SD-Measure: A Social Distancing Detector

Savyasachi Gupta[*], Rudraksh Kapil[†],
Goutham Kanahasabai[‡], Shreyas Srinivas Joshi[§], Aniruddha Srinivas Joshi[¶]
B. Tech., Department of Computer Science and Engineering

National Institute of Technology, Warangal, Telangana, India - 506004
gsavya10@gmail.com[*], rkapil@student.nitw.ac.in[†],
gauthamkanags@gmail.com[‡], sj841871@student.nitw.ac.in[§], aniruddha980@gmail.com[¶]

Abstract—The practice of social distancing is imperative to curbing the spread of contagious diseases and has been globally adopted as a non-pharmaceutical prevention measure during the COVID-19 pandemic. This work proposes a novel framework named SD-Measure for detecting social distancing from video footages. The proposed framework leverages the Mask R-CNN deep neural network to detect people in a video frame. To consistently identify whether social distancing is practiced during the interaction between people, a centroid tracking algorithm is utilised to track the subjects over the course of the footage. With the aid of authentic algorithms for approximating the distance of people from the camera and between themselves, we determine whether the social distancing guidelines are being adhered to. The framework attained a high accuracy value in conjunction with a low false alarm rate when tested on Custom Video Footage Dataset (CVFD) and Custom Personal Images Dataset (CPID), where it manifested its effectiveness in determining whether social distancing guidelines were practiced.

Index Terms—Social distancing, Person detection, Mask R-CNN, Centroid-based Object Tracking, COVID-19

I. Introduction

The onset of the COVID-19 pandemic has led to an increase in the importance of social distancing to intervene in the spread of the virus by curbing social interactions and maintaining physical distance. Social distancing can be defined as a non-pharmaceutical disease prevention and control intervention enforced to curb contact between those who are infected with a disease and those who are not, so as to stop or diminish the rate and extent of the transmission of the disease within a community. Eventually, this leads to a decrease in the spread of the disease and the fatalities caused by it. Centers for Disease Control and Prevention (CDC) safety guidelines dictate that a distance of at least 6 feet must be maintained between two individuals in both indoor and outdoor spaces [1].

Common social distancing measures include the closure of schools, universities, non-essential work-spaces, and public transportation. Researchers state that the rapid spread of COVID-19 is likely due to the movement of people with little to no symptoms [2], namely those who are unaware that they even contracted the virus. This is why the researchers have stressed upon the fact that social distancing is such a paramount containment measure.

Based on an Indian Council of Medical Research (ICMR) study [3], health officials have elucidated that the current R_o or R naught for the coronavirus infection is somewhere between 1.5 and 4, where R naught is a contagious virus'

basic reproductive number. If we take the R_o value to be 2.5, then a single infected patient can spread the disease to 406 other humans in roughly 30 days if appropriate social distancing norms are not followed. However, if social exposure is decreased by about 75% then a single infected patient will only be able to infect 2.5 people on average [3]. This emphasises how imperative social distancing can be in curbing the transmission of contagious agents.

Since the imposition of lockdowns and public restrictions in many parts of the world, the task of identifying whether a safe distance that complies with social distancing norms is presented as a new area of study in the field of computer vision. However, it is difficult for humans to estimate the distance between two people accurately from video footage alone, considering that cameras are usually angled in such a way as to maximise the field of vision. As a result, the proportions of objects in the images are distorted and make estimation inaccurate.

In this work, we propose a novel approach that allows a system to automatically detect when people are standing too close to each other, and to output an accurate estimate of the distance between them as well

II. Related works and Literature

In this section, we review some of the related methodologies that aim to tackle the same problem as our proposed approach. As social distancing has been in the limelight of being practiced in the recent months, there is a dearth of published work in this field. In [4], the author proposes an approach to estimating the distance between people to analyze whether social distancing is maintained. After obtaining the bounding box of people using YOLOv3, a width threshold is set for objects among which the distance is measured. By measuring the ratio of pixels to metres, the distance between two people in a given frame is approximated. However, this approach only calculates the distance between people without taking the individual distances of each person from camera into account. The major assumption is that each person is the same distance from the camera.

In a similar approach [5], the author uses the YOLOv3 object detection framework to identify people in a given frame. After computing a pairwise distance between the centroids of the detected bounding boxes of people, this value is compared to a predefined minimum pixel threshold. By mapping these

pixels to measurable units, detection of violations in social distancing norms are identified.

Punn *et al.* [6] proposed a methodology for monitoring COVID-19 social distancing with person detection and tracking. The framework utilizes the YOLOv3 object detection model to segregate people from the background region. A deepsort technique was used to track the identified people with the help of bounding boxes and assigned tracking IDs. After computing the pairwise vectorized L2 norm from the 3-D space obtained from the centroid coordinates and the dimensions of the bounding boxes, a violation index is proposed to identify the infringement of social distancing protocols. However, this approach doesn't measure the distance between each pair of people. As the authors themselves acknowledge, it could have achieved better precision if they had used Faster R-CNN instead of YOLOv3.

III. PROPOSED APPROACH

This section elucidates a novel framework called SD-Measure which aims to determine whether a set of people are following 'Social Distancing' guidelines of maintaining a minimum distance of 6 feet (or 1.8 metres) [1] when observed from video footage of a public area. The proposed framework performs four major tasks in the following order:

A) Person Detection
B) Person Tracking
C) Distance from Camera Estimation
D) Pairwise Social Distancing Estimation

The following sections explain the different stages involved in performing these tasks. The workflow for this proposed framework is illustrated in Figure 1.

A. Person Detection

We perform object detection using Mask R-CNN [7] to detect people in a video frame. We use Mask R-CNN as it extends and improves Faster R-CNN [8] by adding a mask branch and using Region of Interest (RoI) Align instead of RoI Pooling. RoI Align overcomes the issue of location misalignment existent in RoI Pooling. It achieves this by dividing the input proposals from the Region Proposal Network (RPN) into 'bins' using bilinear interpolation. Apart from the improvement in the core accuracy over Faster R-CNN, Mask R-CNN is able to create masks for the detected object allowing the authorities to distinguish individual persons in a frame more easily.

In this application, the Mask R-CNN model takes 30 video frames per second and outputs a list containing coordinates of the bounding boxes, masks, and detection scores for each detected person. The given step is illustrated in Figure 1 (a).

B. Person Tracking

After performing object detection, correctly detected people are retained based on their class IDs and detection scores after filtration from all the detected objects. This paves the way for the next task which is person tracking over the later time frames of the recording. Person tracking is accomplished by

utilising the Centroid Tracking algorithm [9]. This algorithm works by calculating the Euclidean separation between the centroids of identified people over successive frames as illustrated in Figure 1 (b).

The primary premise of tracking people is to be able to consistently determine whether a pair of people are following social distancing or not even if the object detection algorithm is unable to detect the previously detected person in intermediate frames. This ensures that the proposed framework is robust against fluctuations in the output of the object detection model.

C. Distance from Camera Estimation

Once the people in the frame have been correctly identified and tracked, our goal is to estimate the real distance of each person from the camera in metres using an effective mathematical principle known as 'Triangle Similarity'. This principle relies on a two-step process: (1) Focal length (F) determination using a located marker, and (2) Distance estimation of each person.

1) Focal length (F) determination using a located marker: This step uses a marker to calibrate and determine the focal length of the camera used in capturing the video frames. This is done by locating a marker object and relating its perceived dimensions with its known dimensions. To achieve this, we first locate any person and denote it as the marker. Then, we calculate its width in pixels by taking the difference of the horizontal coordinates of the bounding box's top-left corner (X_1) and bottom-right corner (X_2) and denote it as P as shown in Eq. 1.

$$P = X_2 - X_1 \qquad (1)$$

We estimate this marker person to be at a distance D from the camera either using its known distance or human perception. Then, we use a known value W which is the width of the person in metres.

To actually determine the value of W, we determine the elbow-to-elbow distance of a person since elbows extend the furthest from the body when standing and facing the camera. To determine this distance, we consider the average waist size of the population from which the video footage originates and account for some buffer for the placement of the arms of the person around his/her body. In case the origin of the video footage is unknown, we use other features of the marker person such as race or gender to determine the average waist size.

Once the known width W, distance from camera D, and the width in pixels P of the marker person have been tabulated, we calculate the focal length F of the camera as shown in Eq. 2.

$$F = \frac{D \times P}{W} \qquad (2)$$

2) Distance estimation of each person: Once we have determined the focal length F of the camera used for capturing the given video footage, we can estimate the real distance of a person from the camera Y based on the width of its bounding box in pixels P and the known width W (as determined previously) using Eq. 3.
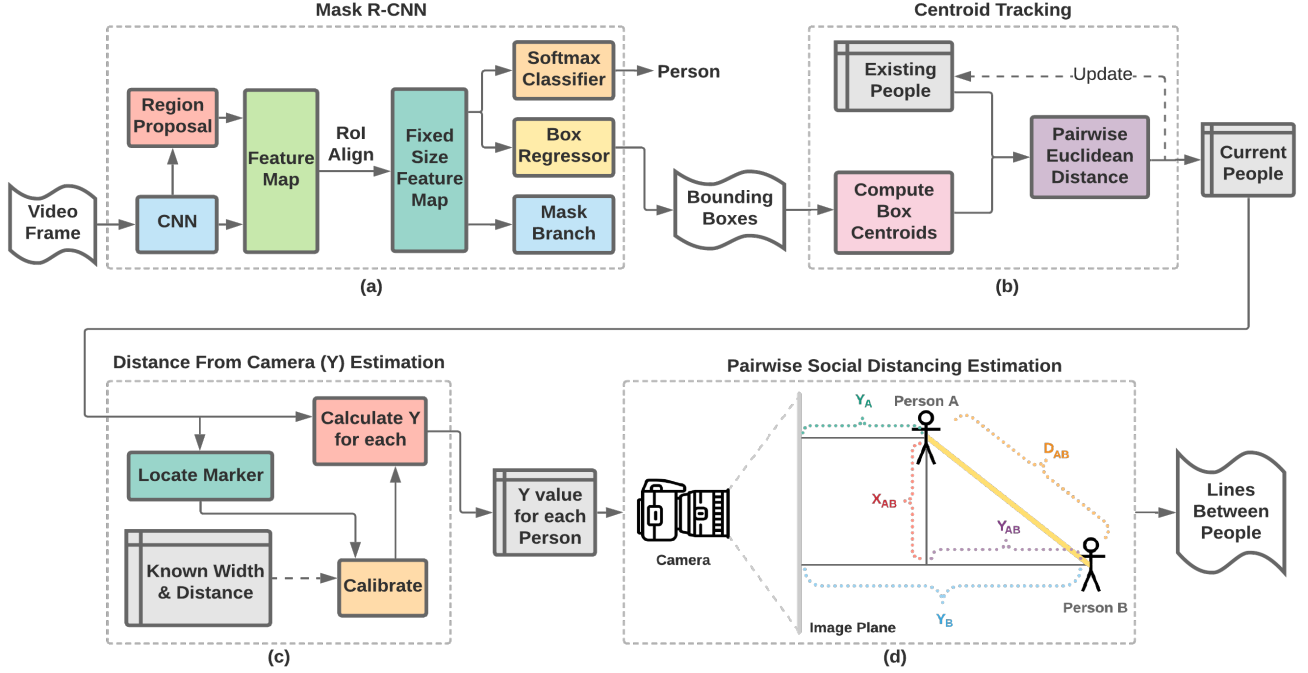
Fig. 1: Workflow of SD-Measure Framework

$$Y = \frac{F \times W}{P} \qquad (3)$$

This distance from the camera Y is subsequently calculated for every tracked person in the frame. The procedure for this step of the proposed framework is illustrated in Figure 1 (c).

D. Pairwise Social Distancing Estimation

After we have estimated the real distance of each tracked person from the camera, our next objective is to determine the person-to-person distance between each each pairwise-distinct set of people. This equates to $_nC_2$ number of distinct pairs where n is the number of tracked people. The following steps explain the calculation of the real distance between a pair of tracked people which is also illustrated in Figure 1 (d). For explanation purposes, we'll denote the two people in a pair as Person A and Person B.

Firstly, we retrieve the estimation of real distance calculated in the previous step and denote them as Y_A and Y_B for the two people respectively. Then, we calculate the absolute difference in the distance from the image plane between the two people and denote it as Y_{AB} using Eq. 4.

$$Y_{AB} = |Y_B - Y_A| \qquad (4)$$

Secondly, we determine the horizontal distance between the two people by finding the absolute difference in the horizontal coordinates of their centroids, which are denoted as X_A and

X_B respectively. The absolute difference is denoted as X_{AB} and is calculated as shown by Eq. 5.

$$X_{AB} = |X_B - X_A| \qquad (5)$$

A key point to note is that the distance Y_{AB} is calculated in metres whereas the distance X_{AB} is calculated in pixels. Hence, the next imperative step is to convert the units of the distance X_{AB} into metres.

To achieve this, we first calculate the widths of both Person A and Person B using Eq. 1 and denote them as P_A and P_B respectively. Then, we calculate the average of these widths P_{AB} using Eq. 6.

$$P_{AB} = \frac{|P_A + P_B|}{2} \qquad (6)$$

We calculate this average width P_{AB} in order to normalize the discrepancies in individual widths when determining our next measure which is 'pixels per metre' denoted as PPM. We determine the PPM based on the average width P_{AB} and the known width W (obtained from Section III-C) using Eq.7.

$$PPM = \frac{P_{AB}}{W} \qquad (7)$$

Once we have determined the PPM, we calculate the absolute difference in the horizontal distance in metres X'_{AB} using Eq.8.

$$X'_{AB} = \frac{X_{AB}}{PPM} \qquad (8)$$

After we have determined the absolute difference in distance from the camera Y_{AB} in metres and the absolute difference in the horizontal distance X'_{AB} in metres, we calculate the centroid-to-centroid distance in metres between Person A and Person B. We use the Pythagorean Theorem to obtain the real distance D_{AB} in metres as shown in Eq. 9.

$$D_{AB} = \sqrt{(X'_{AB})^2 + (Y_{AB})^2} \tag{9}$$

This is the distance in metres estimated between a pair of people to determine whether 'Social Distancing' guidelines [1] are being followed. If this distance is less than 1.8 metres, a line is drawn connecting the centroids of the two people in the pair to visualize people who are in violation of the given guidelines when observed from video footage of a public area. Visualization of the results obtained by the proposed approach is depicted in Figure 2.

Fig. 2: Visualisation of the results obtained by SD-Measure framework

IV. EXPERIMENTAL EVALUATION

In this section, we discuss the dataset used for conducting this study and the results obtained by the proposed approach. The experiments were conducted on Google Colab [10] with Intel(R) Xeon(R) 2.00 GHz CPU, NVIDIA Tesla T4 GPU, 16 GB GDDR6 VRAM and 13 GB RAM. All programs were written in *Python* - 3.6 and utilised *OpenCV* - 4.2.0, *Keras* - 2.3.0 and *TensorFlow* - 2.2.0.

A. Dataset Used

The following datasets were used in the development of the proposed framework:

1) MS COCO: The Microsoft Common Objects in Context (MS COCO) dataset [11] is a well-known dataset that has annotations for instance segmentation and bounding boxes used to evaluate how well object detection and image segmentation models perform. The dataset has been utilized to perform training on the Mask R-CNN [7] framework on the 'person' class.

2) Custom Video Footages Dataset (CVFD): This comprises of a collection of footages of public places from multiple geographical locations, compiled from YouTube. The videos capture the movement of people in public areas after the imposition of various safety rules and regulations in wake of the COVID-19 pandemic. This dataset contains videos having different camera angles, varying illumination conditions, noise, and an average frames per second (FPS) of 30. Figure 3 illustrates a few sample videos present in this dataset.

3) Custom Personal Images Dataset (CPID): We created our own dataset by taking images of two people, namely Person 1 and Person 2. Person 1 was used as a reference point and Person 2 was relocated in fixed positions with known distance from the reference point. The distance between the reference point and the camera was also known and varied at fixed distances. The primary object of this dataset was to determine the efficacy of the proposed framework in determining the true distance between people. The various fixed positions and distances from the camera with respect to the reference position is illustrated in Figure 4.
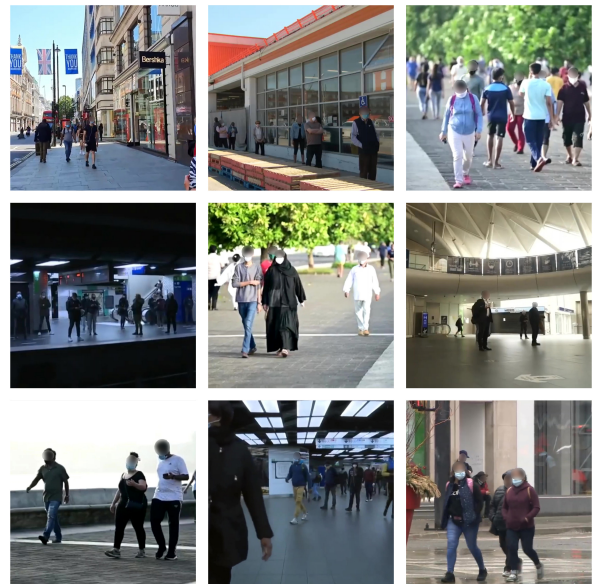

Fig. 3: Video samples from the CVFD dataset

B. Experimental Results and Statistics

We have evaluated the proposed approach in three parts: person detection using Mask R-CNN, detection of social
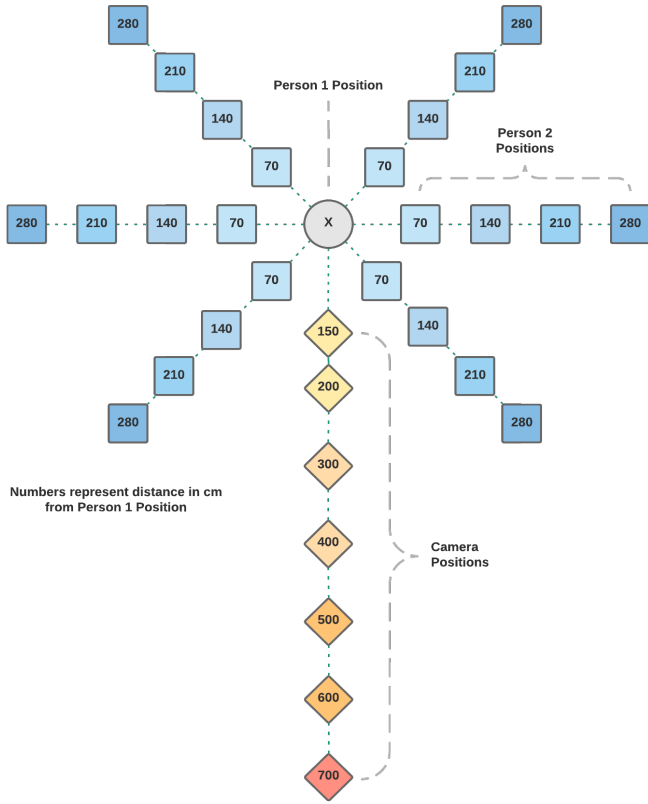
Fig. 4: Layout for CPID dataset

Precision	97.44%
Recall	90.39%

TABLE I: Evaluation result for Mask R-CNN tested on CVFD dataset

Precision	86.06%
Recall	83.99%
False Alarm Rate	3.27%
Accuracy	94.26%

TABLE II: Evaluation result for proposed framework for social distancing lines drawn when tested on CVFD dataset

detection algorithm is to detect people with a high precision even with a trade-off of a slightly reduced recall. This is because we are using a tracking algorithm which can track people over several frames to allow the detection algorithm to have false negatives without affecting the efficacy of the overall framework. On the contrary, a lower precision would translate to a greater number of false positives which would lead to incorrect detections being tracked for several frames and affecting the effectiveness of the framework.

We evaluated the detection of social distancing lines based on four parameters: precision, recall, false alarm rate (Eq. 12), and accuracy (Eq. 13). As explained in Section III, we draw a line between a pair of people if the distance between them has been estimated to be less than 1.8 metres. We analyzed each frame of the CVFD dataset (refer to Section IV-A) and determined whether a line has been correctly drawn or not for each pair-wise distinct set of people. We considered a True Positive (TP) when a line was perceived to be correctly drawn, a False Positive (FP) when a line was incorrectly drawn, a True Negative (TN) when a line was correctly not drawn, and a False Negative (FN) when a line was incorrectly not drawn for a pair of people. Since it is not possible to identify the actual distance between each pair of people from a video footage, we used human perception versus the detections made by the proposed framework to determine the aforementioned values. To ensure robustness and to minimize errors due to bias and other random errors, we used a majority voting system among the five authors to determine whether a line should be drawn or not between each pair of people in a given frame and then compared with the output of the proposed framework. The results have been tabulated in Table II.

We evaluated the efficacy of the measure of the distance between a pair of people based on two statistical parameters: standard deviation (Eq. 14) and percent error (Eq. 15). Since the proposed framework estimates the actual distance between each pair of people (as explained in Section III), we used statistics to evaluate the output of the proposed framework when tested on CPID dataset (refer to Section IV-A).

We used standard deviation to determine the framework's precision, which is a measure of how close the measurements are with each other. We calculated the standard deviation for

distancing lines, and the efficacy of the measure of the distance between a pair of people.

$$Precision = \frac{TP}{TP + FP} \times 100\% \qquad (10)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \qquad (11)$$

$$False\ Alarm\ Rate = \frac{FP}{TN + FP} \times 100\% \qquad (12)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \qquad (13)$$

$$Standard\ deviation\ (S) = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n - 1}} \qquad (14)$$

$$\%\ error = \frac{|true\ distance - estimated\ distance|}{true\ distance} \times 100\% \qquad (15)$$

We estimate the performance of the person detection using Mask R-CNN [7] based on two parameters: precision (Eq. 10), and recall (Eq. 11), which are calculated from the number of True Positives (TP), False Positives (FP), and False Negatives (FN). These parameters have been determined based on the results when tested on the CVFD dataset (as explained in Section IV-A) and are shown in Table I.

Since every part of the image where we do not predict a person is considered a negative, the measurement of True Negative is considered ineffectual. The focus of the person

various fixed distances between Person 1 and Person 2 at each of the fixed distances between the camera and Person 1 in the layout which is shown in Figure 4. The standard deviation values have been represented as stacks for each distance between the camera and Person 1 as shown in Figure 5. It can be observed that the proposed approach shows greater precision in the estimated distance when the reference person is at a close-medium distance from the camera.
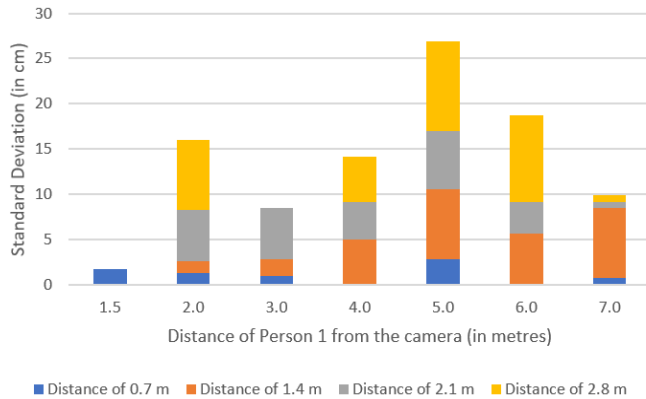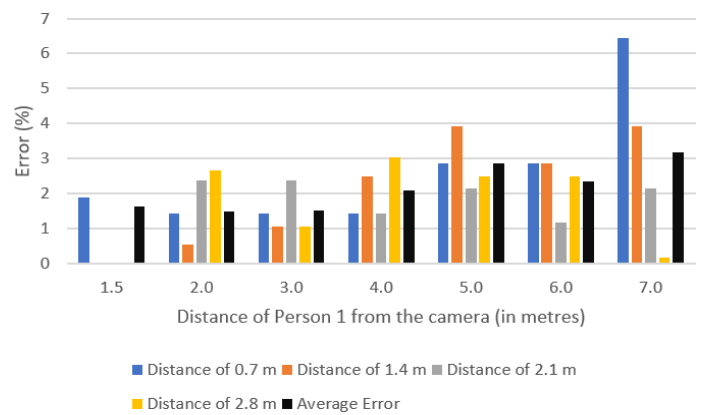

Fig. 6: Percent error of the output of the proposed framework when tested on the CPID dataset. The key represents the distance between Person 1 and Person 2 in the given layout.

in turn correct estimation of social distancing, which can be addressed in future work.

Fig. 5: Standard deviation of the output of the proposed framework when tested on the CPID dataset. The key represents the distance between Person 1 and Person 2 in the given layout.

Similarly, we used percent error to determine the framework's accuracy, which is a measure of how close the measurements are to the accepted or true value. We calculated the percent error for various fixed distances between Person 1 and Person 2 at each of the fixed distances between the camera and Person 1 and is shown in Figure 6. We have also represented the average error for each distance between Person 1 and the camera as well. It can be observed that as the distance between the camera and the reference person (i.e. Person 1) is increased, the error increases. This demonstrates that the proposed approach has a higher accuracy for determining the distance between people who are located at a closer range with respect to the camera.

V. CONCLUSION AND FUTURE WORKS

In this work, a novel framework for detecting social distancing from video footage is proposed. By leveraging state-of-the-art object detection models for identifying people followed by an authentic distance from camera and a pairwise distance estimation algorithms, we deduce whether social distancing guidelines are being followed. The resulting approach is effective and was evaluated on a custom video footages dataset obtained for this research, as evidenced by the high accuracy value along with good precision and recall values. The low false alarm rate further validates the potency of the proposed approach. The approach can be fine-tuned for better performance according to the specific environment in consideration. In addition, large obstacles obstructing the field of view of the cameras may affect the tracking of people and

VI. ACKNOWLEDGEMENT

REFERENCES

[1] "Social distancing," Jul 2020. [Online]. Available: https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/social-distancing.html

[2] W. H. Organization, "Modes of transmission of virus causing covid-19: implications for ipc precaution recommendations: scientific brief, 29 march 2020," World Health Organization, Technical documents, 2020.

[3] S. Mandal, T. Bhatnagar, N. Arinaminpathy, A. Agarwal, A. Chowdhury, M. Murhekar, R. Gangakhedkar, and S. Sarkar, "Prudent public health intervention strategies to control the coronavirus disease 2019 transmission in India: A mathematical model-based approach," *Indian Journal of Medical Research*, vol. 151, no. 2, pp. 190–199, 2020.

[4] M. Jain, "Finding social distance using yolo and opencv," Apr 2020. [Online]. Available: https://medium.com/@mayur87545/finding-social-distance-using-yolo-and-opencv-6ac2595d3a27

[5] A. Rosebrock, "Opencv social distancing detector," Jun 2020. [Online]. Available: https://www.pyimagesearch.com/2020/06/01/opencv-social-distancing-detector/

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

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2980–2988.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *in IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, June 2017.

[9] J. C. Nascimento, A. J. Abrantes, and J. S. Marques, "An algorithm for centroid-based tracking of moving objects," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 6, March 1999, pp. 3305–3308.

[10] "Google colab." [Online]. Available: https://colab.research.google.com/

[11] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2014.