A Vision-based Social Distancing and Critical Density Detection System for COVID-19

SCHOLARONE™
Manuscripts

A Vision-based Social Distancing and Critical Density Detection System for COVID-19

Abstract—**Social distancing (SD) is an effective measure to prevent the spread of the infectious COronaVIrus Disease 2019 (COVID-19). However, a lack of spatial awareness may cause unintentional violations of this new measure. Against this backdrop, we propose an active surveillance system to slow down the spread of COVID-19 by warning individuals in a region-of-interest. Our contribution is twofold. First, we introduce a vision-based real-time system that can detect SD violations and send non-intrusive audio-visual cues using state-of-the-art deep learning models. Secondly, we define a novel critical social density value and show that SD violation occurrence chance can be held near zero if the pedestrian density is kept under this value. The proposed system is also ethically fair: it does not record data nor target individuals. No human supervisor is present during operation, and our open-source code [1] is publicly accessible. The proposed system was evaluated across real-world datasets.**

Index Terms—**convolutional neural network, social distancing, pedestrian detection, linear regression**

I. INTRODUCTION

Social distancing is an effective measure [1] against the novel COronaVIrus Disease 2019 (COVID-19) pandemic. However, the general public is not used to keep an imaginary safety bubble around themselves. An automatic warning system [2], [3], [4], [5] can help and augment the perceptive capabilities of individuals.

Deploying such an active surveillance system requires serious ethical considerations and smart system design. The first challenge is privacy [6]. If data is recorded and stored, the privacy of individuals may be violated intentionally or unintentionally. As such, the system must be real-time without any data storing capabilities.

Second, the detector must not discriminate. The safest way to achieve this is by building an AI-based detection system. Removing the human out of the detection loop may not be enough – the detector must also be design free. Domain-specific systems with hand-crafted feature extractors may lead to malign designs. A connectionist machine learning system, such as a deep neural network without any feature-based input space, is much fairer in this sense, with one caveat: the distribution of the training data must be fair.

Another critical aspect is being non-intrusive. Individuals should not be targeted directly by the warning system. A non-alarming audio-visual cue can be sent to the vicinity of the social distancing breach to this end.

The system must be open-sourced. This is crucial for establishing trust between the active surveillance system and society.

Against this backdrop, we propose a non-intrusive augmentative AI-based active surveillance system for sending

omnidirectional visual/audio cues when a social distancing breach is detected. The proposed system uses a pre-trained deep convolutional neural network (CNN) [7], [8] to detect individuals with bounding boxes in a given monocular camera frame. Then, detections in the image domain are transformed into real-world bird's-eye view coordinates. If a distance smaller than the threshold is detected, the system emits a non-alarming audio-visual cue. Simultaneously, the system measures social (pedestrian) density. If the social density is larger than a critical threshold, the system sends an advisory inflow modulation signal to prevent overcrowding.

Our main contributions are:

- A novel vision-based real-time social distancing and critical social density detection system.
- Definition of critical social density and a statistical approach to measure it.
- Measurements of social distancing and critical density statistics of common crowded places such as the New York Central Station, an indoor mall, and a busy town center in Oxford.

II. RELATED WORK

Social distancing for COVID-19. COVID-19 has caused severe acute respiratory syndromes around the world since December 2019 [9]. Recent work showed that social distancing is an effective measure to slow down the spread of COVID-19 [1]. Social distancing is defined as keeping a minimum of 2 meters (6 feet) apart from each individual to avoid possible contact. Further analysis [10] also suggests that social distancing has substantial economic benefits. COVID-19 may not be completely eliminated in the short term, but an automated system that can help monitoring and analyzing social distancing measures can greatly benefit our society.

Pedestrian detection. Pedestrian detection can be viewed as a sub-task of generic object detection or as a specific task of detecting pedestrians only. A detailed survey of 2D object detectors and the corresponding datasets, metrics, and fundamentals can be found in [11]. Another survey [12] focuses on deep learning based approaches for both the generic object detectors and the pedestrian detectors. Generally speaking, state-of-the-art detectors are divided into two categories. One category is two-stage detectors. Most of them are based on R-CNN [13], [7], which starts with region proposals and then performs the classification and bounding box regression. The other category is one-stage detectors. Prominent models include YOLO [14], [8], SSD [15], and EfficientDet [16]. The detectors can also be classified as anchor-based [13], [7], [14], [8], [15], [16] or anchor-free approaches [17], [18].

---

[1]For the purpose of anonymous review, we intentionally hide the link to the code.
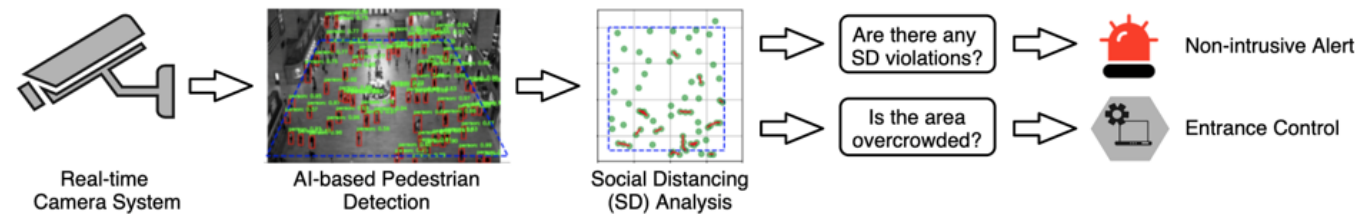
Fig. 1. Overview of the proposed system. Our system is real-time and does not record data. An audio-visual cue is emitted each time an individual breach of social distancing is detected. We also make a novel contribution by defining a critical social density value ρ_c for measuring overcrowding. Entrance into the region-of-interest can be modulated with this value.

The major difference between them is whether to use a set of predefined bounding boxes as candidate positions for the objects. Evaluating these approaches was usually done using the datasets of Pascal VOC [19] and MS COCO [20]. The accuracy and real-time performance of these approaches are good enough for deploying pre-trained models for social distancing detection.

Social distancing monitoring. Emerging technologies can assist in the practice of social distancing. A recent work [2] has identified how emerging technologies like wireless, networking, and artificial intelligence (AI) can enable or even enforce social distancing. The work discussed possible basic concepts, measurements, models, and practical scenarios for social distancing. Another work [3] has classified various emerging techniques as either human-centric or smart-space categories, along with the SWOT analysis of the discussed techniques. A specific social distancing monitoring approach [4] that utilizes YOLOv3 and Deepsort was proposed to detect and track pedestrians followed by calculating a violation index for non-social-distancing behaviors. The approach is interesting but results do not contain any statistical analysis. Furthermore, there is no implementation or privacy-related discussion other than the violation index. Social distancing monitoring is also defined as a visual social distancing (VSD) problem in [5]. The work introduced a skeleton detection based approach for inter-personal distance measuring. It also discussed the effect of social context on people's social distancing and raised the concern of privacy. The discussions are inspirational but again it does not generate solid results for social distancing monitoring and leaves the question open.

Very recently, several prototypes utilizing machine learning and sensing technologies have been developed to help social distancing monitoring. Landing AI [21] has proposed a social distancing detector using a surveillance camera to highlight people whose physical distance is below the recommended value. A similar system [22] was deployed to monitor worker activity and send real-time voice alerts in a manufacturing plant. In addition to surveillance cameras, LiDAR based [23] and stereo camera based [24] systems were also proposed, which demonstrated that different types of sensors besides surveillance cameras can also help.

The above systems are interesting, but recording data and sending intrusive alerts might be unacceptable by some people. On the contrary, we propose a non-intrusive warning system with softer omnidirectional audio-visual cues. In addition, our system evaluates critical social density and modulates inflow into a region-of-interest.

III. PRELIMINARIES

Object detection with deep learning. Object detection in the image domain is a fundamental computer vision problem. The goal is to detect instances of semantic objects that belong to certain classes, e.g., humans, cars, buildings. Recently, object detection benchmarks have been dominated by deep Convolutional Neural Network (CNN) models [13], [7], [14], [8], [15], [16]. For example, top scores on MS COCO [20], which has over 123K images and 896K objects in the training-validation set and 80K images in the testing set with 80 categories, have almost doubled thanks to the recent breakthrough in deep CNNs.

These models are usually trained by supervised learning, with techniques like data augmentation [25] to increase the variety of data.

Model Generalization. The generalization capability [26] of the state-of-the-art is good enough for deploying pre-trained models to new environments. For 2D object detection, even with different camera models, angles, and illumination conditions, pre-trained models can still achieve good performance.

Therefore, a pre-trained state-of-the-art deep learning based pedestrian detector can be directly utilized for the task of social distancing monitoring.

IV. METHOD

We propose to use a fixed monocular camera to detect individuals in a region of interest (ROI) and measure the inter-personal distances in real time without data recording. The proposed system sends a non-intrusive audio-visual cue to warn the crowd if any social distancing breach is detected. Furthermore, we define a novel critical social density metric and propose to advise not entering into the ROI if density is higher than this value. The overview of our approach is given in Figure 1, and the formal description starts below.

A. Problem formulation

We define a scene at time t as a 6-tuple $S = (\mathbf{I}, A_0, d_c, c_1, c_2, U_0)$, where $\mathbf{I} \in R^{H \times W \times 3}$ is an RGB image captured from a fixed monocular camera with height H and width W. $A_0 \in R$ is the area of the ROI on the ground plane in real world and $d_c \in R$ is the required minimum physical distance. c_1 is a binary control signal for sending a non-intrusive audio-visual cue if any inter-pedestrian distance is less than d_c. c_2 is another binary control signal for controlling the entrance to the ROI to prevent overcrowding. Overcrowding is detected with our novel definition of critical social density ρ_c. ρ_c ensures social distancing violation occurrence probability stays lower than U_0. U_0 is an empirically decided threshold such as 0.05.

Problem 1. Given S, we are interested in finding a list of pedestrian pose vectors $P = (\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_n)$, $\mathbf{p} \in R^2$, in real-world coordinates on the ground plane and a corresponding list of inter-pedestrian distances $D = (d_{1,2}, \cdots, d_{1,n}, d_{2,3}, \cdots, d_{2,n}, \cdots, d_{n-1,n})$, $d \in R^+$. n is the number of pedestrians in the ROI. Also, we are interested in finding a critical social density value ρ_c. ρ_c should ensure the probability $p(d > d_c | \rho < \rho_c)$ stays over $1 - U_0$, where we define social density as $\rho := n/A_0$.

Once Problem 1 is solved, the following control algorithm can be used to warn/advise the population in the ROI.

Algorithm 1. If $d \leq d_c$, then a non-intrusive audio-visual cue is activated with setting the control signal $c_1 = 1$, otherwise $c_1 = 0$. In addition, If $\rho > \rho_c$, then entering the area is not advised with setting $c_2 = 1$, otherwise $c_2 = 0$.

Our solution to Problem 1 starts below.

B. Pedestrian detection in the image domain

First, pedestrians are detected in the image domain with a deep CNN model trained on a real-world dataset:

$$\{T_i\}_k = f_{\text{cnn}}(\mathbf{I}). \tag{1}$$

$f_{\text{cnn}} : \mathbf{I} \to \{T_i\}_n$ maps an image $\mathbf{I}$ into n tuples $T_i = (l_i, \mathbf{b}_i, s_i), \forall i \in \{1, 2, \cdots, n\}$. n is the number of detected objects. $l_i \in L$ is the object class label, where L, the set of object labels, is defined in f_{cnn}. $\mathbf{b}_i = (\mathbf{b}_{i,1}, \mathbf{b}_{i,2}, \mathbf{b}_{i,3}, \mathbf{b}_{i,4})$ is the associated bounding box (BB) with four corners. $\mathbf{b}_{i,j} = (x_{i,j}, y_{i,j})$ gives pixel indices in the image domain. The second sub-index j indicates the corners at top-left, top-right, bottom-left, and bottom-right respectively. s_i is the corresponding detection score. Implementation details of f_{cnn} is given in Section V-A.

We are only interested in the case of $l =$ 'person'. We define $\mathbf{p}'_i$, the pixel pose vector of person i, with using the middle point of the bottom edge of the BB:

$$\mathbf{p}'_i := \frac{(\mathbf{b}_{i,3} + \mathbf{b}_{i,4})}{2}. \tag{2}$$

C. Image to real-world mapping

The next step is obtaining the second mapping function $h :$ $\mathbf{p}' \to \mathbf{p}$. h is an inverse perspective transformation function that maps $\mathbf{p}'$ in image coordinates to $\mathbf{p} \in R^2$ in real-world
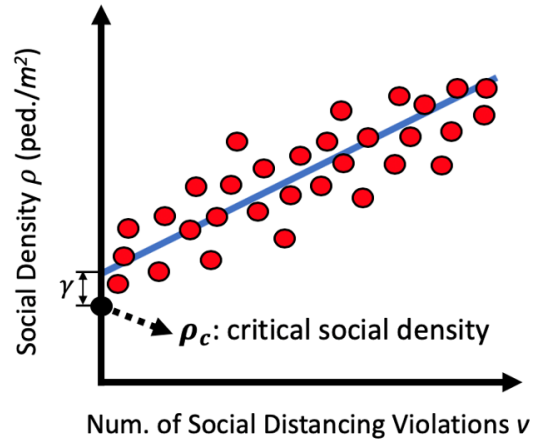

Fig. 2. Obtaining critical social density ρ_c. Keeping ρ under ρ_c will drive the number of social distancing violations v towards zero with the linear regression assumption.

coordinates. $\mathbf{p}$ is in 2D bird's-eye-view (BEV) coordinates by assuming the ground plane $z = 0$. We use the following well-known inverse homography transformation [27] for this task:

$$\mathbf{p}^{\text{bev}} = \mathbf{M}^{-1} \mathbf{p}^{\text{im}}, \tag{3}$$

where $\mathbf{M} \in R^{3 \times 3}$ is a transformation matrix describing the rotation and translation from world coordinates to image coordinates. $\mathbf{p}^{\text{im}} = [p'_x, p'_y, 1]$ is the homogeneous representation of $\mathbf{p}' = [p'_x, p'_y]$ in image coordinates, and $\mathbf{p}^{\text{bev}} = [p_x^{\text{bev}}, p_y^{\text{bev}}, 1]$ is the homogeneous representation of the mapped pose vector.

The world pose vector $\mathbf{p}$ is derived from $\mathbf{p}^{\text{bev}}$ with $\mathbf{p} = [p_x^{\text{bev}}, p_y^{\text{bev}}]$.

D. Social distancing detection

After getting $P = (\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_n)$ in real-world coordinates, obtaining the corresponding list of inter-pedestrian distances D is straightforward. The distance $d_{i,j}$ for pedestrians i and j is obtained by taking the Euclidean distance between their pose vectors:

$$d_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|. \tag{4}$$

And the total number of social distancing violations v in a scene can be calculated by:

$$v = \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} I(d_{i,j}), \tag{5}$$

where $I(d_{i,j}) = 1$ if $d_{i,j} < d_c$, otherwise 0.

E. Critical social density estimation

Finally, we want to find a critical social density value ρ_c that can ensure the social distancing violation occurrence
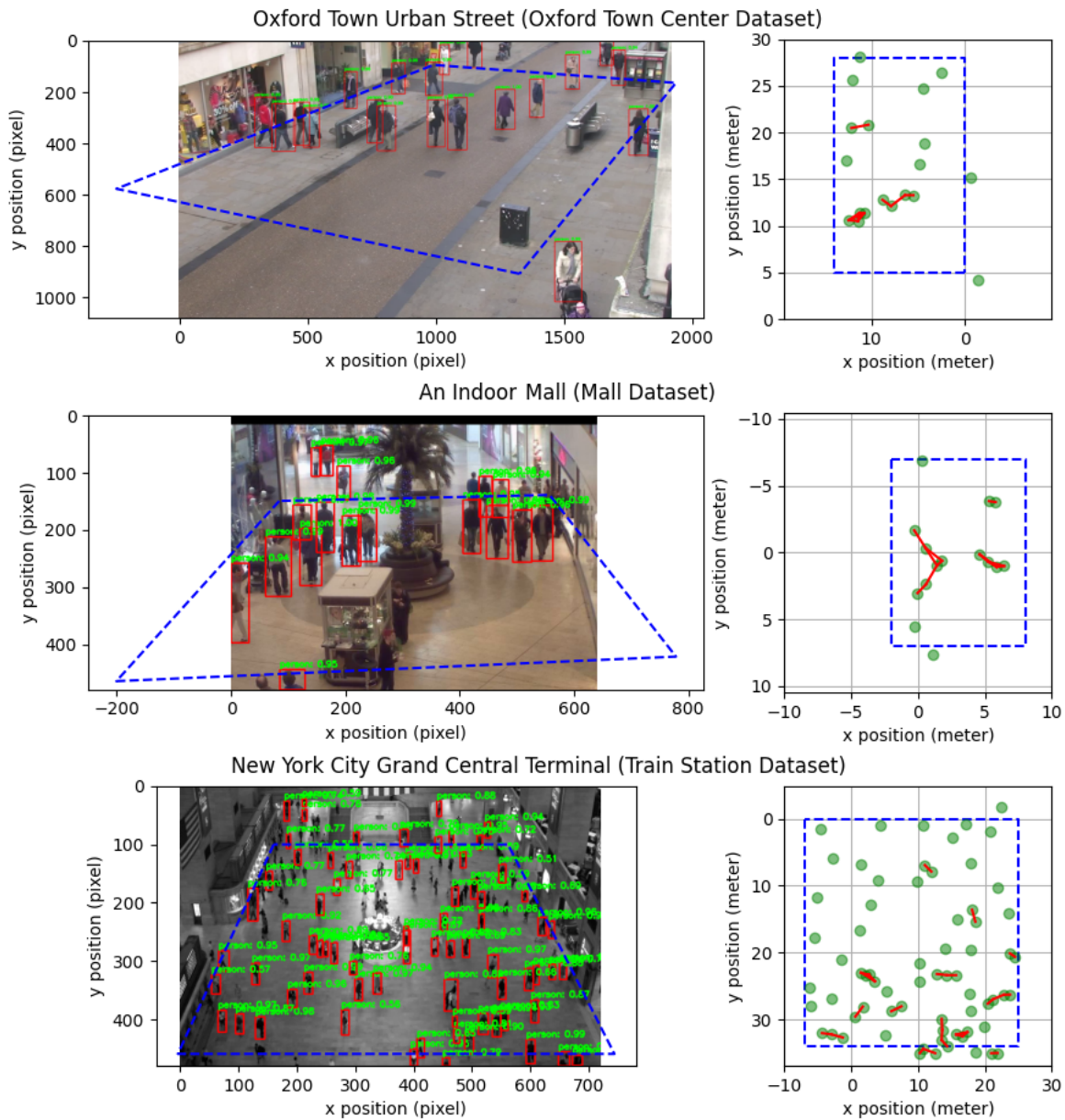
Fig. 3. Illustration of pedestrian detection using Faster R-CNN [7] and the corresponding social distancing.

probability stays below U_0. It should be noted that a trivial solution of $\rho_c = 0$ will ensure $v = 0$, but it has no practical use. Instead, we want to find the maximum critical social density ρ_c that can still be considered safe.

To find ρ_c, we propose to conduct a simple linear regression using social density ρ as the independent variable and the total number of violations v as the dependent variable:

$$v = \beta_0 + \beta_1 \rho + \epsilon, \qquad (6)$$

where $\boldsymbol{\beta} = [\beta_0, \beta_1]$ is the regression parameter vector and ϵ is the error term which is assumed to be normal. The regression

model is fitted with the ordinary least squares method. Fitting this model requires training data. However, once the model is learned, data is not required anymore. After deployment, the surveillance system operates without recording data.

Once the model is fitted, critical social density is identified as:

$$\rho_c = \rho_{\text{lb}}^{\text{pred}}, \qquad (7)$$

where $\rho_{\text{lb}}^{\text{pred}}$ is the lower bound of the 95% prediction interval $(\rho_{\text{lb}}^{\text{pred}}, \rho_{\text{ub}}^{\text{pred}})$ at $v = 0$, as illustrated in Figure 2.

Keeping ρ under ρ_c will keep the probability of social distancing violation occurrence near zero with the linear
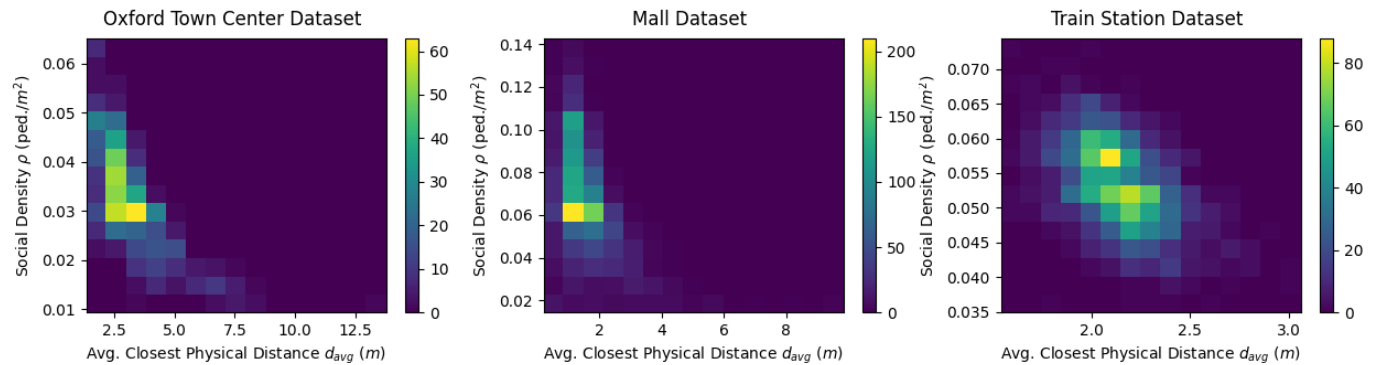
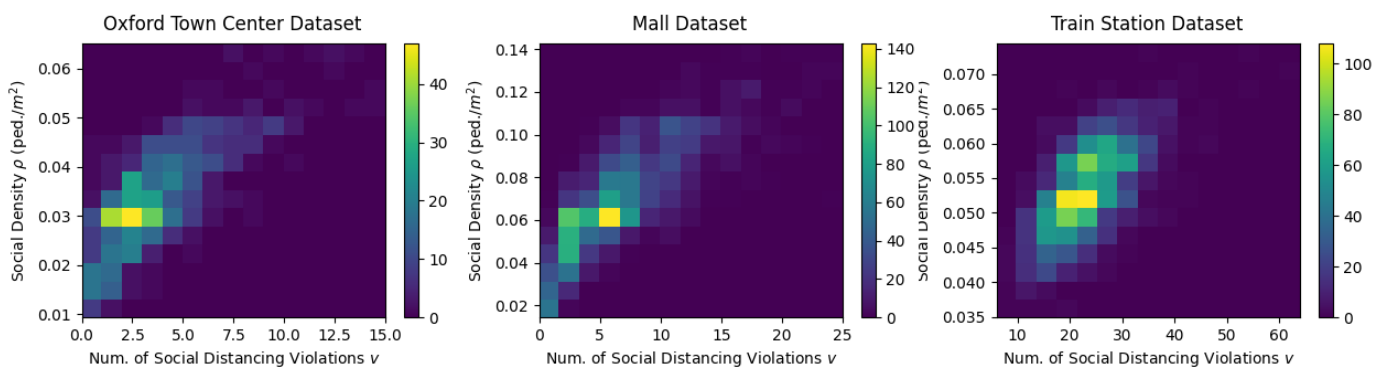Fig. 4. 2D histograms of the social density ρ versus the average closest physical distance d_{avg}.

Fig. 5. 2D histograms of the social density ρ versus the number of social distancing violations v. From the histograms we can see a linear relationship with positive correlation.

regression assumption.

V. EXPERIMENTS

We conducted 3 case studies to evaluate the proposed method. Each case utilizes a different pedestrian crowd dataset. They are the Oxford Town Center Dataset (an urban street) [28], the Mall Dataset (an indoor mall) [29], and the Train Station Dataset (New York City Grand Central Terminal) [30]. Table I shows detailed information about these datasets.

A. Implementation details

The first step was finding the perspective transformation matrix $\mathbf{M}$ for each dataset. For the Oxford Town Center

TABLE I
INFORMATION OF EACH PEDESTRIAN DATASET.

Dataset	FPS	Resolution	Duration
Oxford Town Ctr.	25	1920×1080	5 mins
Mall	~ 1	640×480	33 mins
Train Station	25	720×480	33 mins

Dataset, we directly used the transformation matrix available on its official website. For the Train Station Dataset, we found the floor plan of NYC Grand Central Terminal and measured the exact distances among some key points that were used for calculating the perspective transformation. For the Mall Dataset, we first estimated the size of a reference object in the image by comparing it with the width of detected pedestrians and then utilized the key points of the reference object to calculate the perspective transformation.

The second step was applying the pedestrian detector on each dataset. The experiments were conducted on a regular PC with an Intel Core i7-4790 CPU, 32GB RAM, and an Nvidia GeForce GTX 1070Ti GPU running Ubuntu 16.04 LTS 64-bit operating system. Once the pedestrians were detected, their positions were converted from the image coordinates into the real-world coordinates.

The last step was conducting social distancing monitoring and finding the critical density ρ_c. Only the pedestrians within the ROI were considered. The statistics of the social density ρ, the inter-pedestrian distances $d_{i,j}$, and the number of violations v were recorded over time. The analysis of statistics is described in the following section.
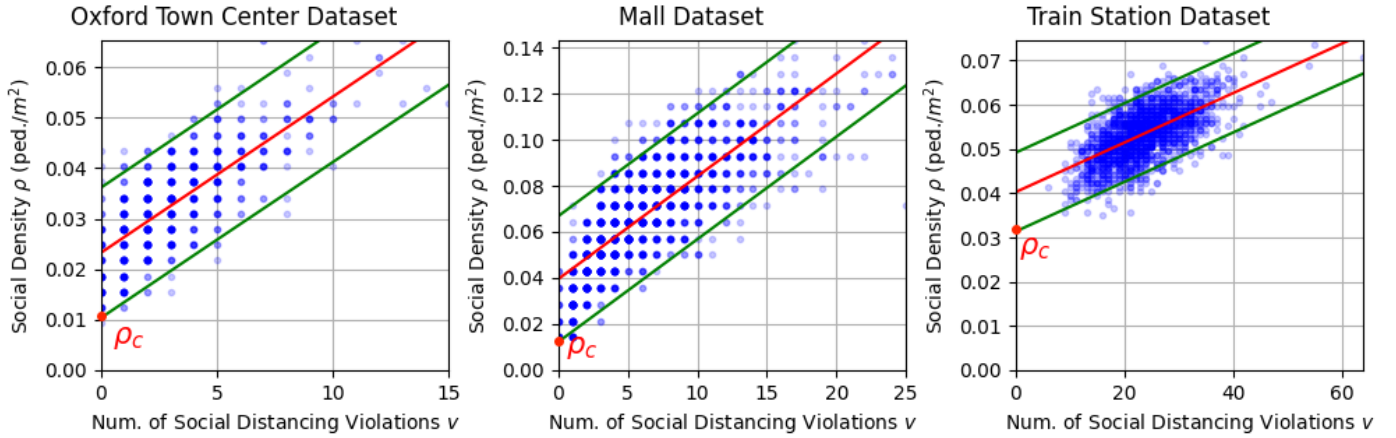
Fig. 6. Linear regression (red line) of the social density ρ versus number of social distancing violations v data. Green lines indicate the prediction intervals. The critical social densities ρ_c are the x-intercepts of the regression lines.

Method	mAP (%)	Inference Time (sec)
Faster R-CNN [7]	42.1-42.7	0.145 / 0.116 / 0.108
YOLOv4 [8]	41.2-43.5	0.048 / 0.050 / 0.050

Dataset	Intercept β_0	Critical Density ρ_c
Oxford Town Ctr.	0.0233	0.0104
Mall	0.0396	0.0123
Train Station	0.0403	0.0314

VI. RESULTS

A. Pedestrian Detection

We experimented with two different deep CNN based object detectors: Faster R-CNN and YOLOv4. Figure 3 shows the pedestrian detection results using Faster R-CNN [7] and the corresponding social distancing in world coordinates. The detector performances are given in Table II. As can be seen in the table, both detectors achieved real-time performance. For detection accuracy, we provide the results of MS COCO dataset from the original works [7], [8].

B. Social Distancing Monitoring

For pedestrian i, the closest physical distance is $d_i^{\min} = \min(d_{i,j})$, $\forall j \neq i \in \{1, 2, \cdots n\}$. Based on $d_i^{\min}$, we further calculated two metrics for social distancing monitoring: the minimum closest physical distance $d_{\min} = \min(d_i^{\min}), \forall i \in \{1, 2, \cdots .n\}$ and the average closest physical distance $d_{\mathrm{avg}} = \frac{1}{n} \sum_{i=1}^n d_i^{\min}$. Figure 7 shows the change of $d_{\min}$ and d_{avg} as time evolves. They are compared with the social density ρ. From the figure we can see that when ρ is relatively low, both the $d_{\min}$ and d_{avg} are relatively high, for example, $t = 85s$ in Train Station Dataset, $t = 80s$ in Mall Dataset, and $t = 100s$ in Oxford Town Center Dataset. This shows a clear negative correlation between ρ and d_{avg}. The negative correlation is further visualized as 2D histograms in Figure 4.

C. Critical Social Density

To find the critical density ρ_c, we first investigated the relationship between the number of social distancing violations v and the social density ρ in 2D histograms, as shown in Figure 5. As can be seen in the figure, v increases with an increase in ρ, which indicates a linear relationship with a positive correlation.

Then, we conducted the simple linear regression, using the regression model of equation (6), on the data points of v versus ρ. The skewness values of ρ for Oxford Town Center Dataset, Mall Dataset, and Train Station Dataset are 0.25, 0.16, and -0.07, respectively, indicating the distributions of ρ are symmetric. This satisfies the normality assumption of the error term in linear regression. The regression result is displayed in figure 6. The critical density ρ_c was identified as the lower bound of the prediction interval at $v = 0$.

Table III summaries the identified critical densities ρ_c as well as the intercepts β_0 of the regression models. The obtained critical density values for all datasets are similar. They also follow the patterns of the data points as illustrated in Figure 6. This verified the effectiveness of our method.
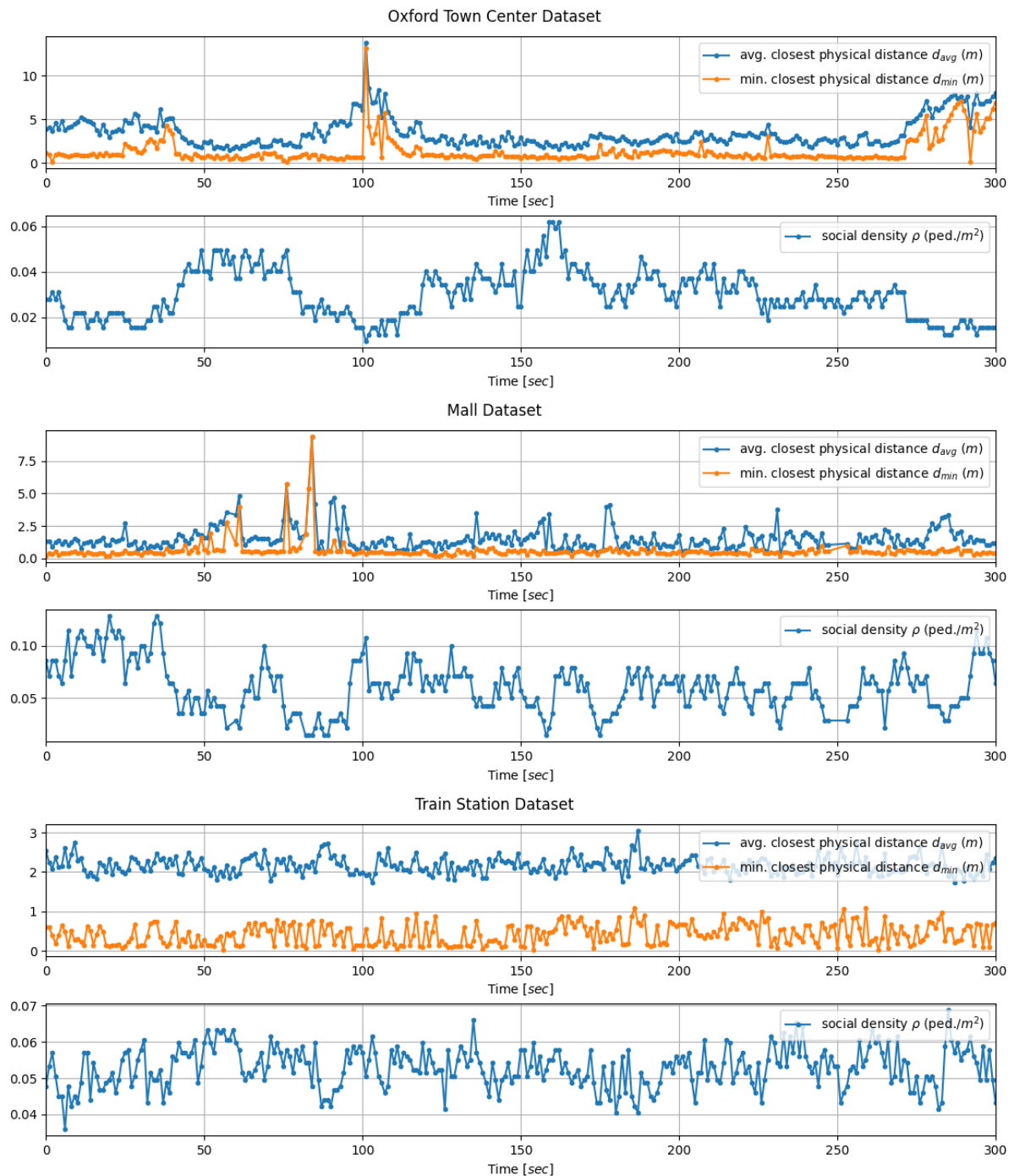
Fig. 7. The change of minimum closest physical distance $d_{\min}$, average closest physical distance d_{avg}, and the social density ρ over time.

VII. CONCLUSION

This work proposed an AI and monocular camera based real-time system to monitor social distancing. In addition, our system utilizes the proposed critical social density value to avoid overcrowding by modulating inflow to the ROI. The proposed approach was verified using 3 different pedestrian crowd datasets.

There are some missing detections in the Train Station Dataset, as in some areas the pedestrian density is extremely high and occlusion happens. However, after some qualitative analysis, we concluded that most of the pedestrians were captured and the idea of finding critical social density is still valid.

Pedestrians who belong to a group were not considered as a group in the current work, which can be a future direction. Nevertheless, one may argue that even individuals who have close relationships should still try to practice social distancing in public areas.

REFERENCES

[1] C. Courtemanche, J. Garuccio, A. Le, J. Pinkston, and A. Yelowitz, "Strong social distancing measures in the united states reduced the covid-19 growth rate: Study evaluates the impact of social distancing measures on the growth rate of confirmed covid-19 cases across the united states." *Health Affairs*, pp. 10–1377, 2020.

[2] C. T. Nguyen, Y. M. Saputra, N. Van Huynh, N.-T. Nguyen, T. V. Khoa, B. M. Tuan, D. N. Nguyen, D. T. Hoang, T. X. Vu, E. Dutkiewicz *et al.*, "Enabling and emerging technologies for social distancing: A comprehensive survey," *arXiv preprint arXiv:2005.02816*, 2020.

[3] S. Agarwal, N. S. Punn, S. K. Sonbhadra, P. Nagabhushan, K. Pandian, and P. Saxena, "Unleashing the power of disruptive and emerging technologies amid covid 2019: A detailed review," *arXiv preprint arXiv:2005.11507*, 2020.

[4] 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," *arXiv preprint arXiv:2005.01385*, 2020.

[5] M. Cristani, A. Del Bue, V. Murino, F. Setti, and A. Vinciarelli, "The visual social distancing problem," *arXiv preprint arXiv:2005.04813*, 2020.

[6] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–7.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[8] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[9] D. S. Hui, E. I. Azhar, T. A. Madani, F. Ntoumi, R. Kock, O. Dar, G. Ippolito, T. D. Mchugh, Z. A. Memish, C. Drosten *et al.*, "The continuing 2019-ncov epidemic threat of novel coronaviruses to global health—the latest 2019 novel coronavirus outbreak in wuhan, china," *International Journal of Infectious Diseases*, vol. 91, pp. 264–266, 2020.

[10] M. Greenstone and V. Nigam, "Does social distancing matter?" *University of Chicago, Becker Friedman Institute for Economics Working Paper*, no. 2020-26, 2020.

[11] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[12] Z.-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.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[14] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[16] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10781–10790.

[17] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 734–750.

[18] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE international conference on computer vision*, 2019, pp. 9627–9636.

[19] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[21] LandingAI, *Landing AI Creates an AI Tool to Help Customers Monitor Social Distancing in the Workplace*, 2020 (accessed May 3, 2020). [Online]. Available: https://landing.ai/landing-ai-creates-an-ai-tool-to-help-customers-monitor-social-distancing-in-the-workplace/

[22] P. Khandelwal, A. Khandelwal, and S. Agarwal, "Using computer vision to enhance safety of workforce in manufacturing in a post covid world," *arXiv preprint arXiv:2005.05287*, 2020.

[23] J. Hall, *Social Distance Monitoring*, 2020 (accessed June 1, 2020). [Online]. Available: https://levelfivesupplies.com/social-distance-monitoring/

[24] StereoLabs, *Using 3D Cameras to Monitor Social Distancing*, 2020 (accessed June 20, 2020). [Online]. Available: https://www.stereolabs.com/blog/using-3d-cameras-to-monitor-social-distancing/

[25] B. Zoph, E. D. Cubuk, G. Ghiasi, T.-Y. Lin, J. Shlens, and Q. V. Le, "Learning data augmentation strategies for object detection," *arXiv preprint arXiv:1906.11172*, 2019.

[26] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *arXiv preprint arXiv:1611.03530*, 2016.

[27] D. A. Forsyth and J. Ponce, *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002.

[28] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *CVPR 2011*. IEEE, 2011, pp. 3457–3464.

[29] K. Chen, C. C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting." in *BMVC*, vol. 1, no. 2, 2012, p. 3.

[30] B. Zhou, X. Wang, and X. Tang, "Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2871–2878.