International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 05 | May 2021 www.irjet.net p-ISSN: 2395-0072

Social Distancing Detector using Deep Learning

Neha Yadav[1], Nikita Sule[2], Sakshi Yadav[3], Sujata Kullur[4]

[123]*Student, Dept. of Information Technology, Usha Mittal Institute of Technology, Maharashtra, India*
[4]*Professor, Dept. of Information Technology, Usha Mittal Institute of Technology, Maharashtra, India*

Abstract - *Pandemics are not a new thing and keep taking place from time to time. Over the centuries, it has been found that the pandemic diseases spread very easily and quickly because of not maintaining a distance from other people during the pandemic. In 2020, India and many other countries had to face a lockdown due to the pandemic. This caused recession and hurt the economy of the nation as well as the lives of the people who had no economic support. Therefore, in the long run, the better alternative would be to practice social distancing very strictly while still continuing to earn. This Python project aims to curb the spread of the pandemic diseases by building a social distancing detector that will analyze a camera's live video feed and detect if a social distance is being maintained or not. This will be done using Object Detection, Computer Vision and Deep Learning. The model will display the social distancing violations that are being made and would be able to help the concerned authorities to take necessary actions if the number of violations are greater than a threshold value.*

Key Words: **Pandemic, Social Distancing, Object Detection, Deep Learning, Computer Vision.**

1.INTRODUCTION

Currently, the outbreak of a pandemic disease has put the world at a halt because of how rapidly it spreads among people. One can act as a catalyst in spreading the disease by coming in contact with an infected person and then further spreading it to other people by not maintaining the required social distance. This is not the first time that the world is going through something like this. It is very important to be ready to face such pandemics now and, in the future, as well. Wearing masks and maintaining distance from others are the two significant preventive measures that one should take in order to reduce the rate at which the disease spreads.

It has been found that the number of people who will contract a disease over a period of time is comparatively lesser when people follow the social distancing guidelines as shown in Fig. 1. Not following the social distancing guidelines can increase the rate of infection exponentially. If the number of infected people will increase exponentially, it will be impossible to treat each and every one as the number will be too huge as compared to the healthcare capacity available. This will increase the load on the healthcare workers and will increase the death rate of people infected by the disease. While going through a situation like this, it is necessary to take collective action till the vaccines are not available. This involves wearing masks, washing hands and maintaining social distancing. These measures have proven to be effective in the past.

Social Distancing Detector, a technology-oriented solution, is an initiative to reduce the rate at which a virus spreads by detecting and analysing people visible in the live input video feed that comes from a bird's point of view and check if a social distance is being maintained between them or not and then alert the concerned authorities if there are any social distancing violations taking place so that they can take necessary actions

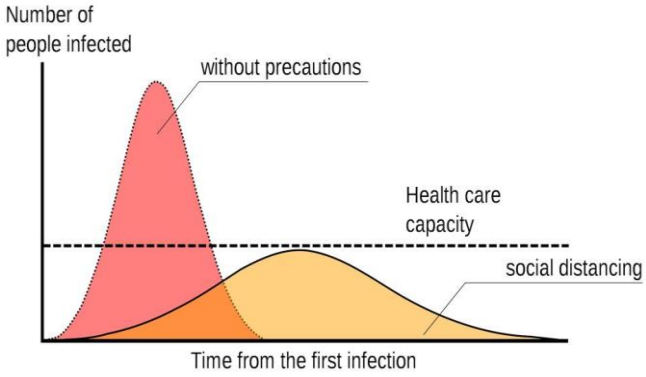

Fig-1: A sample epidemic curve in the absence and presence of social distancing. (Image credit: Johannes Kalliauer/ CC BY-SA 4.0)

2. LITERATURE REVIEW

Ever since the recent virus has spread, many countries are trying to take help of technology to find ways to slow down the pandemic and eventually diminish it. Indian government launched an app called Aarogya Setu that tracks one's interaction with someone who could have tested Covid-19 positive through Bluetooth and GPS generated social graph. This app alerts a person if he/she comes in close proximity of someone who tests Covid-19 positive. This is one way to slow down the pandemic, another way involves Object Detection which has proved to be very helpful.

Many researchers have found out that object detection can be used to create intelligent systems to monitor human activities and calculate the distance between those humans. One of the researchers Yadav et al. [3] put forward the idea to use raspberry pi4 with a camera to monitor public places. It would continuously detect if people maintain a social

distance or not as well as check if people wear masks or not. If a person would be found roaming without a mask, his/her picture would be sent to the concerned authorities and if a person violates the social distancing guideline continuously for a period that exceeds the threshold period, an alarm would start ringing to alert the concerned authorities. Another method proposes to use critical social density to avoid crowds by modulating inflow to the region of interest. This method was proposed by Yang et al [1]. A localization method was proposed by Nadikattu et al. that would track human's positions un the surrounding based on sensors [2]. Various methods involve using R-CNN (Region based Convolutional Neural Networks), YOLO (You Only Look Once) and SSD (Single Shot Detectors). However, this project will use object detection along with deep learning to calculate the number of social distancing violations in real time using YOLO v3. The goal is to build a model that will prove to be an efficient and faster way to curb the pandemic and alert the people to maintain social distancing if they are found in close proximity of others.

3. PROPOSED METHODOLGY

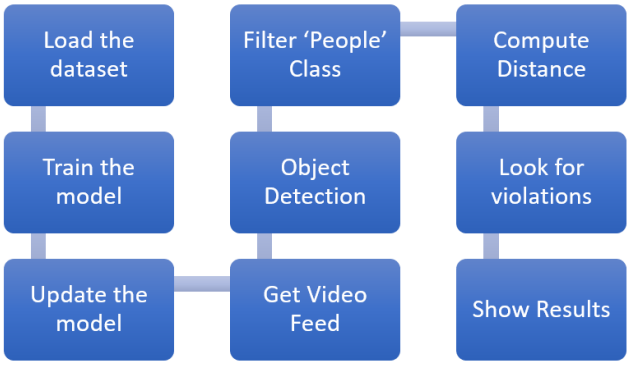

Fig-2: Flowchart of Social Distancing Detector that gives an overview of the tasks to be performed.

This project will be developed using Python and its packages like OpenCV, NumPy, SciPy, Keras, TensorFlow and Argparse. We will be using OpenCV to collect the live input video feed from the laptop webcam and feed it into a Deep Learning Model. The deep learning model will be trained and tested using a convolutional neural network to classify the different object classes visible in the video and give us the objects that we are interested in i.e. People and display a bound box around them and we will then compare it to the distance set forth by public health professionals. If two people are too close, they will be added to the violation set and the results of the same will be displayed accordingly.

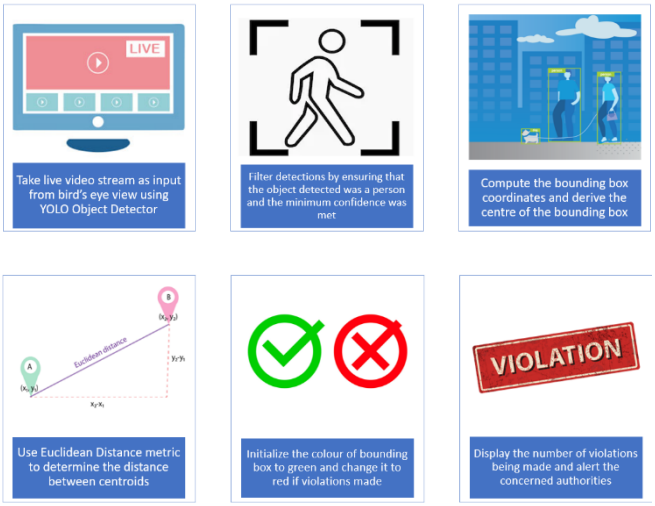

Fig-3: [Read from left-to-right] Working of Social Distancing Detector. This image displays the six significant steps that are necessary to understand the working of the model.

The first task is to find an appropriate dataset to train and test the object detection model. Once that is done, we can take the live input stream from a webcam placed such that we get a bird's point of view and start detecting the objects. Since we need to detect only humans, we will filter out the detections and keep only those objects that are detected as humans. We will create a bounding box around these objects and derive the center of these bounding boxes. If the input stream has more than one human present then we will then calculate the distance between the two bounding boxes using a distance measurement technique. The color of the bounding boxes will be green initially but will change to red if the distance between that bounding box and some other bounding box is less than the distance that has been decided as the threshold distance. This project focuses on working with input video files along with real time detections.

3.1 Dataset

Training the computer vision model with the appropriate dataset is a common challenge. Huge amount of data is required to train deep learning models. ImageNet, COCO (Common Objects in Context), Google's Open Images, MNSIT handwritten dataset and cityscapes dataset are the most popular datasets that are freely available online

The Microsoft COCO dataset is also known as the gold standard benchmark for evaluating the performance of state-of-the-art computer vision models. Although it is widely used among the computer vision research community, the COCO dataset is less well known to general practitioners. The COCO dataset stands for Common Objects in Context, and represents a vast array of objects that we regularly encounter in day-to-day life. The COCO dataset is labelled and provides data to train supervised computer vision models that are able

to identify the common objects in the dataset. These models are still far away from perfect; therefore, the COCO dataset provides a benchmark for evaluating the periodic improvement of these models through computer vision research. Another motivation for the COCO dataset is to supply a base dataset to train computer vision models. Object Detection Model.

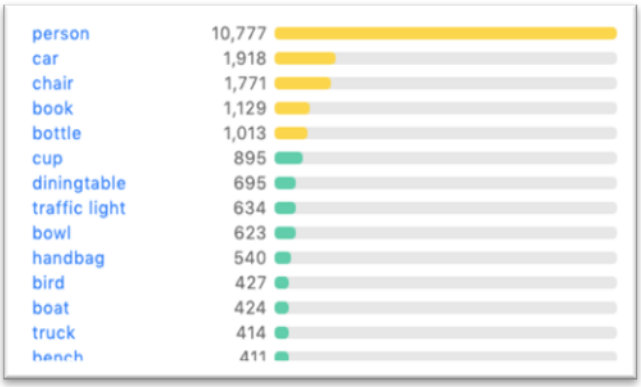

Fig-4: COCO Dataset has 10,777 images for the 'Person' class and is the most appropriate dataset to train our model. (since the project deals with detecting people)

3.2 Object Detection Model

Other font types may be used if needed for special purposes. There are many object detections models but the 3 main object detection models are RCNN, SSD and YOLO. The R-CNN uses regions to localize the objects within the image. Instead of looking at the entire image, it only looks and processes that part which have a better chance of having an object present there. SSD can detect multiple objects within the image in just one single shot and has high accuracy. It eliminates the bounding boxes and includes a convolutional filter for predicting object categories and bounding box locations.

The YOLO (You Only Look Once) has a different approach altogether when it has to deal with object detection. It is named YOLO because it looks at an image only once and doesn't need to look at it more than once to generate the coordinates of the bounding box and determine the classes and their probabilities. The biggest advantage of using YOLO is that it has very good speed - it's fast and 45 frames per second can be processed at once. Generalized object representation is easily understood by YOLO. IT is easy to use and understand and it is comparatively better than RCNN and SSD because they provide accuracy whereas YOLO provides more speed and since our project deals with real time object detection, it needs to be quick. YOLO V3 is an improvement of the YOLO detection network. Compared to prior versions, it features multi-scale detection, stronger feature extractor network, and a few changes within the loss function. As a result, this network can now detect more targets from small to big. We will be using YOLO V3 for our project and we can configure it detect any type of object. One can modify the CNN architecture itself and play around with it. Based on different versions, there can be many more configurations from V1 to V3 to tiny layers to full training.

The last step that is performed by the object detection models is non maximum suppression and it is used to select the best bounding box that perfectly covers the object and whose borders are the most appropriately placed. The objects within the image are often of different shapes and sizes. The detection algorithms create multiple bounding boxes to fit the objects perfectly. There should be at least one bounding box for all the objects that are present in the image. NMS (NON-Maximum Suppression) is used to the best and most suitable bounding out of a number of bounding boxes. this system is used to "suppress" the bounding boxes that don't perfectly fit the object and keep the one that does. the aim of non-max suppression is to pick the simplest bounding box for an object and suppress or reject all the other bounding boxes visible. This technique takes considers the objectiveness score that is given by the model and the IOU or overlap of the bounding boxes. The bounding box with the absolute best objectiveness score is selected by the NMS while all the other bounding boxes with high overlap are discarded.

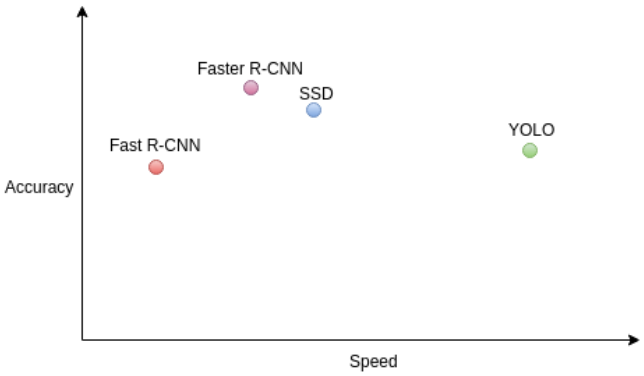

Fig-5: YOLO is faster than all the other object detection models which makes it suitable for real time object detection. (Image credit: Sayon Dutta)

3.3 Distance Metric

Euclidean Distance is the most used distance metric. Most machine learning algorithms use Euclidean Distance to find out the similarities between observations.

The Euclidean is usually the "default" distance utilized in KNN i.e., K-nearest neighbors (classification) or K-means (clustering) to seek out the "k closest points" of a specific sample point. Other prominent examples are hierarchical clustering, agglomerative clustering (complete and single linkage) where you would like to seek out the space between clusters.

$$\sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}.$$

Fig-6: Distance measurement between the samples p and q in an n-dimensional feature space

4. RESULTS

Following are the results generated after creating the social distancing detection model and testing the model on two sample input videos as well as on a real time input. The green bounding boxes around people indicate that those people are not violating the social distancing rules while the red bounding boxes indicate otherwise. The total number of social distancing violations are displayed at the bottom left of the screen.

Fig-7: Graphical User Interface of Social Distancing Detector

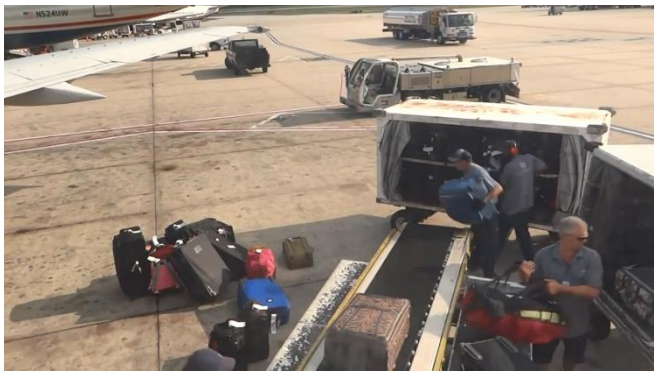

Fig-8: Snapshot of input video 1 that shows people working at an airport.

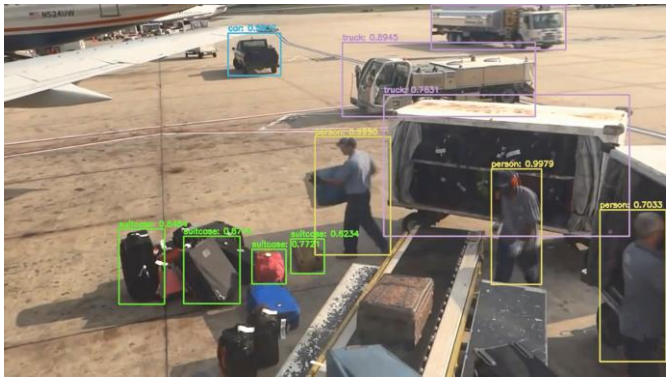

Fig-9: Snapshot of output video 1 after performing object detection.

The snapshot shows the different objects that were detected. The objects include 'person', 'suitcase', 'car' and 'truck'. The class labels are displayed along with the percentage of confidence at the top of the bounding boxes. The colors of the bounding boxes are different based on the range of their confidence percentage.

Fig-10: Snapshot of output video 1 that shows the violations being made by the people working at an airport.

Fig-11: Snapshot of input video 2 that shows pedestrians walking on a footpath

Fig-12: Snapshot of output video 2 that shows the 2 violations being made by the pedestrians walking on a footpath.

Fig-13: Snapshot of real time social distancing detection that shows the total number of violations being made by the pedestrians walking.

Fig-14: Snapshot of real time social distancing detection that shows zero number of violations being made by the pedestrians walking.

5. CONCLUSION

This project detects people, calculates the distance between them and helps the concerned authorities to keep a check on the violations being made by the people. It also gives us the confidence percentage of a label which tells us how like is an object to belong to a class it has been identified to. This model can be used to help maintaining social distancing on roads, outside shops, inside banks and all kinds of public places. Nowadays, social distancing alongside other basic sanitary measures is extremely important to stop the spread of the virus. By monitoring the space between two individuals, we can confirm that a person is maintaining social distancing which will reduce the load on healthcare workers, reduce the death rate due to pandemic, curb the pandemic and eventually diminish it. By monitoring the space between two individuals, we will confirm that a private is maintaining social distancing within the right way which can enable us to curb the pandemic.

ACKNOWLEDGEMENT

We wish to express our heartfelt gratitude to all the people who have played a crucial role in the completion of this project. Without their active cooperation, the preparation of this project would not have been possible within the specified time limit. We are thankful to our respected project guide Prof.Sujata Kullur who supported our team throughout this project with the utmost cooperation and patience and for helping us to do this project. We would like to express our sincere gratitude to Dr. Sanjay Shitole, HOD of Department of Information Technology and our principal Dr. Sanjay Pawar and all the teaching faculty who helped us to complete this project with complete focus and attention.

REFERENCES

[1] D. Yang, E. Yurtsever, V. Renganathan, K.A. Redmill, U. Ozg¨uner, A vision-based social distancing and critical density detection system for covid-19, arXiv e-prints pp.arXiv–2007 (2020)

[2] R.R. Nadikattu, S.M. Mohammad, P. Whig, Novel economical social distancing smart device for covid-19, International Journal of Electrical Engineering and Technology (IJEET) (2020)

[3] S. Yadav, Deep learning based safe social distancing and face mask detection in public areas for covid-19 safety guidelines adherence

[4] W. Liu et al., "SSD: Single Shot MultiBox Detector", 2021.

[5] M Bielecki, R Z¨ust, D Siegrist, D Meyerhofer, G A G Crameri, Z G Stanga, A Stettbacher, T W Buehrer, J W Deuel, "Social distancing alters the clinical course of covid-19 in young adults: A comparative cohort study Clinical Infectious Diseases", 2020.

[6] N Singh Punn, S K Sonbhadra, S Agarwal, "Monitoring covid-19 social distancing with person detection and tracking via fine-tuned yolo v3 and deepsort techniques", 2020.

[7] L Ramadass, S Arunachalam, Z Sagayasree, "Applying deep learning algorithm to maintain social distance in public place through drone technology, International Journal of Pervasive Computing and Communications", 2021.