Social distancing detector

Abhay Singh Rana[1], Umang Sahu[2], Yugank[3], Aryan Srivastava[4]
[1,2,3,4] *Rajkumar Goel Institute of Technology/AKTU*

***Abstract* - Computer vision and machine learning techniques have been used in order to detect object in an image or in a video. Using these techniques the social distancing followed by the people can be evaluated which is really a necessity for the people nowadays, so that they can avoid or protect themselves from the ongoing coronavirus. It is an application of computer vision and deep learning. Computer Vision is a branch of computer that works like an eye of the computer that can recognize and also understand images and activities while deep learning is a function of artificial intelligence that try to mimic the working of human brain. Using highly accurate object detection-algorithms and methods such as R-CNN, Fast-RCNN, Faster-RCNN, Darknet and fast yet highly accurate ones like SSD and YOLO4. we can detect each and every human in image by the area object in an highlighted rectangular boxes and identify whether or not they are following social distancing or not. This also includes the accuracy of each method for the identification of the objects. It also shows the number of people violating the social distancing rule at a given moment or a frame.**

***Index Terms* - Computer Vision, Deep Learning, Social distancing detection, Machine learning.**

I.INTRODUCTION

As of June 2021, there are about 11,640,063 active cases of coronavirus and are continuing to fluctuate around that number. Also, more than 38 lakh people have lost their lives all over the world in this pandemic. The amount of deaths could have been greatly decreased if people would have followed the proper covid-19 guidelines issued by the WHO (world health organization) which also included the most important precaution i.e. social distancing. Social distancing is the most useful way to block the chain of coronavirus spread from person to person. A person must maintain an appropriate distance from other people to avoid the transferring of this virus. Making sure that people is following the minimum distance rule is not possible for a human eye to execute over large areas, though they can cover smaller areas like gates and narrow roads but it is not possible to cover broad areas.

So the best way to tackle this situation is to deploy automated systems to detect the social distancing followed by the people. The computer vision can be used along with the availability of the capabilities of neural networks to detect human and distances between them. After going through and surveying multiple research papers is observed that we can train the model using deep learning techniques and frameworks to detect humans and computer vision would be able to capture those humans in the image using the model trained by the deep learning frameworks. Using a deep learning framework such as YOLO (You only look once) or SSD we can check whether the people are following the social distancing rule or not. The system would be able to detect the human and mark them with a green colored rectangle around their body. The distance between those rectangles can easily be calculated with the help of simple Python based calculations. The distance would be calculated in the form of pixels that can later be converted into meters depending on various parameters like, the distance from the camera or even the size of the rectangles that may denote the size of the humans. The performance of these systems would greatly depend on the hardware of the system because to produce real-time results in reasonable time may require a powerful system. So it is sometimes required to process the images or the video in the recorded form instead of real-time state.

II. LITERATURE REVIEW

Deep learning multiple levels of learning and complex neural networks and represent data with multiple levels of abstraction and mimics how the brain receives and understands complex or layered information, thus correctly capturing structures of large, scaled data. Deep learning has various tested methods, comprising neural networks, probability-based models, and a number of supervised and unsupervised learning algorithms. The recent explosion of interest in deep learning is due to the fact that they have been shown to outperform previously used techniques in several tasks along with the presence of loads of complex data from different sources (e.g., images, audios, videos, sensory and etc.).

We find that object detection is the procedure of determining the presence of the instance of an image to which the object belongs and estimates the coordinates of the object by displaying the box bounded around the object.[4] We also face different challenges such as partial and full occlusion, various illumination conditions, poses, scales, etc. are needed to be taken care of while performing the object detection. The availability of deep learning frameworks present today is exhaustive. Most commonly used frameworks are YOLO and SSD.

The most commonly used technique and one of the most efficient one, YOLO is a single model for object detection. In this we learn about how to make our model simple to construct and train that directly on full images. Unlike classifier-based approaches, YOLO has been built upon on a loss function that directly points to detection performance and the entire model is trained as a whole. [3]

After studying various research papers, we came to know various Deep learning concepts and came across various techniques that we can use with the computer vision to detect different objects by training our own custom model.[1] After learning about YOLO we are confident enough that we can adopt this technique in our project and can achieve the best possible results in detecting the humans.[2] Using deep learning we can train our model and using YOLO4 we can detect the specific object that we want to detect.[3] And finally we will combine the designed model with a well-designed smooth front-end to make interactions with our program easy and user friendly.

III. METHODOLOGY

After going through various different research papers, the following frameworks or techniques have been identified:

A. YOLO 4

This framework does not look at the whole image but instead it look for the parts of the image which may have high probability of containing the image. YOLO is also known as “You Only Look Once” is an algorithm used for detecting objects and is very different from the region-based algorithm like Relational convolutional neural networks. In YOLO one single convolutional network predicts the bounded boxes and the class probability for those boxes. YOLO works by taking an image and divide it into an YxY and in those grids we take n number of bounding boxes. For every bounding box, the network provides an output in the form of the class probabilities and offset values for the bounded box. The bounded boxes have the class probabilities higher than a threshold value is selected and used to identify the location of the object inside the image. YOLO is much faster (45 fps) than any other object detecting algorithm.[3][4] The limitation of YOLO algorithm is that it finds it hard to deal with small objects inside the image, for example, it might have difficulties in identifying a group of insects. This is due to the space related constraints of this algorithm. [3]

B. SSD

The SSD also known as Single Shot Detection object detection composes of 2 parts:

1. Extract feature maps from the image, and
2. Apply the convolution filter to detect objects in the image

Single Shot Detection uses VGG to extract the feature maps. Then it detects the objects using the 3rd layer. For each location in the image it makes four object predictions. Each prediction comprises of a boundary box and scores for each class and we pick the maximum score as the class for the object that is bounded. 4 predictions per cell irrespective of the depth of the feature maps. As we expected, multiple predictions contain no object. We explain the SSD detects objects from a single one layer. It uses more layers for detecting objects with depending on others. As CNN reduces the space related dimensions step by step, the overall resolution of the feature map also reduces. SSD uses lower resolution layer for the detection of large-scale objects. For example, the 4× 4 feature maps are used for the large-scale object. SSD adds six more auxiliary convolution layers to image after VGG. 5 of these layers would be added for detection of object. In which 3 of those layers, we make six predictions instead of four. In total, SSD makes 8732 predictions using six convolution layer.

IV. HELPFUL HINTS

The rise of coronavirus has always been unpredictable. The total number of cases has decreased and increased at different places with no specified pattern.

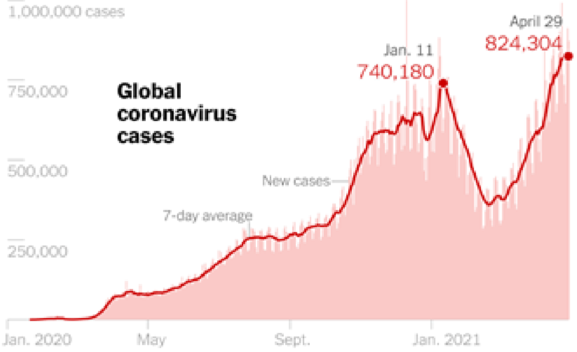

Fig 1. Covid-19 rising cases graph

It might take some more time to reduce the effects of this pandemic all we can do is just to reduce the spreading of the virus. All we can do is to just follow the rules and maintain good distance with others. The basic roadmap to follow while building a social distancing detection system.[2][6] These are the steps while building an efficient system.

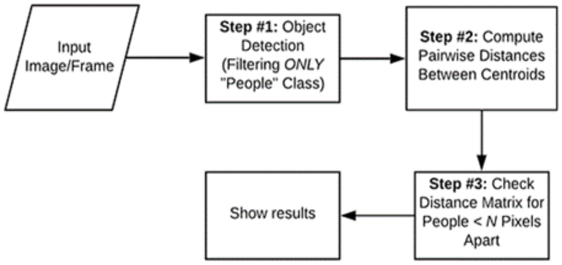
Input Image/Frame
Step #1: Object Detection (Filtering ONLY "People" Class)
Step #2: Compute Pairwise Distances Between Centroids
Step #3: Check Distance Matrix for People < N Pixels Apart
Show results

Fig 2. Social distancing detector idea roadmap

The built system might act differently on different system based on the hardware present and the quality of model traine. A successful system might deliver similar result like the figure 3.

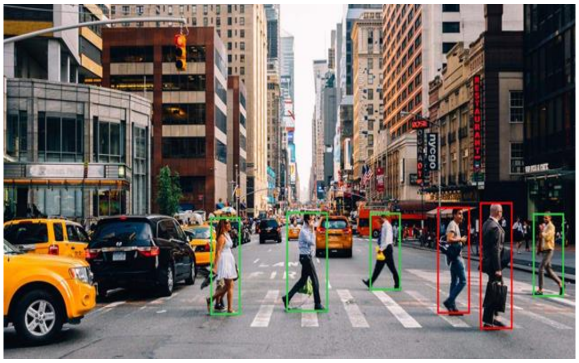

Fig 3. Social distancing detector results

V. CONCLUSION

By using this and based on experimental results we are able to detect object more precisely and identify the objects individually with exact location of an object along with the difference between them in the picture in X, Y axis.[1][4]

Our social distancing detector did not leverage a proper camera setting, meaning that we could not map distances in pixels to actual measurable units (i.e., meter, foot, etc.). Therefore, the first step to improve our social distancing detector is to utilize a proper camera setting. Doing so will result in better results and enable us to compute actual measurable units (rather than pixels).

Secondly, we can consider applying a top-down transformation of your viewing angle, as this implementation has done. Applying a perspective transform or using stereo computer vision would allow you to get a more accurate representation of social distancing detector with OpenCV library. [1][2]

Thirdly is to improve the people detection process. OpenCV's YOLO implementation is quite slow not because of the model but because of the additional post-processing required by the model which needs a really powerful system. [3][6]

To further speedup, we can consider utilizing a Single Shot Detector running on your dedicated GPU — that will improve frame throughput rate considerably. For Nighttime visual detection, night vision mode should be available as an inbuilt feature in the camera used.

REFERENCES

[1] Computer vision, https://en.wikipedia.org/wiki/Computer_vision.

[2] Deep learning, https://www.pyimagesearch.com/category/deep-learning/.

[3] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi (2016), You Only Look Once: Unified, Real-Time Object Detection

[4] Ajeet Ram Pathaka, Manjusha Pandeya, Siddharth Rautaray (2018), Application of Deep Learning for Object Detection

[5] Athanasios Voulodimos, 1,2 Nikolaos Doulamis,2 Anastasios Doulamis,2 and Eftychios Protopapadakis (2018), Deep Learning for Computer Vision: A Brief Review

[6] The beginner's guide to implementing YOLO (v3) in TensorFlow 2.0, https://mc.ai/the-beginners-guide-to-implementing-yolo-v3-in-tensorflow-2-0-part-1