.1109/ICIMU49871.2020.9243478.

Rinkal Keniya and Ninad Mehendale. Real-time social distancing detector using socialdistancingnet-19 deep learning network. 2020. doi: http://dx.doi.org/10.2139/ssrn.3669311. URL `SSRN:https://ssrn.com/abstract=3669311`.

MathWorks. What is deep learning? how it works, techniques and applications. Available at `https://in.mathworks.com/discovery/deep-learning.html`.

Duc Thanh Nguyen, Wanqing Li, and Philip O. Ogunbona. Human detection from images and videos: A survey. *Pattern Recognition*, 51:148–175, 2016. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2015.08.027. URL `https://www.sciencedirect.com/science/article/pii/S0031320315003179`.

Marlena Robakowska, Anna Tyranska-Fobke, Joanna Nowak, Daniel Slezak, Przemyslaw Zuratynski, Piotr Robakowski, Klaudiusz Nadolny, and Jerzy Robert Ładny. "the use of drones during mass events". *"Disaster and Emergency Medicine Journal"*, 2(3):129 – 134, 2017. ISSN 2543-5957. doi: 10.5603/DEMJ.2017.0028. URL `https://journals.viamedica.pl/disaster_and_emergency_medicine/article/view/DEMJ.2017.0028`.

Pierrick RUGERY. Explanation of yolo v4 a one stage detector. *becoming human.ai*, Sep 7, 2020.

Yesin Sahraoui, Chaker Abdelaziz Kerrache, Ahmed Korichi, Boubakr Nour, Asma Adnane, and Rasheed Hussain. Deepdist: A deep-learning-based iov framework for real-time objects and distance violation detection. *IEEE Internet of Things Magazine*, 3(3):30–34, 2020. doi: 10.1109/IOTM.0001.2000116.

Vinitha.V and Velantina. Social distancing detection system with artificial intelligence using computer vision and deep learning. *International Research Journal of Engineering and Technology (IRJET).* URL `https://www.irjet.net/archives/V7/i8/IRJET-V7I8698.pdf`.

WHO. Advice for the public on covid-19 – world health organization. Available at `https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public`.

Shashi Yadav. Deep learning based safe social distancing and face mask detection in public areas for covid-19 safety guidelines adherence. *International Journal for Research in Applied Science and Engineering Technology*, 8:1368–1375, 07 2020. doi: 10.22214/ijraset.2020.30560.

ZHAO. Object detection with deep learning: A review. *IEEE*, 2017.

Acknowledgement

We have great pleasure to express our gratitude to all those who have contributed and motivated us during project work. This project cannot be completed without the effort and cooperation from our group members. We also sincerely thank our project guide, Kumud Wasnik for the guidance and encouragement for this project. Last but not the least we would like to express our gratitude towards our friends, family and respondents for the support.

Date:08/06/2021

Snehal Salunke,
Vaishnavi Sawalkar,
Sakshi Sawant