IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

Social Distancing Detection

Chhaya Bangadkar[1], Faroog Jafri[2], Mohsina Sheikh[3], Ummul Memon[4], Ashwini Meshram[5], Prof. Nitesh Hatwar[6]

[1, 2, 3, 4, 5] Student, Department of Computer Engineering (CE), MIET College, (Bhandara), India
[6]Assistent Professor, Department of Computer Engineering (CE), MIET College, Bhandara), India

Abstract: COVID-19 disease caused by a severe respiratory syndrome coronavirus. It was identified in December 2019 in Wuhan city, China. It's resulted in an ongoing pandemic that caused infected cases including many deaths. The rampant coronavirus disease 2019 (COVID-19) has brought global crisis with its deadly spread to several countries. Due to large population and lesser availability of vaccines, to maintain social distancing is that the only one feasible approach to fight against this pandemic situation. To implement Social distancing, group activities such as functions, social gatherings, travelling, workshops and even praying of people are banned by government during quarantine period. Social distancing, also called physical distancing is the only effective measures taken to prevent the spread of contagious disease by maintaining physical distance between people and reducing the number of times people acquire close contact with one another. Today`s unfortunate circumstances due of spread COVID-19, keeping distance among people is crucial. The goal is to detect people using Haar cascade classifier and find the space between people to check whether a norm social distance of 50 inch is maintained by people. Hence, this work aims at minimizing the spread of the COVID-19 virus by evaluating if and the way persons go with social distancing rules.

Keywords: Covid-19, Social distancing, Python, OpenCV, Deep neural network**.**

I. INTRODUCTION

The current outbreak of novel coronavirus also called as COVID-19 as declared as a public health emergency by the WHO where million people have been affected by the disease with over many of us deaths till date. Social distancing has become a mantra in front of the world. The pandemic situation has taken over the world and has made the conditions worst, as of now there's no vaccination developed for the contagious disease and hence social distancing has emerged mutually of the most effective methods to stop the spread of COVID-19. This manner of living has been forced upon us by the fastest growing pandemic the world has ever seen COVID-19. Keeping a secure distance from one another is the final word thanks to prevent the spread of this disease. With the assistance of computer vision and deep learning we will keep a track on humans and compute the gap between them in pixels by using computer distance algorithms and set the standard maintained distance. The World Health Organization (WHO) states that "COVID-19 is spread via droplets, fomites during close unprecautious contact between an infector and infectee. A fomite is an object or material which is probably going to hold infection, like clothes, utensils, and furniture. Therefore, transmission of the infection will be avoided by staying away with safe distance from other people moreover from touching infected fomites." The virus that causes COVID-19 is presently spreading simply from person-to-person. Once a healthy person comes into contact with respiratory droplets from coughs or sneezes of associate infected person, they are can catch the infection. So, the sole effective measure available to regulate the virus and protect public health is to reduce the frequency of close contact among people. This can be used by governments to monitor the movement of people and alert them if the situation turns serious.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

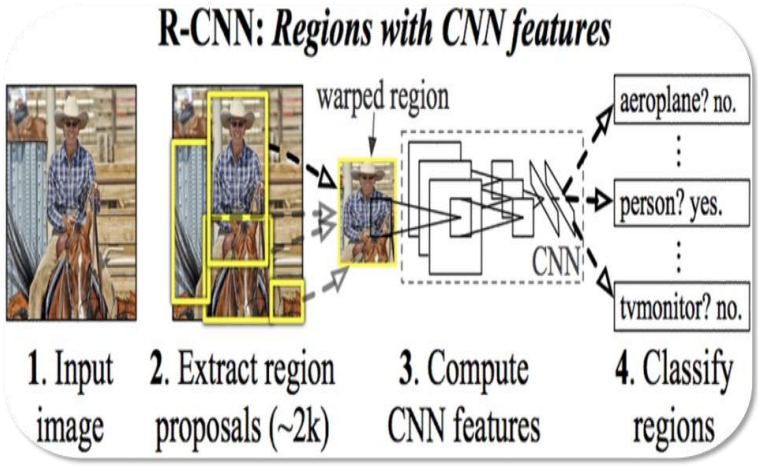
Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

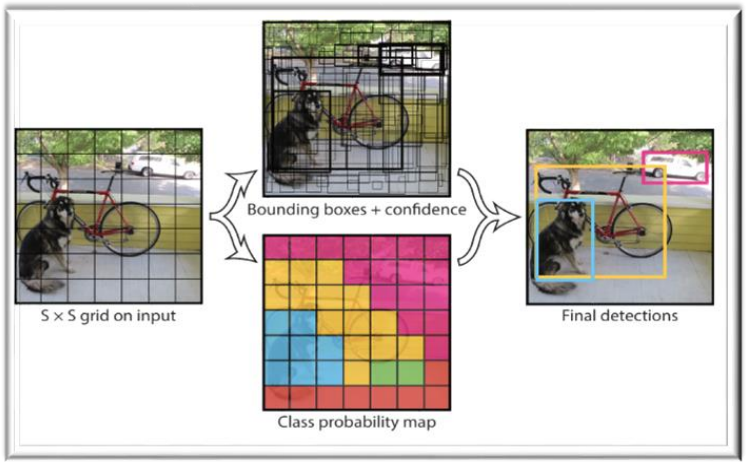
A. YOLO v3 (You Only Look Once v3)

YOLOv3 is that the latest variant of a preferred object detection algorithm YOLO – You Only Look Once that is extremely fast and correct algorithm program. It is a clever convolutional neural network (CNN) for performing object detection in the real-time. It applies single neural network to the complete image. The CNN separates an image into multiple regions to calculate its bounding box & probabilities for every region The method of extracting the regions that are likely to contain the object is known as Region Proposals. Region proposals have a better probability of containing an object. Several Region Proposal algorithms are projected to pick out a Region of Interest (ROI). So, R-CNN was projected with an idea of using the exterior region proposal algorithm. R-CNN stands for Region-based Convolutional Neural Network. It uses one in all the external region proposal algorithms to pick out the region of interest (ROI).

R-CNN uses selective search algorithms to come up with region proposals. R-CNN must train each half severally. Fast R-CNN is next version of the regional proposal algorithm. Fast R-CNN enhanced the drawbacks from R-CNN to create faster object detection algorithm. Faster R-CNN replaces the outside region proposal algorithm program with a Region Proposal Network (RPN). RPN learns to propose the region of interests that successively saves a plenty of your time and computation as compared to a Fast R-CNN. YOLO, that makes use of the entire upmost feature map to predict each confidences for multiple classes and bounding boxes.



YOLO works is that we have tendency to take an image and split it into an SxS grid, we are able to divide the image into any variety grids, depending on the complexity of the image. Once the image is split, every grid undergoes classification and localization of the object. In each of the grid we have a tendency to take m bounding boxes. For every of the bounding box, the network outputs a people class probability as well as offset values for the bounding box. The bounding boxes having the people class probability above a threshold value is chosen and want to find the object within the image. That means YOLO aims to predict class of object and bounding boxes specifying object location.



B. Dnn (Deep Neural Network)

DNN module extend the functionality of DNN framework. This module adds up as a module represents a gaggle of reusable code which is able to be put into any DNN website. Once put into any DNN modules, will simply dragged and dropped to a page. It will have multiple instance of same on one page. The DNN finds the proper mathematical

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

manipulation to show the input into the output, whether it be a linear relationship or a non-linear relationship. The network moves through the layers calculating the probability of every output. For each iteration, every frame of the input video gets processed beside the social distancing detection between every individual during an exceedingly crowd and is returned to the main function.

C. OpenCV (Computer Vision)

It stands for open source computer vision library in python which designed to solve computer vision problems. It is programming functions library mainly aimed toward real-time computer vision. It is an open source computer vision library as well as machine learning software library. It is used to monitor whether people are in distance or not. We will count the number of people who are unsafe because they are not maintaining social-distance. OpenCV-Python makes use of Numpy, which can be a highly optimized library for numerical operations with a MATLAB-style syntax. All the OpenCV arrays structures are converted to and from Numpy arrays.

D. Numpy

It is a python library used for work with (n-dimensional) arrays. It is created in 2005 by Travis Oliphant. It is open source project library. It can use freely. It is numerical computing tool.

II. RELATED WORK

In this section, we briefly discussed about previous work done on the social distancing in the context of the COVID-19. As the disease spread very fast, many researchers started giving attention to stop the spread of the disease. The researchers concluded that Social distancing was the alternative solution to slow down the spread of the disease. In the same background two students Vinitha, Velantina proposed a system.

There proposed system focuses on the way to identify the person on image/video stream whether the social distancing is maintained or not with the help of computer vision and deep learning algorithm by using the OpenCV and Tensor flow library. In there project they used 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 within the real-time with the help of bounding boxes.

Another students who worked on same system are Adina Rahim, Ayesha Maqbool and Tauseef Rana. There proposed system focuses on an efficient solution for real-time social distance monitoring in low light environments. They used the YOLO v4 algorithm for real-time person detection, which is trained on the ExDARK dataset. And for monitoring social distance, the authors used a motionless ToF camera to observe people at fixed camera distance and show resultant distance in real-world units.

III. PROBLEM DEFINITION

Due to sudden outbreak of COVID-19, maintaining some distance among people is crucial. And to slow down or to stop the spreading of corona virus social distancing is come as one of the best method among all the preventive measures. By keeping distance among people in public places we can break the chain of COVID-19. Previously there was no such type of application developed which detects people and calculate the distance between them. So these all problems are resolved with this application which detects and calculate the distance between people and gives alert if the distance between people is not approximately 50 inch or less than 50 inch so as to prevent or slow down the affect of COVID-19.

IV. PROPOSED APPROACH TECHNIQUE

The proposed system focuses on the to identify the person on image/video stream whether the social distancing is maintained or not with the assistance of computer vision and deep learning algorithm by using the OpenCV.

- Detect humans within the frame with yolo v3.
- Calculates the space between every human who is detected within the frame.
- Shows what percentage people are at High, Low and Not in danger.
- Find the distance between people to check whether a norm social distance of 50inch is maintained by people or not.

V. IMPLEMENTATION

Step 1: Calculate Euclidean distance between two points

We want to compute the Euclidean distance so as to spot the space between two bounding boxes. The function is used to obtain two points, as inputs for calculating the Euclidean distance and returns the calculated distance dst.

IJARCCE

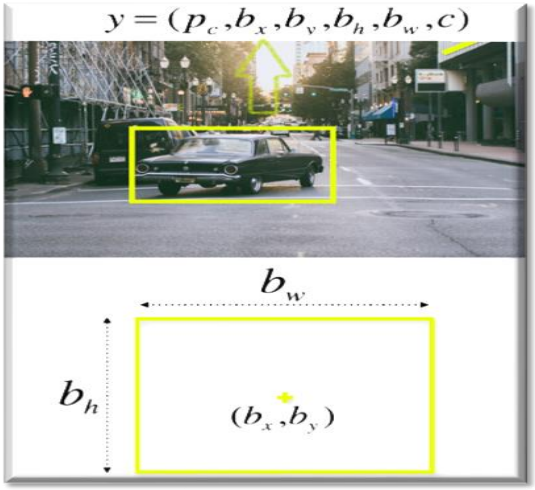
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

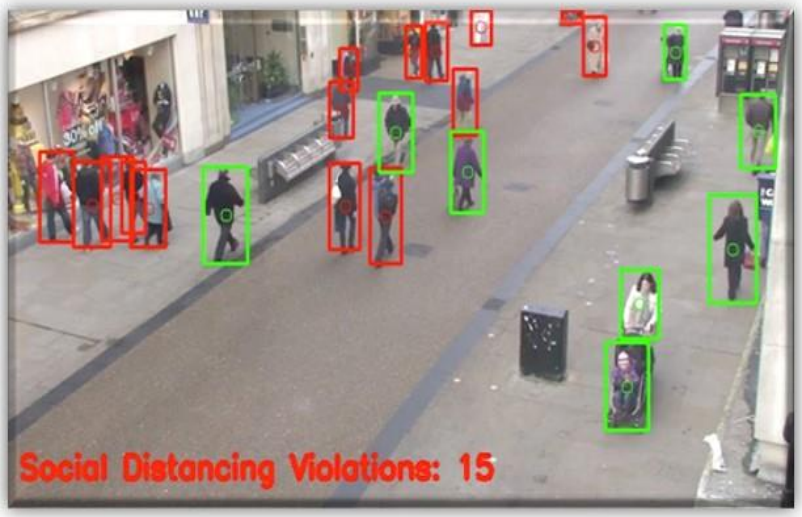
Step 2: Converts Center Coordinates into Rectangle Coordinates
The parameters x, y—the midpoint of the bounding box—and w and h are the width and height of the bounding box—taken as inputs. Then it will be converted into the centre coordinates to rectangle coordinates and return the converted coordinates, xmin, ymin, xmax, and ymax.



Step 3: Filtering the People Class from Detections and Getting a Bounding Box Centroid for Every People Detection
Initially, we check whether there is any detection within the image using a len(detections)>0 condition. If the condition successfully verified, then we create a dictionary called centroid dictionary and a variable called object ID set to zero. Then it will run through another condition that filters out all the other detection types, apart from people. We will store the center points of all the people detections and append it to the bounding box for people detected.

Step 4: Check which people bounding boxes are close to each other
First, we create an inventory that contains all object IDs of the under-threshold distance conditions. Through iteration, we will get all the possible combinations of close object detection to calculate the Euclidean distance using the function from ***Step 1.***
Then we set the social distance threshold value (equivalent to 6 feet) and check whether it satisfies the condition distance < 6 feet. Finally, we set colors for the bounding boxes. ***Red*** for at-risk (unsafe) people and ***Green*** for a protected people.

Step 5. Display risk analytics and risk indicators
In this final module, we indicate the number of people in danger during a given frame. Starting from the text we have to display with a counted number of people at risk. For better graphical representation, Using **for** check **in** range parameter, we tell the application to draw the lines between the nearby bounding boxes and iterate through the red_box_list.



IJARCCE

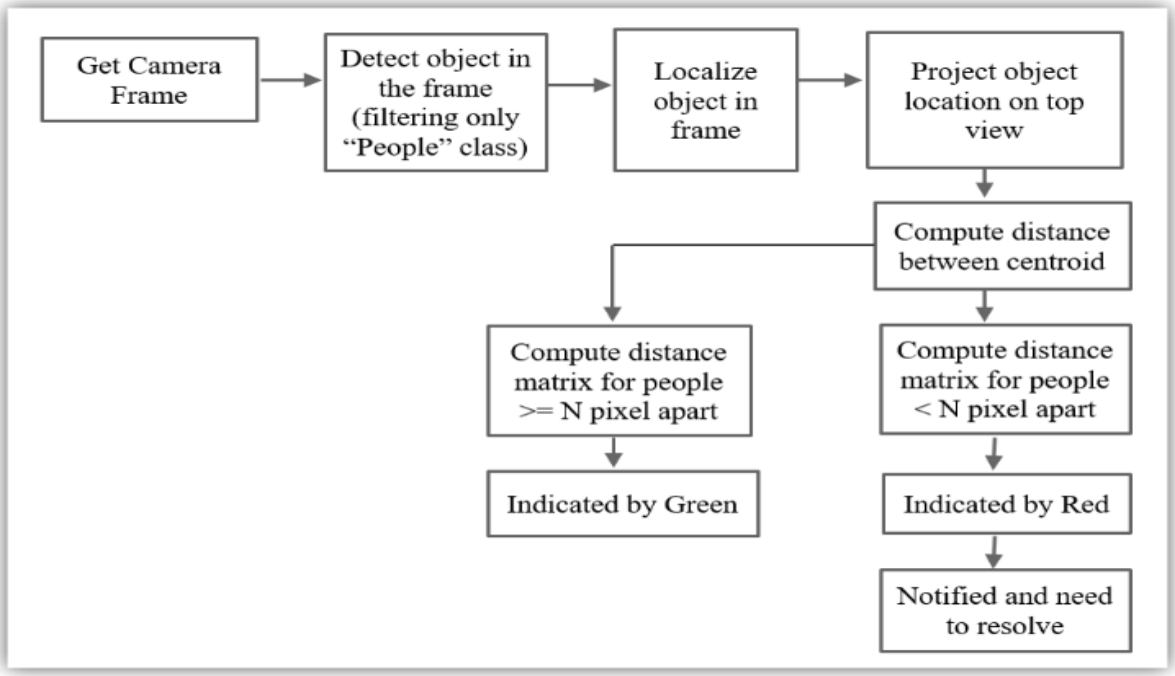
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

VI. SYSTEM FLOW



VII. CONCLUSION

The paper presents a methodology of social distance detection using python opencv. By using this system, we can easily detect the people and calculate the distance between them. Using this system, we can keep track on people on the public places such as traffic signals that they are maintaining the distance of approximately 50 inches between them or not. If the required distance is not maintaining by the people, then this system shows the red box on that person and by this we will easily found out the person who is not maintaining the distance and breaking the rules so that government can take strict action on this. This system works very effectively and efficiently to identify the distance between the people. This system can install in CCTV at public places such as restaurant, cinema halls, markets, shopping malls, schools, colleges, railway stations, airports and many more. It typically involves keeping a particular distance from others. The system also helpful in avoiding large gathering of peoples at wedding halls and at cemetery. This system aims to monitoring social distance within group of people.

REFERENCES

[1] Vinitha.V, and Velantina.V, "Social Distancing Detection System with Artificial Intelligence using computer vision and deep learning," *International Research Journal of Engineering and Technology (IRJET), Vol.7, Issue 8,* August 2020.
[2] Q. Zhao, P. Zheng, S-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems, vol.30, no.11, pp.3212-3232, 2019.*
[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems,* 2012, pp. 1097-1105.
[4] N. S. Punn and S. Agarwal, "Crowd analysis for congestion control early warning system on foot over bridge," in 2019 Twelfth International Conference on Contemporary Computing (IC3*). IEEE, 2019*, pp. 1–6.
[5] D.C. Cire¸san, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. "High-performance neural networks for visual object classification," *Arxiv preprint* arXiv:1102.0183, 2011.
[6] S. Agarwal, A. Awan, and D. Roth," Learning to detect objects in images via a sparse, part-based representation," *PAMI*, 26(11):1475– 1490, Nov 2004.
[7] L. Zelnik-Manor and M. Irani. "Event-based analysis of video," in *CVPR*, pages II:123–130, 2001.
[8] S. Ullman, "High-Level Vision: Object Recognition and Visual Cognition," *MIT Press*, 1996.
[9] H.A. Rowley, S. Baluja, and T. Kanade, "Neural Network-Based Face Detection," *IEEE Trans*. Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 23-38, Jan. 1998.
[10] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *CVPR, 2003*.
[11] M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," in *ECCV*, pp. 18–32, 2000.
[12] Y. LeCun, K. Kavukcuoglu, and C. Farabet. "Convolutional networks and applications in vision," in Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, pages 253–256. *IEEE*, 2010.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.105113

[13] C. Wojek, P. Dollar, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans*. Pattern Anal. Mach. Intell., vol. 34, no. 4, p. 743, 2012.
[14] D. Ribeiro, A. Mateus, J. C. Nascimento, and P. Miraldo, "A real-time pedestrian detector using deep learning for human-aware navigation," arXiv:1607.04441, 2016.
[15] Pranav Shinde, Srinandini Yadav, Shivani Rudrake, Pravin Kumbhar. "Smart Traffic Control System using YOLO," *International Research Journal of Engineering and Technology (IRJET), Volume: 06, Issue: 12, Dec 2019.*

AUTHOR BIOGRAPHY

Chhaya Bangadkar is pursuing her B.E. (Final year) in Computer Engineering from MIET, University of Nagpur, India. She is mostly interested in coding in C, C++, Java, HTML and Laravel.

Faroog Fatma Jafri is pursuing her B.E. (Final year) in Computer Engineering from MIET, University of Nagpur, India. She is mostly interested in coding in C, C++, Java, HTML and Laravel.

Mohsina Sheikh is pursuing her B.E. (Final year) in Computer Engineering from MIET, University of Nagpur, India. She is mostly interested in coding in C, C++, Java, HTML and Laravel.

Ummul Khair Memon is pursuing her B.E. (Final year) in Computer Engineering from MIET, University of Nagpur, India. She is mostly interested in coding in C, C++, Java, HTML and Laravel.

Ashwini Meshram is pursuing her B.E. (Final year) in Computer Engineering from MIET, University of Nagpur, India. She is mostly interested in coding in C, C++, Java, HTML and Laravel.