DSN2099

Project Exhibition - II

Team - 16

MACHINE LEARNING

DE-SOCIALIZE

SOCIAL DISTANCING TOOL

REVIEWS

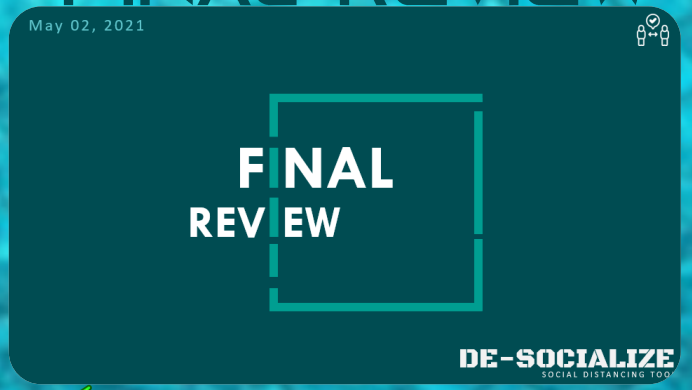
ZEROTH REVIEW

Completed on March 04, 2021

FIRST REVIEW

Completed on May 02, 2021

FINAL REVIEW



Completed on May 02, 2021

March 04, 2021

ZEROTH REVIEW

DE-SOCIALIZE

SOCIAL DISTANCING TOOL

TEAM 16

MEMBERS

Aakash Mattoo	19BAI10152
Abhinav Kumar Singh	19BAI10078
Sachin Sharma	19BAI10082
Piyush Saraf	19BAI10041

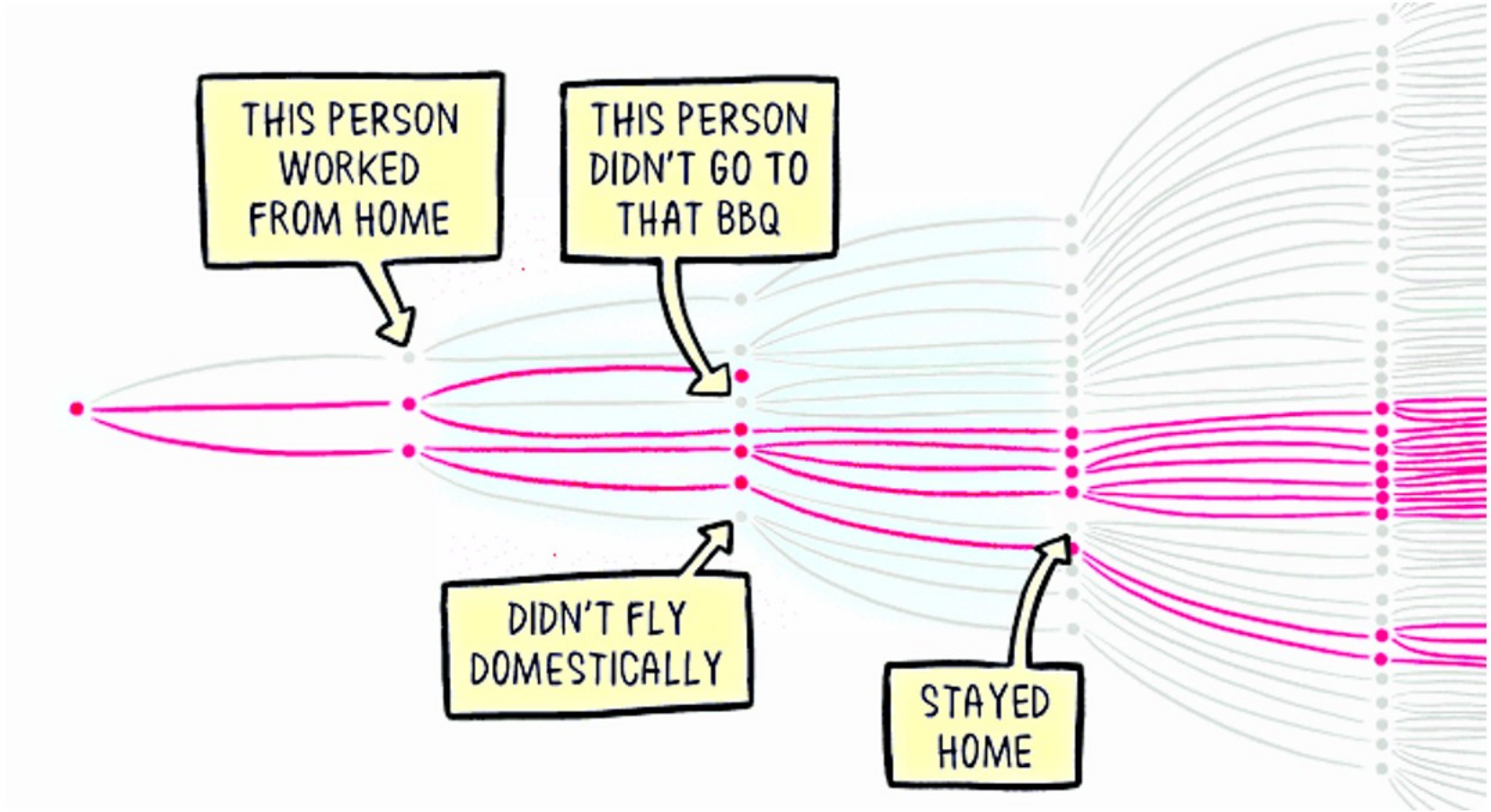
GUIDE

Dr. Ashish Kumar Sahu
(100148)

INTRODUCTION

The rampant coronavirus disease 2019 (COVID-19) has brought global crisis with its deadly spread to more than 180 countries. WHO declared this a pandemic. This situation forces the global community to look for 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.

It aims at reducing the physical contact between possibly infected individuals and healthy persons. WHO prescribed that people should maintain at least 6 feet of distance among each other in order to follow social distancing.


THIS PERSON WORKED FROM HOME
THIS PERSON DIDN'T GO TO THAT BBQ
DIDN'T FLY DOMESTICALLY
STAYED HOME

NEWS 18 INDIA

With No Social Distancing at Protests, Farmer Leaders Say New Laws Bigger Threat Than Coronavirus

There had been few signs of social distancing as thousands of farmers from Punjab and Haryana began their protest last week against the three new agro-marketing laws, setting off on a march to the national capital.

- PTI
- SONIPAT
- LAST UPDATED: NOVEMBER 30, 2020, 17:07 IST
- FOLLOW US ON: Facebook Twitter Instagram Telegram Google News

THE NEW INDIAN EXPRESS

No masks, no social distancing: COVID-19 norms go for a toss at BJP MLA's wedding

Viral videos of the wedding showed state BJP chief Chandrakant Patil and former chief minister Devendra Fadnavis intearcting with people at close quarters without wearing masks.

Covid-19 in Delhi: Social distancing not on track yet

Sidharatha Roy & Paras Singh | TNN | Updated: Mar 21, 2020, 11:27 IST

A- A+

NEW DELHI: The stringent advisory given by Delhi Metro Rail Corporation (DMRC) on Thursday to prevent spread of Covid-19 in the mass transit system remained only on paper a day later.

Coronavirus in India latest updates

YES, IT IS NECESSARY!

The Tribune

VOICE OF THE PEOPLE

No social distancing here

Residents throng congested markets in city without masks

THE ECONOMIC TIMES

Thousands protest in London against India's farming reforms

No masks, sanitisers or social distancing; Bhopal seems to have thrown Covid caution to the winds

EXISTING WORK

- Since the novel coronavirus pandemic began, many countries have been taking the help of technology based solutions in different capacities to contain the outbreak.

- Existing Social Distancing Analyzer was a human-intervention system, where various appointed persons would run through all the recordings of CCTV surveillance and then find out, using manual analytics, that in which areas of the state is the social distancing norm most violated.

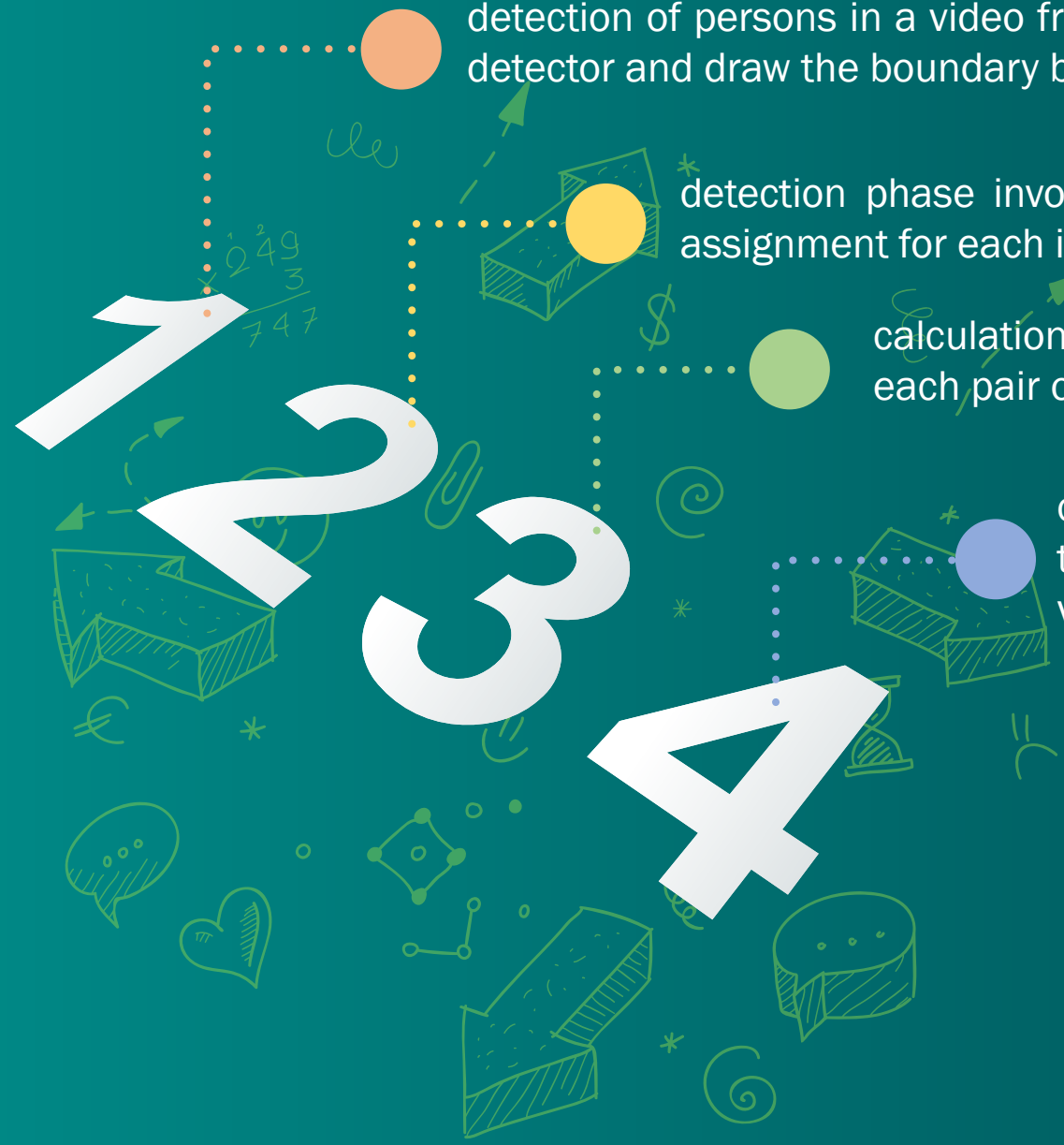
- And then, they would take necessary actions with the help of corresponding authorities to stop the violation of the norm in the red-zoned areas.

- Normally, we humans have a tendency to react differently when dealing with knowns and unknowns. We tend to be liberal with our knowns thus taking the norm for granted. But, a model will not have this tendency and therefore, it will treat everyone as same.

LIMITATIONS

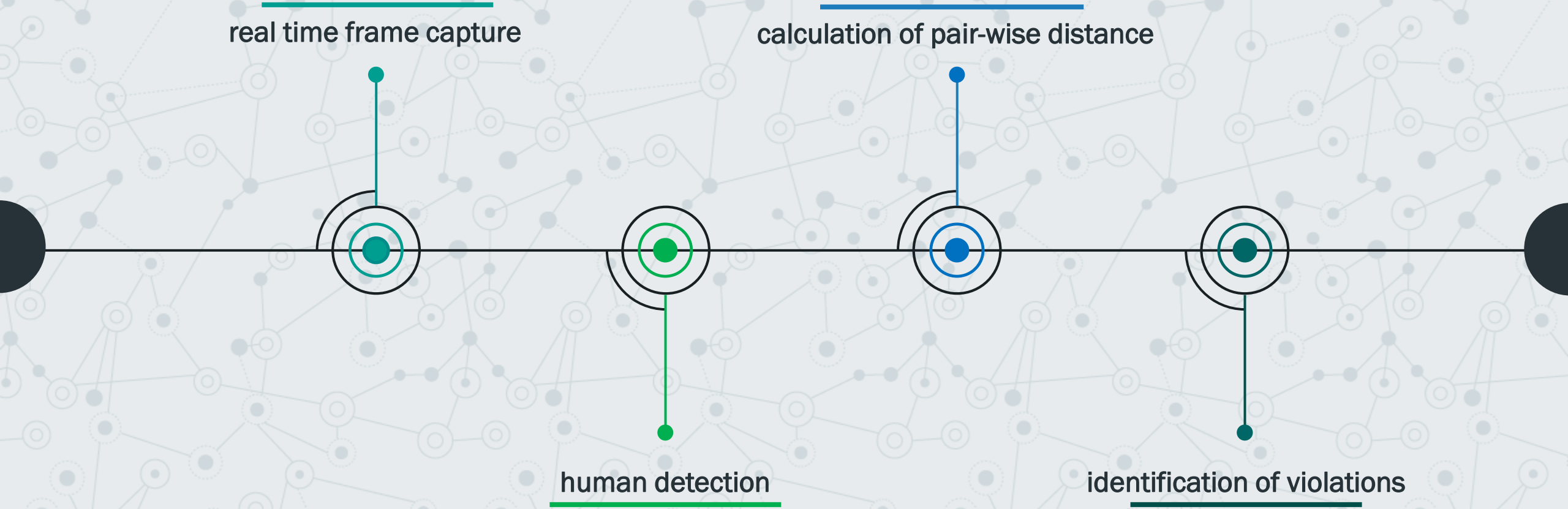
- There are several non commercial social distancing analyzer, but the Object detector used in these are slow and inaccurate.
- Human intervention would cause biasing in judging and therefore create problems.
- Humans, either doing something repetitive, would then focus on something creative else. Machines never get tired of doing the same job repetitively.
- The authorities warning them to maintain distancing would create a negative impact in the people's mind. Instead, if an alarm is raised, there wouldn't be any negative emotional sentiment.
- Sometimes, due to human inefficiency, some areas might be unchecked. Machines would not do such mistakes.

PROPOSED WORK



1. detection of persons in a video frame with help of the object detector and draw the boundary boxes
2. detection phase involves people tracking and ID assignment for each individual.
3. calculation of the Euclidean distance for each pair of persons in an image/frame
4. comparison of distances with threshold to identify and violation of social distancing

PROPOSED WORK



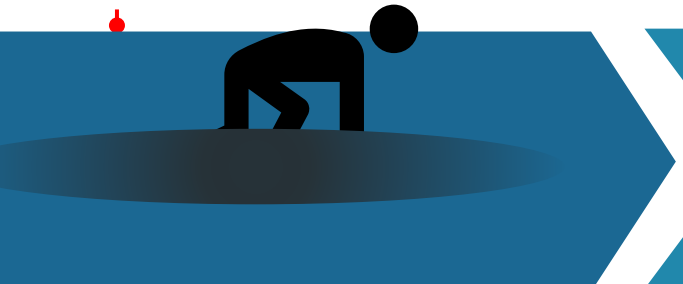
METHODOLOGY

METHODOLOGY

1. We need to collect the dataset, various images of persons

2. After Gathering all the required data we have to do the annotation of the images using **Labelimg**

3. XML files will be created for the annotation of each image.



4. Then we need to divide the dataset into train and test folders. We have to choose the model for training, There are various state of the art DNN models available for Object detection like **RCNN, Faster RCNN, SSD** and different versions of **YOLO.SO**
We have to choose the best.

About YOLO

5. For using **YOLOv3** we have to download the pretrained weight, its **config file** and the **.name** file (contains the details of trained classes).

6. After that we have to feed each frame of our input video stream/web cam to our model .

7. We have to extract only person class from the model. This can be done by using **'personIdx'** function

8. Then we have to extract the co-ordinates of the boundary boxes drawn over persons in a frame. This can be implemented using **numpy.array().**

METHODOLOGY

9 Then calculate the centroids of the boundary boxes of persons. This will be done by **numpy method.**

10 Now we have to calculate the Euclidian Distance between each centroids pairs in a frame. For this we have to use Distance metric method contained in the **scipy.spatial** module.

11 After that we have to compare each calculated distance with a threshold pixel and if less then make the boundary box Red otherwise make its color to green

12 We have to make a counter which will increase whenever calculated distance will be less then the threshold.

13 Print counter value on screen for each frame. This will represent the number of violations.

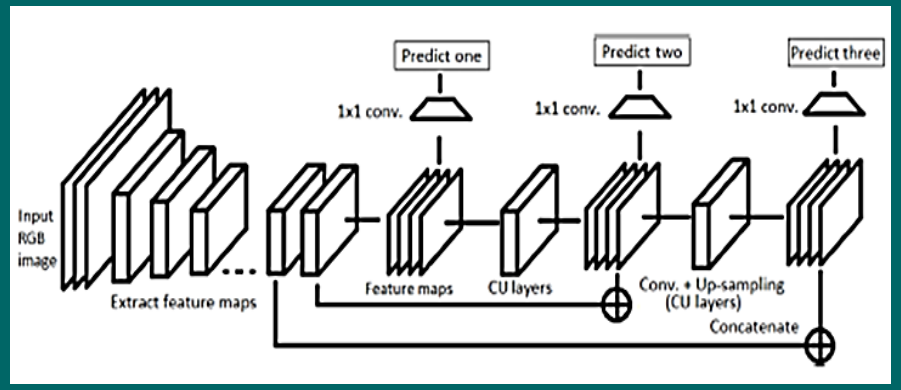
14 If input is video stream then Combine all frames in a video stream and write it to the memory using **OpenCV**.

15 If input video stream is from the webcam, then display each of the processed frames in a real time

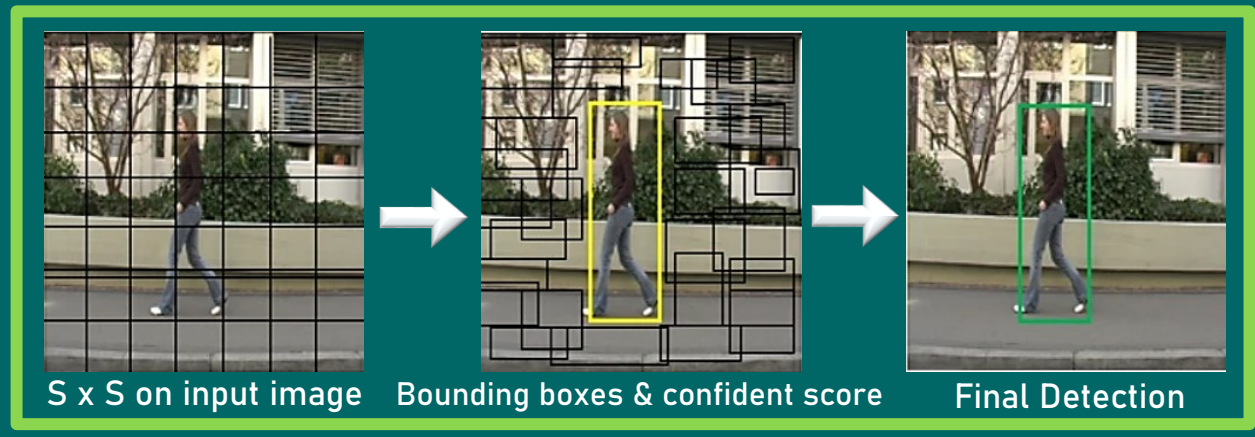

YOLO

YOU ONLY LOOK ONCE

- As the name suggests, YOLO 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. Instead of selecting the interesting regions from the image, YOLO takes the problem of object detection as the regression problem where the object detection and classification take place in a single neural network.
- This network is extremely fast, it processes images in real-time at 45 frames per second. A smaller version of the network, Fast YOLO, processes an astounding 155 frames per second.



Schematic representation of YOLO v3 architecture



S x S on input image | Bounding boxes & confident score | Final Detection

- From the figure it is clear that YOLOv4 offers the best trade-off for the speed and the accuracy for a multi-class object detection purpose. however we will use YOLOv3 model in our project as it has perfect balance between processing time and accuracy.
- YOLO Is pretrained on more than 80 classes in which person is also there so we don't need to train our model.

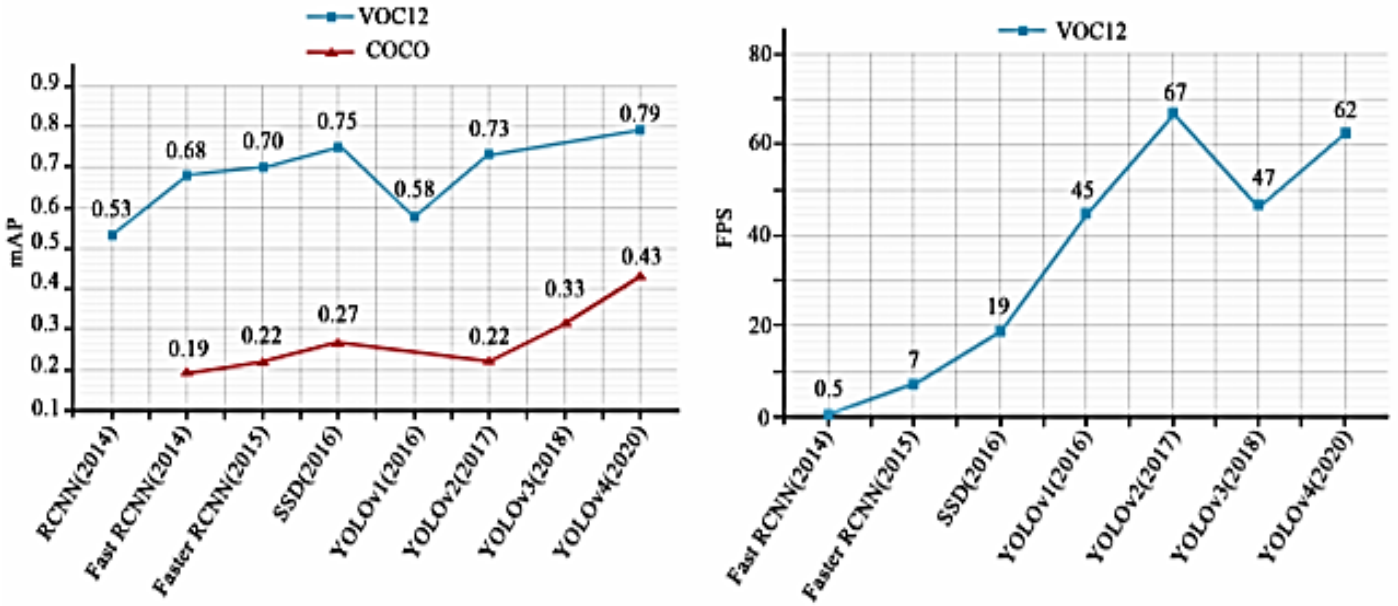
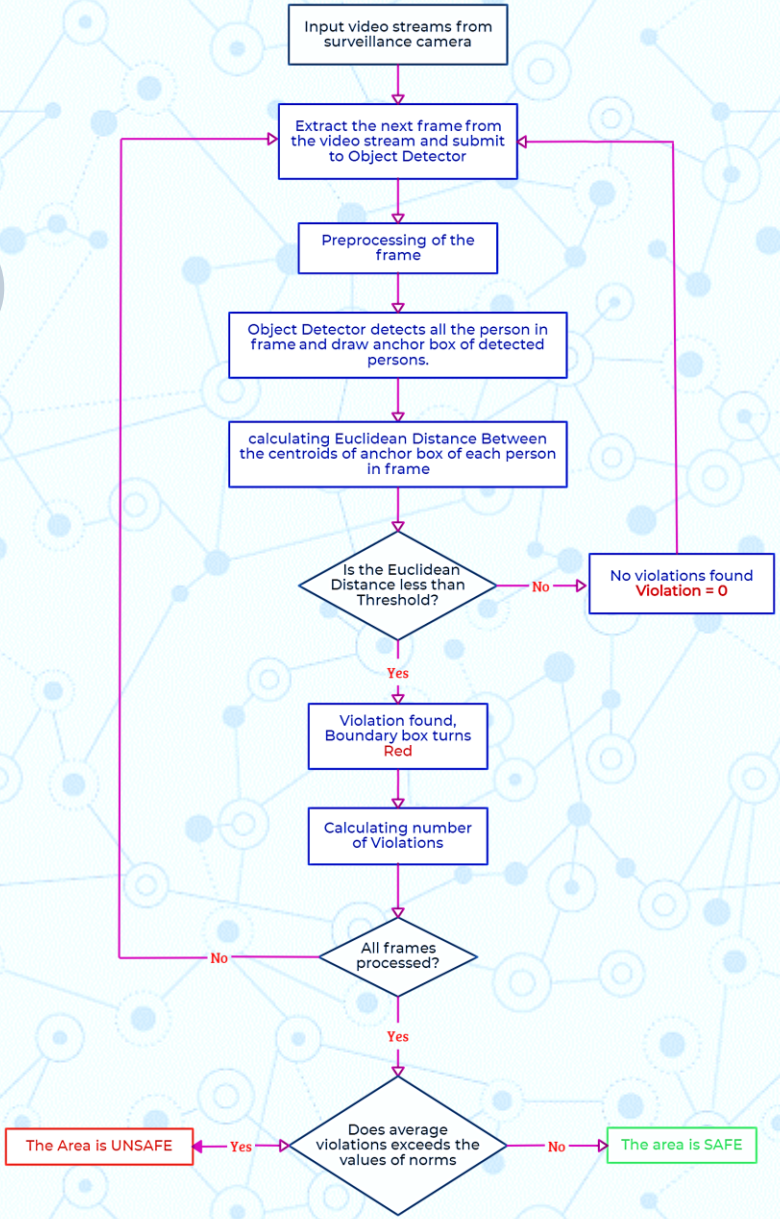


Fig:- Mean Average Precision (mAP) and Speed (FPS) overview of eight most popular object detection models on Microsoft Common Objects in Context (MS-COCO) and PASCAL Visual Object Classes (VOC) datasets.

SYSTEM ARCHITECTURE

conceptual model that defines the structure, behavior, and more views of a system


Input video streams from surveillance camera
Extract the next frame from the video stream and submit to Object Detector
Preprocessing of the frame
Object Detector detects all the person in frame and draw anchor box of detected persons.
calculating Euclidean Distance Between the centroids of anchor box of each person in frame
Is the Euclidean Distance less than Threshold?
No
No violations found
Violation = 0
Yes
Violation found, Boundary box turns Red
Calculating number of Violations
All frames processed?
No
Yes
Does average violations exceeds the values of norms
Yes
The Area is UNSAFE
No
The area is SAFE

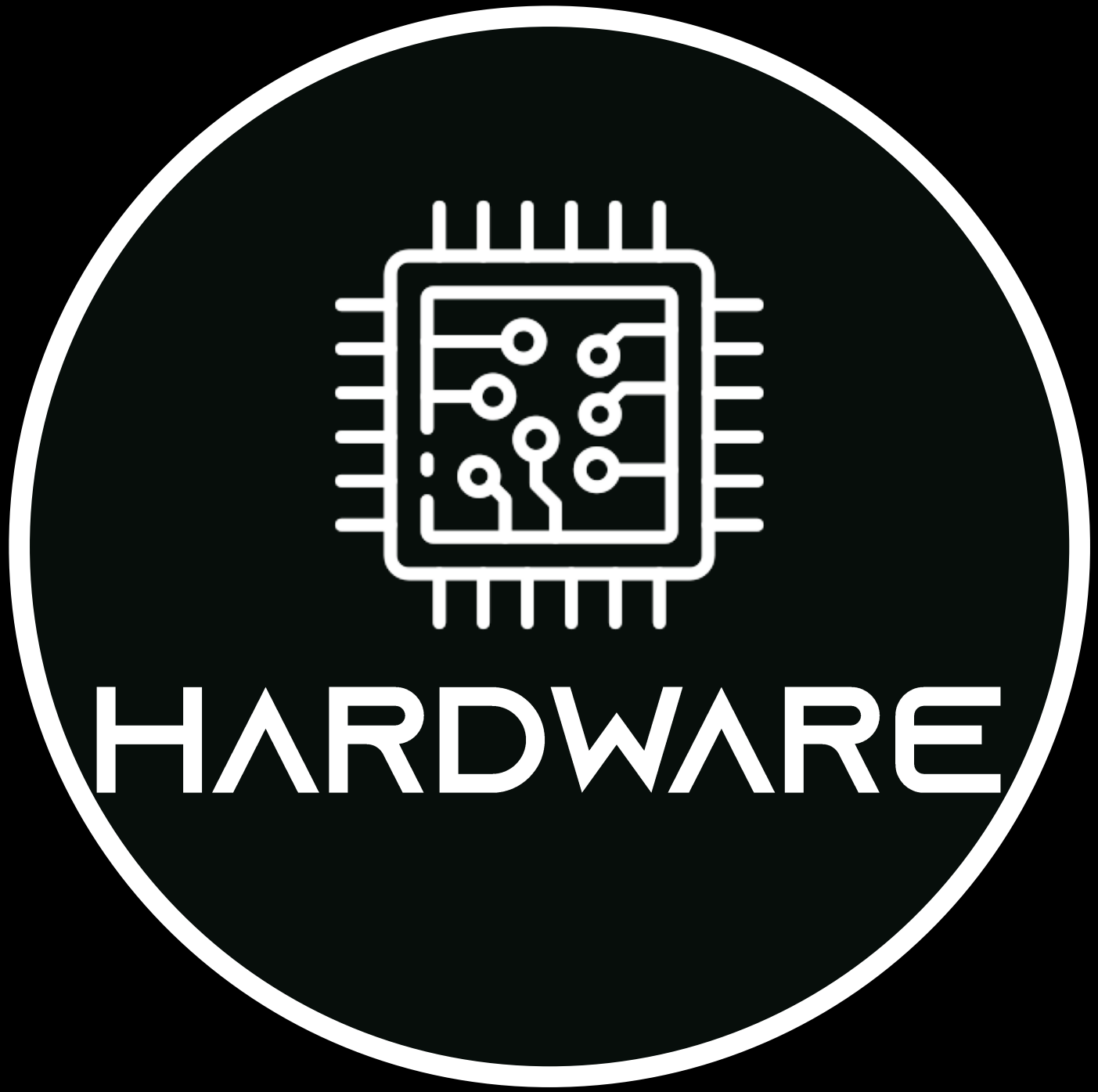
NOVELTY OF THE PROJECT

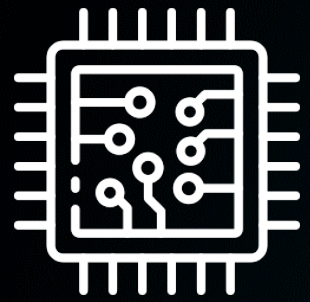
- This tool can be incorporated into existing surveillance camera systems in various public places.
- There is an alert that sounds when people are too near to each other (less than 6 feet).
- The software that is used to screen individuals' activities by means of a security system should must have a evident ethical and privacy implications else the information linked with this system is vulnerable to get exploit.

REAL TIME USAGE

- It can potentially detect where each person is in the input frame in real-time, and return a bounding box that turns red if the distance between two people is dangerously close.
- This can be used by governments to analyze the movement of people over the public places and alert them if the situation turns serious.
- It can be used to contain the spread of the virus among people by alerting as soon as the distance between the people crosses a threshold value.
- This software can be incorporated in camera system of various public places, for example: mall, railway station, streets, etc.

SYSTEM REQUIREMENTS

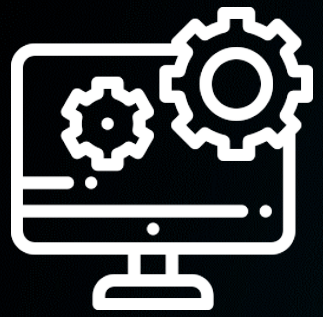

HARDWARE



HARDWARE
requirements

- CCTV surveillance cameras spread across important and crowded streets for getting input data.
- Interconnecting wires or network, bringing live streamed data from CCTV to the model.
- Alarms on each CCTV for raising alert.
- High power computation unit GPU like Geforce GTX 1080 (4GB).
- Minimum RAM – 4GB DDR5 CORE : i5 minimum 1.6 GHz or faster processor.


SOFTWARE



SOFTWARE
requirements

- Software Needed :
  OpenCV 3 | PIL | Python3 | Darknet | Numpy
- A model to process the input data and give out necessary signal (Red or Green zone)
- A database for storing the area location and camera number to identify the coordinates of red zone.

May 02, 2021

FIRST REVIEW

DE-SOCIALIZE

SOCIAL DISTANCING TOOL

LITERATURE REVIEW

Monitoring Covid-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques : *Narinder Singh Punn, Sanjay Kumar Sonbhadra, Sonali Agarwal*

- This paper discusses about the COVID -19 situation. It involves maximum topic of our project and is the main paper reference for our project.

Monitoring Social Distancing under various low light conditions with deep learning and a single motionless time of flight camera : *Adina Rahim, Ayesha Maqbool, Tauseef Rana*

- This also is an important paper and discusses about camera capturing in different lighting conditions specially in low lighting conditions. It also highlights about the proposed framework of YOLOv4.

Motivations for Social Distancing as Complementary Measures to Combat the COVID-19 Pandemic: Quantitative Survey Study : *Kai Kasper*

- This paper discusses about the current developments in several countries show that this measure can be technologically accompanied by mobile apps; meanwhile, privacy concerns are being intensively discussed.

You Only Look Once: Unified, Real-Time Object Detection : *Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi*

- From this paper we learnt about YOLO, a new approach to object detection and how it outperforms other detection methods, including DPM and R-CNN, when generalizing from natural images to other domains like artwork.

Robust Real-Time Object Detection : *Paul Viola, Michael J. Jones*

- This paper describes a visual object detection framework that is capable of processing images extremely rapidly while achieving high detection rates. It also tells how the system yields face detection performance comparable to the best previous systems.

Optical Flow-Based Person Tracking by Multiple Cameras : *Hideki Tsutsui, Jun Miura, and Yoshiaki Shira*

- This paper describes optical flow-based person tracking using multiple cameras in an indoor environment. There are usually several objects in indoor environments which may obstruct a camera view and how we can overcome them.

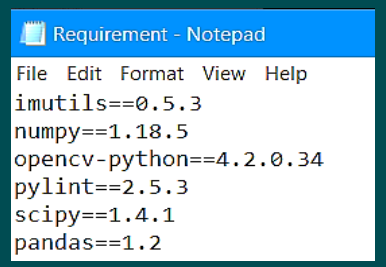
MODULES

1. Preparing Workspace
2. Person Detection (YOLO)
3. Data Computation
4. Evaluating Model

Module Workflow

Preparing Workspace

- Creating a folder in any location named ***Social Distancing Analyzer.***
- Creating an Environment in Anaconda with Python (v. 3.6.9) in the above directory.
- Installing required libraries by using pip installer
  - Run `pip install -r Requirement.txt` in cmd.



- Download all the necessary files required like yolo.config, coco.name and the pretrained yolo3.weight files.

  https://pjreddie.com/media/files/yolov3.weights ←for pretained weight !

  - Use the command git clone github.com/AlexeyAB/darknet.

Person Detection

- Extract and pre-process the frames from the input video stream.

```
vs = cv2.VideoCapture(args["input"]
if args["input"] else 0)
while True: (grabbed, frame) = vs.read() #for extracting each frame
```

- Next we will load the pretrained yolov3 object detector.

```
weightsPath = os.path.sep.join([config.MODEL_PATH, "yolov3.weights"])
configPath = os.path.sep.join([config.MODEL_PATH, "yolov3.cfg"])
net = cv2.dnn.readNetFromDarknet(configPath, weightsPath)
```

- We Only need The person class to be detected.

```
LabelsPath = os.path.sep.join([config.MODEL_PATH, "coco.names"])
LABELS = open(labelsPath).read().strip().split("\n")
LABELS.index("person")
```

- Now Each frame is to feed into the pretrained yolov3 object detector which will output boundary boxes around person in each frame with their confidence score


Camera 01


02-06-2019 Wed 09:05:13
person: 0.9614
person: 0.2778
person: 0.9730
person: 0.9781
person: 0.9727
Camera


02-06-2019 Wed 09:05:11
person: 0.3229
person: 0.9834
Camera 01

Data Computation

- The Output of YOLO will be centroids coordinates(x,y), confidence score of prediction and width, height of the boundary box for each prediction.
- For confidence Score less than 0.3 we will ignore the detection with help of if statement .
- We will calculate the top Left and Bottom Right co-ordinates of boundary box for further Computation.

```
x = int(centerX - (width / 2)) y = int(centerY - (height / 2)).
```

- We will Calculate each pair of centroid Distance in each frame.
- Methods For Calculating Distances are –

Minkowski Euclidean Hamming

We will choose Euclidean Distance because of its simplicity.

```
centroids = np.array([r[2] for r in results]) D =
dist.cdist(centroids, centroids, metric="euclidean")
```

Data Computation

- After That *D* is compared with the threshold value. This value will vary based on the angle view of camera.

  Webcam : Threshold = 200-300

  CCTV : in range 40-50px.

- We will increase a created counter in *D<Threshold*

```
for i in range(0, D.shape[0]):
   for j in range(i+1, D.shape[1]):
       if D[i, j] < config.MIN_DISTANCE:
       violate.add(i)
       violate.add(j)
```

- We will Assign red color boundary to violations and green to non violations.

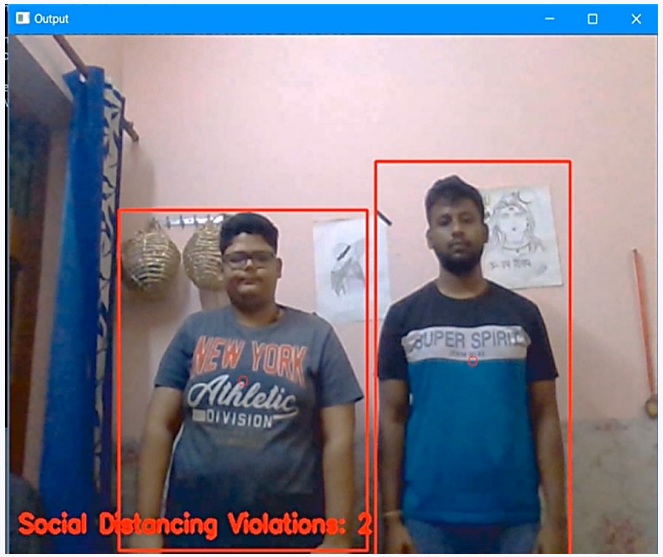
```
 if i in violate: color = (0, 0, 255)
```

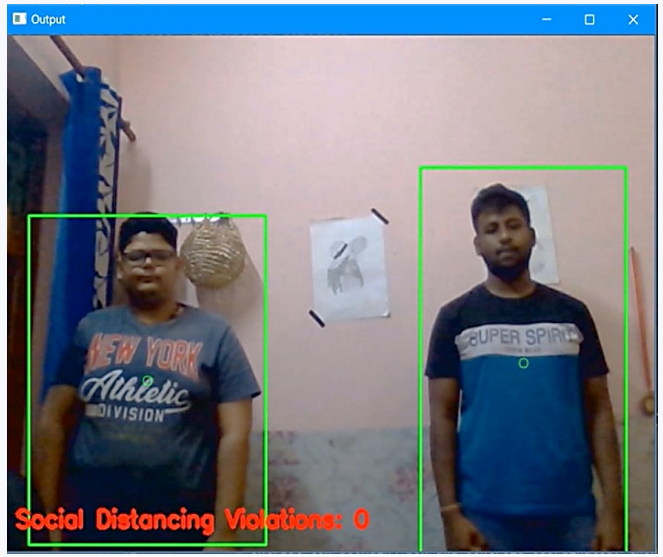
- We will print no. of violations for each frame

```
text = "Social Distancing Violations: {}".format(len(violate))
```

Evaluating Model

In order to evaluate the model, we will test it upon two persons. This file is used for evaluating the model via webcam.





COMMENTS MAY LIE

CODING

CODES NEVER LIE

Social Distancing Analyzer

File Home Share View

Pin to Quick access | Copy | Paste | Cut | Copy path | Paste shortcut — Clipboard

Move to | Copy to | Delete | Rename — Organize

New folder | New item | Easy access — New

Properties | Open | Edit | History — Open

Select all | Select none | Invert selection — Select

This PC > Desktop > Social Distancing Analyzer

Search Social Distancing Analyzer

- Quick access
- OneDrive
- This PC
- Network


config


detection


social_distancing_detector


coco


yolov3


yolov3

6 items

Type here to search

ENG 15:05 10-04-2021

config.py

```
# base path to YOLO directory
MODEL_PATH = "yolo-coco"

# initialize minimum probability to filter weak detections along with the
# threshold when applying non-maxim suppression
MIN_CONF = 0.3
NMS_THRESH = 0.3

# for using gpu
USE_GPU = True

# define the minimum safe distance (in pixels) that two people can be from each other
MIN_DISTANCE = 50
```

detection.py

```
# imports
from .config import NMS_THRESH
from .config import MIN_CONF
import numpy as np
import cv2

# function to detect people
def detect_people(frame, net, ln, personIdx=0):
    # grab dimensions of the frame and initialize the list of results
    (H, W) = frame.shape[:2]
    results = []

    # construct a blob from the input frame and then perfrom a forward pass
    # of the YOLO object detector, giving us the bounding boxes and
    # associated probabilities
    blob = cv2.dnn.blobFromImage(frame, 1 / 255.0, (416, 416), swapRB=True, crop=False)
    net.setInput(blob)
    layerOutputs = net.forward(ln)

    # initialize lists of detected bounding boxes, centroids, and confidence
    boxes = []
    centroids = []
    confidences = []

    # loop over each of the layer outputs
    for output in layerOutputs:
        # loop over each of the detections
```

social_distancing_detector.py

```
# imports
from configs import config
from configs.detection import detect_people
from scipy.spatial import distance as dist
import numpy as np
import argparse
import imutils
import cv2
import os

# construct the argument parser and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--input", type=str, default="", help="path to (optional) input video file")
ap.add_argument("-o", "--output", type=str, default="", help="path to (optional) output video file")
ap.add_argument("-d", "--display", type=int, default=1, help="whether or not output frame should be dis
args = vars(ap.parse_args())

# load the COCO class labels the YOLO model was trained on
labelsPath = os.path.sep.join([config.MODEL_PATH, "coco.names"])
LABELS = open(labelsPath).read().strip().split("\n")

# derive the paths to the YOLO weights and model configuration
weightsPath = os.path.sep.join([config.MODEL_PATH, "yolov3.weights"])
configPath = os.path.sep.join([config.MODEL_PATH, "yolov3.cfg"])

# load the YOLO object detector trained on COCO dataset (80 classes)
print("[INFO] loading YOLO from disk...")
```

Demo & Video Implementation


IMPLEMENTATION
via WEBCAM video


IMPLEMENTATION
via C.C.T.V. video

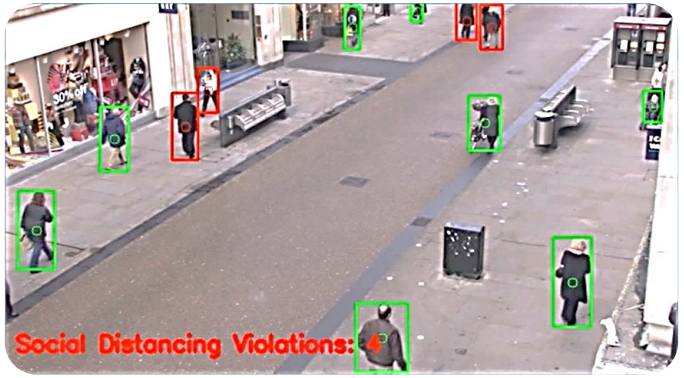
IMPLEMENTATION

via WEBCAM video

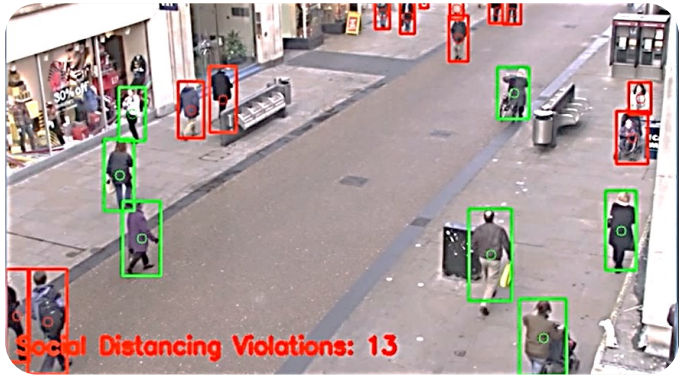
IMPLEMENTATION

via C.C.T.V. video

SNAPshots


Social Distancing Violations:


Social Distancing Violations:


Social Distancing Violations: 13

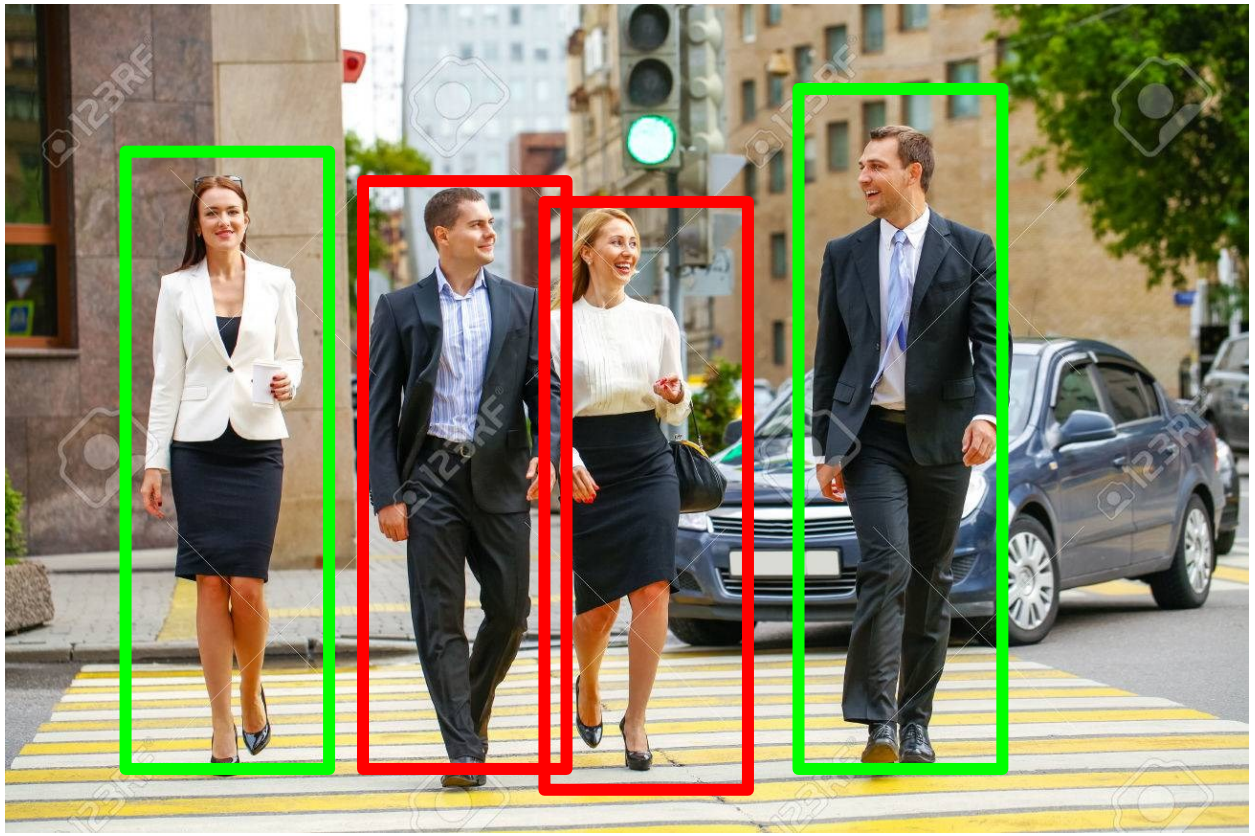
May 02, 2021

FINAL REVIEW

DE-SOCIALIZE

SOCIAL DISTANCING TOOL

The model is fed with a video and thereafter it is processed frame by frame. The objects identified as people are confined in bounding boxes.

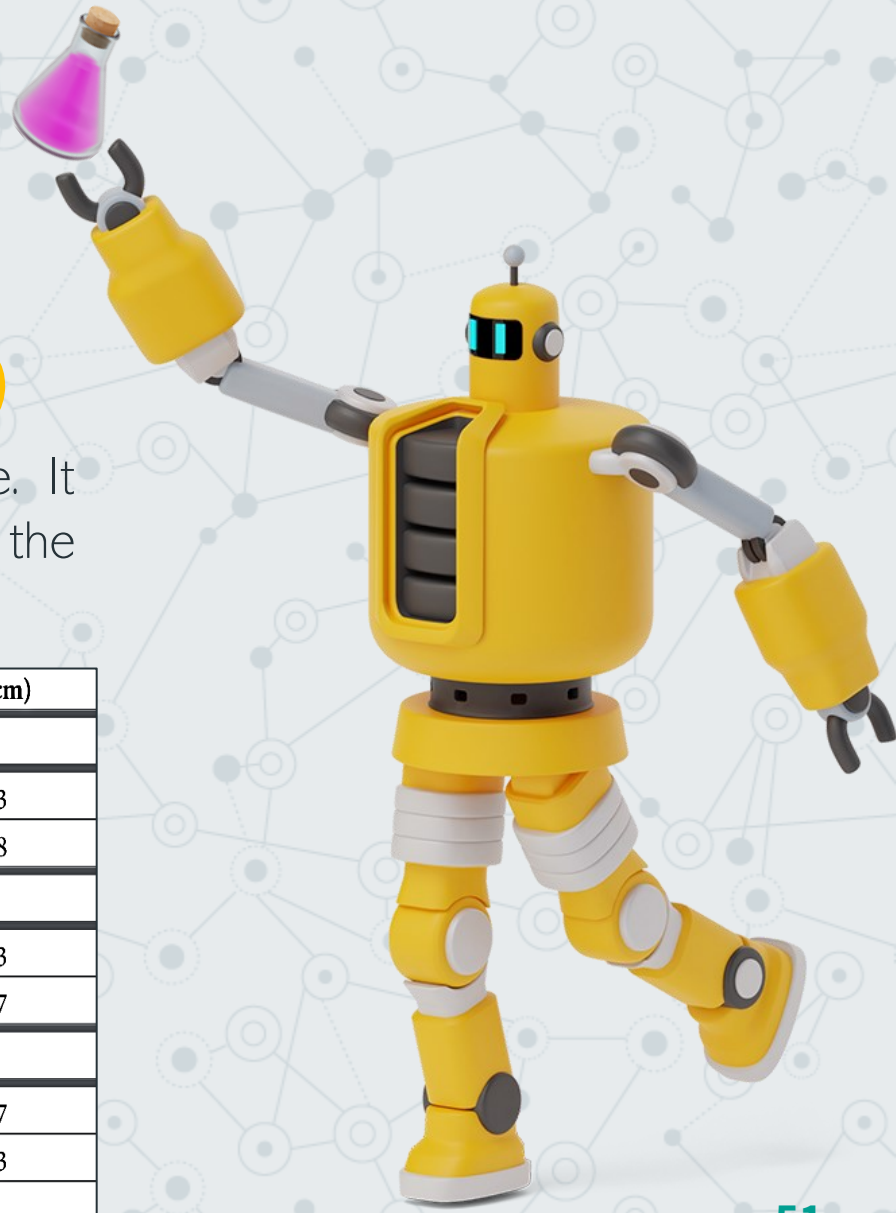


RESULTS

Experimental RESULTS

Our Project uses deep learning to monitor social distancing in real-time. It monitors the same using object detection and tracking approach with the assistance of bounding boxes.

Test	C_d	k	PD (pixels)	Du (cm)	Ad (cm)	AE (cm)
D_{T1T2} = 100 cm	400 cm	0.34236	292.098	-	-	-
1	-	-	153.0	52.3	52	0.3
2	-	-	414.1	141.8	140	1.8
D_{T1T2} = 100 cm	500 cm	0.40635	246.099	-	-	-
3	-	-	131.1	53.3	52	1.3
4	-	-	340.3	138.3	140	-1.7
D_{T1T2} = 100 cm	600 cm	0.49022	203.994	-	-	-
5	-	-	107.5	52.7	52	0.7
6	-	-	285.0	139.7	140	-0.3
Mean Absolute Error (MAE) = 1.01 cm						




(a)
100.0cm
(b)
52.3cm
(c)
141.8cm
(d)
100.0cm
(e)
53.3cm
(f)
138.3cm
(g)
100.0cm
(h)
52.7cm
(i)
139.7cm

Conclusion

- Our Project uses deep learning to monitor social distancing in real-time.
- It monitors the same using object detection and tracking approach with the assistance of bounding boxes.
- Thereafter the cluster of peoples are identified and checked whether the threshold distance between them is maintained or not aided by the bounding boxes again.
- Take into consideration that this approach is highly sensitive to the spatial location of the camera, to fine tune adjust the corresponding field of view.

limitations

We are mapping the actual distance in feet into pixel distance via hit and trial method. No Algorithm is Used

Our Model has depth Error for front view i.e. If a person in frame is behind another person but far from him/her still our model detect it as a violation.

For different scenarios (CCTV ,web cam view) different threshold is required but we are changing the threshold manually.

We are using yolo which theoretically generates good FPS but still only 10-12 frames are processed per second.

recommendations
for future work

- Using proper camera calibration is the first step in developing our social distancing detector
- For accurate mapping we can use pixel calibration algorithms in our model.
- For depth Error we can use depth Estimation camera or we can train our model with deep neural networks.*
- We can automatically estimate the threshold for different scenarios with the help of calibration algorithms
- We can use YOLO v4 for faster executions for the video stream.

Thanks!

Project Exhibition SUCCESSFULLY completed.

DEDICATED TEAM 16

Aakash Mattoo	19BAI10152
Abhinav Kumar Singh	19BAI10078
Sachin Sharma	19BAI10082
Piyush Saraf	19BAI10041

PROJECT GUIDE

Dr. Ashish Kumar Sahu (100148)

Oceans of gratitude to SIR, who has been a constant support for the completion of the project and guided us wherever needed.

Thank you for your efforts and guidance