Managing Crowd Density and Social Distancing

Medhini Kulkarni, Rishi Deedwania, Parth Mudgal, Prof. Aditya Bhope
Department of Electronics and Telecommunication Engineering
Mukesh Patel School of Technology Management and Engineering,
Juhu, Mumbai, India

***Abstract*— The aim of this paper is to propose a model that aids in estimating crowd density of a particular area as well as in the means of social distancing as per certain guidelines. This is done using model based multi-source approach. This model surveys a small public gathering where manual counting is not possible. This is done by extracting input video frames and processing each frame and passing it into the model for further detection and implementation. The concept of object detection is used to detect individuals in the given survey feed. The model then computes the number of people present in the area. The distance between people present in that place is also computed using Euclidean metric. These parameters aid in ensuring that the particular area's social-distancing guidelines and protocols are being followed. The proposed model is based on YOLO, a deep learning algorithm which performs the function of object (here-human) detection and helps classify the required parameters required in this project. This algorithm will be described in detail ahead in the paper.**

I. INTRODUCTION

Crowd analysis and monitoring is an essential task in public places to provide safe and secure environments. In past decades, many crowd disasters have taken place due to lack of crowd control management. Even though crowds are made up of independent individuals, each with their own objectives and behaviour patterns, the behaviour of crowds and individual feature have been widely understood in order to have collective characteristics that can be described in general terms. Crowd data such as density and flow are an important factor in the planning, design and management of public facilities subject to dense pedestrian traffic.
In the recent break out of the COVID-19 pandemic social distancing and limited crowd gathering play a major role in the safety of people. This project and the model proposed is an attempt to have a system that estimates crowd density as well as makes sure social-distancing guidelines are being followed to ensure public safety.

This project revolves around the counting of the crowd using model-based multi-source approach method. It involves taking into consideration a small public gathering where manual counting is not possible. It begins with an image processing step which then goes on to detect humans in this area. Final computation step involves calculating number of people present and the corresponding distance between them. This is done to ensure social-distancing guidelines are being followed in accordance. As a model, to monitor human crowd is the need of an hour to ensure no guidelines and protocols are tempered with, in a public place.

II.LITERATURE SURVEY

Object detection and image processing both have been studied extensively and are in fact still in the works to be made more efficient and prevailing. With machine learning and deep learning advancing in the last few years, several classifiers and algorithms have been developed and many are also in the works of further research and improvements. Initially, Haar wavelet transform (HWT) for image processing and SVM for classification was proposed in 2001. [14]. This paper proposed a vision-based approach but lacked in detection efficiency. Over the decades, crowd density and analysis prevailed and improvements were made.

In 2011, a paper was proposed to work on a density-based approach to human detection and crowd analysis. A baseline tracking procedure consisting of detection, geometric filtering and agglomerative clustering was computed. This made sure to improve detection as compared to earlier paper.[15]. In an attempt to focus on SVM classification, video-frames were collected meticulously so as to create a data cluster that automatically pre-processes the SVM data sets. This minimized the presence of noise key frames [16]. A paper using k-means clustering was also proposed which estimated the number of pedestrians which was used to give speedy results in low density crowds. [11]. Fourier analysis were also implemented in order to detect in a largely dense crowd video frames coupled with head-detections, interest-point based counts and the use of confidence maps. [18]. The area of image processing also involved the usage of gray level co-occurrence matrix (GLCM) whose characteristics were used to feed into SVM data for further classification. [12].

In 2015, the use of an extensive dataset provided by Microsoft COCO made image processing and labelling easier. It used the concept of bounding boxes while labelling which gave further clarity in detection and made room for future scope in object detection.[9]. In the coming decade, the use of a new, faster deep learning algorithm named YOLO came into the works, along with the use of COCO datasets. [8]. It was incorporated into a model that viewed object detection as a regression problem and was used to predict bounding boxes as well as classify them, all in a single neural network. This ensured an end-to-end detection to be made possible along with desired optimizations. [7][13].

Diving further into the image processing aspect, a multi-source-based approach involving Fourier analysis, Local binary pattern, Gray level dependence matrix (GLDM) features and Histogram of oriented gradient (HOG) for head detection to estimate the total count was proposed in 2017. [17]. This paper focused mainly on still images for crowd density estimation. A better version of YOLO called O-YOLO (optimised YOLO) was proposed which was 1.18 times faster than YOLO. [1]. In addition to this, a feature pyramid network (FPN) was introduced which worked on extracting features with speed and accuracy to use in YOLO implementation.[6][5]. YOLO was also proved more efficient on comparison with other deep learning algorithms such as CNN and R-CNN. [2]. A deeper convolutional network was also proposed in the year 2019, which focused on both crowd-counting as well as high density crowd analysis. [10]. R-CNN and its faster versions – Fast R-CNN and Faster R-CNN were also compared with YOLO and YOLO- v1, v2, v3 which gave us the desired results and route to the usage of YOLO proving to be most useful. [3][4].

III.METHODOLOGY

As discussed earlier, various methods have been studied extensively, various deep learning algorithms, classifiers coupled together to result in the field of this project. However, in this paper we will be using YOLO, a new deep learning algorithm which aids in detection as well as classification of required parameters that are further used to compute crowd number and the distance between individuals in the given area.

A) *Algorithm*

In this paper we have used YOLO v3 method and the language used is Python. The basic working of this model can be depicted by a flowchart attached as follows:

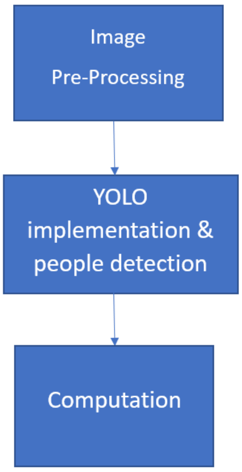


Fig. 1 Program Algorithm

B) *System description*

As shown earlier the model has been divided into three main stages which will be explained in detail further.

1. IMAGE PRE-PROCESSING –

This is the most important stage of this model. It involves extracting of frames of input video feed. These frames are individually processed and smoothened out. This is done to obtain required computation parameters and the output of the processed stage is later fed into the YOLO model for further implementation and detection.

2. DETECTION AND IMPLEMENTATION –

In this stage YOLO is implemented on the input frames by darknet and people are detected. This is precisely done by taking measure of parameters – bounding boxes, centroids and confidences. COCO dataset has been used in order to detect classID (i.e. personIdX in this case) and give accurate detection. The number of detections made i.e. number of people detected in the respective video frame is also computed. These values aid in the computation stage later. This is known to be the fastest real-time object detection algorithm and takes 45 seconds to implement per frame as compared to other deep algorithms discussed earlier. The output of this stage is passed on to computation stage.

3. COMPUTATION –

This stage has two main objectives. One is to count the crowd and second is to determine distance between any two individuals in the given video feed. This is done by loading the YOLO object detector that is trained on COCO names on to the darknet. A minimum threshold is set for minimum distance required to be followed as per guidelines. The Euclidean distance is calculated from centroids of the bounding boxes and compared with minimum threshold. These values including the total number of people detected and the number of social distance guidelines being violated are all displayed on to the screen of each video frame.

C) *YOLO (You Only Look Once)*

YOLO is a clever convolutional neural network (CNN) for doing object detection in real-time. It is a new approach to object detection. As opposed to the past work in the field of object detection, classifiers are repurposed and used for detection as well. YOLO network comprises of only a single neural network. This single network alone predicts bounding boxes and class probabilities directly from full images in one evaluation. This model has several advantages over classifier-based systems. In this, the entire image is viewed at once at test-time and the predictions are formed by the global context in the image. Unlike systems like R-CNN, which require numerous prediction values for an evaluation, YOLO is extremely fast and has an efficient evaluation process. It is found to be more than 1000x faster than R-CNN and 100x faster than Fast R-CNN.

YOLOv3 is the third version of YOLO which has been implemented in this project. It makes use of darknet 53. [4]. With the help of YOLO, end-to-end training is applied along with real-time speed. This allows in maintaining high average precision. The mean average precision of a YOLOv3 model is 63.4 for 45 seconds per frame processing.[8]. In this model, we use this advantage of

YOLO along with the concepts of non-maximum suppression [2] which help remove redundant, low confidence bounding boxes and threshold values which help qualify bounding boxes only above a certain confidence level. These values are standards set by the YOLO algorithm.

The working of YOLO is given as follows: An input image is divided into a S × S grid and this grid is particularly responsible for detection.[5] Each of these grid cells predict corresponding bounding boxes and confidence scores for each of the boxes. If no object exists in that cell, the confidence scores should be zero. If an image is detected, then the confidence score must be equal to the intersection over union (IOU) between the predicted box and the ground truth.[4][6]. YOLO v3 makes prediction across 3 different scales. The detection layer is used make detection at feature maps of three different sizes, having strides 32, 16, 8 respectively. In YOLO v3 trained on COCO, number of bounding boxes is 3 and number of classes is 80. [3]. This makes the kernel size come to 1 x 1 x 255.

The prediction system in YOLO works as follow: The first detection is made by the 82nd layer.[8] For the first 81 layers, the image is down sampled by the network, such that the 81st layer has a stride of 32. If we have an image of 416 x 416, the resultant feature map would be of size 13 x 13. One detection is made here using the 1 x 1 detection kernel, giving us a detection feature map of 13 x 13 x 255.

Further, layers are up-sampled by a factor of 2 and concatenated with feature maps of a previous layers having identical feature map sizes. Another detection is now made at layer with stride 16. The same up-sampling procedure is repeated, and a final detection is made at the layer of stride 8. Then, the second detection is made by the 94th layer, yielding a detection feature map of 26 x 26 x 255.

The third and final detection is made at 106th layer, yielding feature map of size 52 x 52 x 255.
For an image of size 416 x 416, YOLO predicts ((52 x 52) + (26 x 26) + 13 x 13)) x 3 = 10647 bounding boxes. However, in case of our image, there's only one object or in other words a single class. Now to reduce the detections from 10647 to 1 because we only need single class detection non-maxima suppression is used.

IV.FEATURES

This model is customized to a particular class that aids in the accurate detection of humans with high confidence values. It makes use of the fastest single neural network – YOLO that ensures object detection and classification along with computation of required parameters are done within a single layer. This optimizes the task at hand which makes this model a desirable choice. It also makes sure to display all the results in a presentable and legible manner, hence, aids in ensuring the social-distancing guidelines are being followed and the crowd density is in accordance to the particular area's protocols.

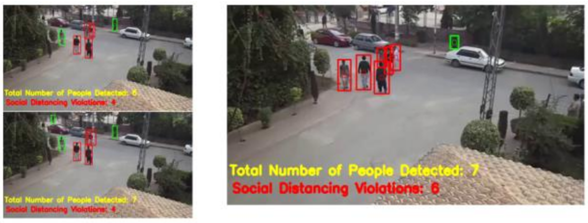
V.RESULTS

The people detection algorithm was carried out using you only look once (YOLO) algorithm to detect people and these results were computed to check distance and count total number of people. Yolo V3 is specifically used and the weights for yolov3 are used along with COCO names data set for object detection.

Results for the test were carried out for different scenario that are given below for different crowd density which are low, medium and high in scenario 1, scenario 2 and scenario 3 respectively.

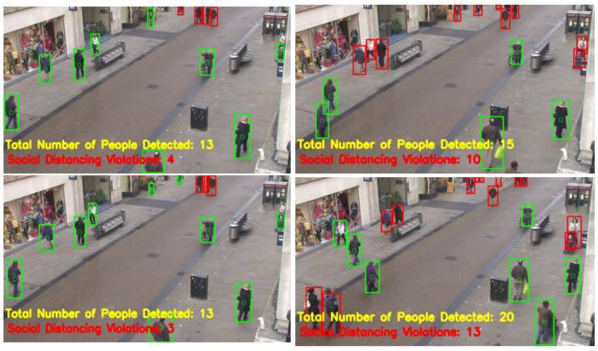
Scenario 1:

In this result it can be seen that the model we made is able to detect people and display the detected number of people in low density of people present in an area and also violation is also detected for the people in close proximity.


Total Number of People Detected: 6
Social Distancing Violations: 4
Total Number of People Detected: 7
Social Distancing Violations: 4
Total Number of People Detected: 7
Social Distancing Violations: 6

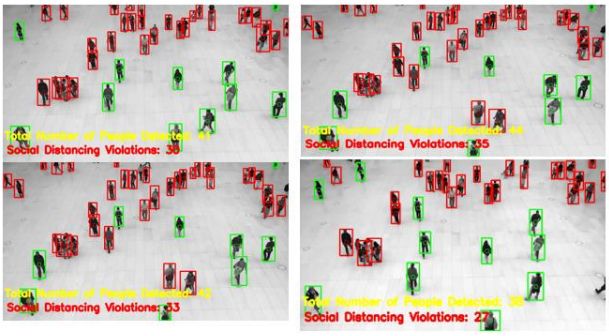
Scenario 2:

In this result it can be seen that the model we made is able to detect people and display the detected number of people in medium density of people present in an area and also violation is also detected for the people in close proximity.


Total Number of People Detected: 13
Social Distancing Violations: 4
Total Number of People Detected: 15
Social Distancing Violations: 10
Total Number of People Detected: 13
Social Distancing Violations: 3
Total Number of People Detected: 20
Social Distancing Violations: 13

Scenario 3:

In this result it can be seen that the model we made is able to detect people and display the detected number of people in relative high density of people present in an area and also violation is also detected for the people in close proximity.


Total Number of People Detected: 41
Social Distancing Violations: 30
Total Number of People Detected: 44
Social Distancing Violations: 35
Total Number of People Detected: 42
Social Distancing Violations: 33
Total Number of People Detected: 38
Social Distancing Violations: 27

VI.CONCLUSION AND DISCUSSIONS

In this paper, YOLO has been successfully implemented to detect people as well as a means of a social distance detector. This model is efficient in giving results comprising of high confidence values for detection of only a customized class ID which is that of a person. It has been tested for videos from different scenarios and different viewing angles, which make it unique to the application of maintaining a social distancing guideline in various areas according to their respective safety measures and protocols. For future scope and improvement of this model, the boundary conditions can be closely monitored in order to give accurate results in a highly congested and over-populated areas.

REFERENCES

[1] Jing Tao, Hongbo Wang, Xinyu Zhang, Xiaoyu Li, Huawei Yang, “An Object Detection System Based on YOLO in Traffic Scene” 2017 6th International Conference on Computer Science and Network Technology (ICCSNT)

[2] Liyan Yu, Xianqiao Chen, Sansan Zhou, “Research of Image Main Objects Detection Algorithm Based on Deep Learning”,2018 3rd IEEE International Conference on Image, Vision and Computing

[3] Matija Buric, Miran Pobar, Marina Ivasic-Kos, “Ball detection using Yolo and Mask R-CNN”,2018 International Conference on Computational Science and Computational Intelligence (CSCI)

[4] Pranav Adarsh, Pratibha Rathi, Manoj Kumar, “YOLO v3-Tiny: Object Detection and Recognition using one stage improved model”, 2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)

[5] Parth Goel, Sharnil Pandya, Ketan Kotecha, Dweepna Garg, Amit Ganatra, “A Deep Learning Approach for Face Detection using YOLO”, 2018 IEEE Punecon

[6] Wang Yang, Zheng Jiachun, “Real-time face detection based on YOLO”, 1st IEEE International Conference on Knowledge Innovation and Invention 2018

[7] Joseph Redmon , Santosh Divvala, Ross Girshick, Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection”, arXiv:1506.02640v5 [cs.CV] 9 May 2016

[8] Joseph Redmon, Ali Farhadi, “YOLO9000: Better, Faster, Stronger”, arXiv:1612.08242v1 [cs.CV] 25 Dec 2016

[9] Tsung-Yi Lin Michael Maire Serge Belongie Lubomir Bourdev Ross Girshick James Hays Pietro Perona Deva Ramanan C. Lawrence Zitnick Piotr Dollar, “Microsoft COCO: Common Objects in Context”, arXiv:1405.0312v3 [cs.CV] 21 Feb 2015

[10] Maha Hamdan Alotibia, Salma Kammoun Jarrayaa, Manar Salamah Alia, Kawthar Moriaa, “CNN-Based Crowd Counting Through IoT: Application For Saudi Public Places”, Peer-review under responsibility of the scientific committee of the 16th International Learning & Technology Conference 2019

[11] Shaya A. Alshaya, “Estimation of a high-dense crowd based on a Balanced Communication-Avoiding Support Vector Machine classifier”, IJCSNS International Journal of Computer Science and Network Security, VOL.20 No.6, June 2020

[12] Jianjie Yang, Jin Li, Ye He, “Crowd Density and Counting Estimation Based on Image Textural Feature”, JOURNAL OF MULTIMEDIA, VOL. 9, NO. 10, OCTOBER 2014.

[13] Aditya vora, Vinay chilaka, “fchd: fast and accurate head detection in crowded scenes”, arxiv:1809.08766v3 [cs.cv] 5 may 2019

[14] Sheng-Fuu Lin, Member, IEEE, Jaw-Yeh Chen, and Hung-Xin Chao, “Estimation of Number of People in Crowded Scenes Using Perspective Transformation” ieee transactions on systems, man, and cybernetics—part a: systems and humans, vol. 31, no. 6, november 2001.

[15] Mikel Rodriguez, Ivan Laptev, Josef Sivic, Jean-Yves Audibert, “Density-aware person detection and tracking in crowds. Proceedings of the IEEE International Conference on Computer Vision (2011)”.

[16] Licia Capodiferro, Luca Costantini, Federica Mangiatordi, Emiliano Pallotti, “ Data pre-processing to improve SVM video classification” June 2012

[17] Sonu Lamba and Neeta Nain, “ Multi-Source Approach for Crowd Density Estimation in Still Images” IEEE International Conference on 15 June 2017.

[18] H. Idrees, I. Saleemi, C. Seibert and M. Shah, "Multi-source Multi-scale Counting in Extremely Dense Crowd Images," 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 2013, pp. 2547-2554, doi: 10.1109/CVPR.2013.329.