Knowledge Engineering and Data Science (KEDS)
Vol 5, No 1, December 2022, pp. 17–26

pISSN 2597-4602
eISSN 2597-4637

Social Distancing Monitoring System using Deep Learning

Amelia Ritahani Ismail*, Nur Shairah Muhd Affendy, Ahsiah Ismail, Asmarani Ahmad Puzi

Department of Computer Science, Kulliyyah of ICT, International Islamic University Malaysia
53100 Kuala Lumpur, Malaysia
*amelia@iium.edu.my**
**corresponding author*

ARTICLE INFO

Article history:
Received 19 October 2021
Revised 25 December 2021
Accepted 15 August 2022
Published online 7 November 2022

Keywords:
Deep learning
Object detection
Social distancing

ABSTRACT

COVID-19 has been declared a pandemic in the world by 2020. One way to prevent COVID-19 disease, as the World Health Organization (WHO) suggests, is to keep a distance from other people. It is advised to stay at least 1 meter away from others, even if they do not appear to be sick. The reason is that people can also be the virus carrier without having any symptoms. Thus, many countries have enforced the rules of social distancing in their Standard Operating Procedure (SOP) to prevent the virus spread. Monitoring the social distance is challenging as this requires authorities to carefully observe the social distancing of every single person in a surrounding, especially in crowded places. Real-time object detection can be proposed to improve the efficiency in monitoring the social distance SOP inspection. Therefore, in this paper, object detection using a deep neural network is proposed to help the authorities monitor social distancing even in crowded places. The proposed system uses the You Only Look Once (YOLO) v4 object detection models for the detection. The proposed system is tested on the MS COCO image dataset with a total of 330,000 images. The performance of mean average precision (mAP) accuracy and frame per second (FPS) of the proposed object detection is compared with Faster Region-based Convolutional Neural Network (R-CNN) and Multibox Single Shot Detector (SSD) model. Finally, the result is analyzed among all the models.

This is an open access article under the CC BY-SA license
(https://creativecommons.org/licenses/by-sa/4.0/).

I. Introduction

During the COVID-19 pandemic, social distancing is a feasible approach to reducing the virus's spread [1]. Social distancing means keeping a safe space between people to prevent spreading a contagious disease. During the pandemic, most countries have enforced social distancing toward their people as one of the Standard Operating procedures (SOP). Hence, social distancing has become a new normal during the pandemic. However, some individuals do not take this action seriously or are unaware of their surroundings, making it harder for the authorities to observe the SOP [2][3][4].

At the early phase of the Movement Control Order (MCO) in Malaysia, National Security Council reported that although 92% have complied with the SOP during MCO, most citizens have failed to observe the social distancing [3]. Furthermore, during the Recovery Movement Control Order (RMCO), the numbers of SOP violators kept increasing, and some premises were compounded because of social distancing violations offenses [4]. The current SOP inspections were made manually by the authorities. This requires the workforce to do the observations that involve police officers, The People's Volunteer Corps (RELA), and the city council [4]. With the help of real-time object detection using a deep neural network, the SOP can be monitored remotely by the authorities hence, improving the efficiency of the inspections, especially in a crowded place.

A social distancing detection system requires an object detection system that can detect a person automatically. The previous study has investigated various deep learning algorithms based on Convolutional Neural Network (CNN) to be used for the object detection system such as Faster R-CNN, SSD and YOLOv4.

https://doi.org/10.17977/um018v5i12022p17–26
©2022 Knowledge Engineering and Data Science | W : http://journal2.um.ac.id/index.php/keds | E : keds.journal@um.ac.id
This is an open access article under the CC BY-SA license (https://creativecommons.org/licenses/by-sa/4.0/)

Many research works have been done to promote social distancing during the pandemic. In object detection applications, person detection is crucial for detecting social distancing between them. A new network structure, YOLO-R, was introduced by Lan *et al.* (2018) to improve the network structure of the YOLOv2 algorithm in detecting pedestrians by altering the network structure [5]. Three Passthrough layers are added to the YOLOv2 network to extract the shallow layer pedestrian features, and the shallow layer features extracted from the Route layer of the original algorithm are improved from the 16^{th} layer to the 12^{th} layer, combining shallow layer features with deep layer features to extract more fine-grained features. The dataset used for the model is the INRIA dataset which consists of 2416 data for training and 1126 data for testing. The comparison between YOLOv2 and YOLO-R was shown, and YOLO-R has proven to perform better than the YOLOv2 model. The precision for YOLOv2 is 97.37%, YOLO-R's is 98.56%, and both algorithms' recall is 89.33% and 91.21%, respectively. The missed rate of the YOLO-R network model is also lower than the YOLOv2 model, which is 10.05%, and 11.29% for YOLO v2 network model.

A study presented the monitoring of COVID-19 social distancing with person detection and tracking using YOLO v3 for person detection and Deepsort for person tracking [6]. The YOLO v3 object detection model was used to distinguish the persons and Deepsort to track the identified people and assigned IDs. Apart from YOLO v3, Faster R-CNN and SSD algorithms are also being used to compare the performance of people detection in the real-time video surveillance system. As mentioned, the Deepsort technique is used to track custom objects in the video and is an extension of SORT (Simple Real-Time Tracker). For effective tracking, the Kalman filter and the Hungarian algorithm are used and also include Mahalanobis distance to calculate the distance for social distancing between people. The distance calculation is computed based on 3D feature space obtained using centroid coordinates and a bounding box. The dataset used for the model is from the open image dataset (OID) repository by the Google open-source community consisting of 800 images divided into an 8:2 ratio for training and testing. The model was then tested on surveillance footage of the Oxford Town Center. Between Faster R-CNN, SSD, and YOLO v3, YOLO v3 has achieved the best results for object detection with balanced mAP and FPS scores. Faster R-CNN works on region proposals to create boundary boxes to indicate objects and has shown a better accuracy but has slow processing of FPS, making it unsuitable for real-time detection. The SSD algorithm has improved the FPS of Faster R-CNN by using multiscale features and default boxes in a single process for real-time processing. The results for the mentioned model are 96.9% mAP with 3 FPS for Faster R-CNN, 69.1% mAP with 10 FPS for SSD, and 84.6% and 23 FPS for YOLOv3.

The further study proposed an AI-based real-time social distancing detection and warning system using a monocular camera and deep learning-based real-time object detectors to measure social distancing during the pandemic [7]. A pre-trained deep convolutional neural network (CNN) is being used to detect the individuals who are Faster R-CNN and YOLOv4 using MS COCO dataset. The distance between the pedestrian is calculated using Euclidean distance after getting the image to real-world mapping coordinates. Three experiments were conducted in three different places using Oxford Town Center Dataset (an urban street), Mall Dataset (an indoor mall), and Train Station Dataset (New York City Grand Central Terminal). Both detectors that are using Faster R-CNN and YOLOv4 algorithms achieve the real-time performance shown by mAP in three places with 42.1%-42.7% and 41.2%-43.5% for Faster R-CNN and YOLOv4, respectively.

This research is proposed to develop a social distancing monitoring system based on a deep neural network, evaluate the model performance and develop a monitoring system for the authorities to observe the social distancing in a specific place. The object detection will be using object detection algorithms which are Faster R-CNN, SSD and YOLOv4 to detect the object (Person) using Microsoft Common Objects in Context (MS COCO) Dataset [8]. Next, the system will calculate the distance between two persons and identify the number of violations in one place. The outcome expected for this research is to determine which detection algorithm performs better in monitoring social distancing for the authorities.

The research develops a social distancing detection and monitoring system based on a deep neural network. Furthermore, it evaluates the object detection model performance and compares the detection model performance for a social distancing monitoring system. This research is to investigate a detection algorithm that is suitable for social distancing monitoring to assist the authorities in observing social distancing on-premises. The systems will detect the social distancing

between the people and show the level of the violation on the premises to prepare the authorities if any action should be taken. It also can improve the efficiency of the authorities' inspections and encourage people to abide by the rules. The remainder of this article is prepared as follows: The methodology section describes the approach taken to monitor social distancing. Results and discussion section present the result for each algorithm in detecting acceptable social distancing practice and analyzes the result obtained. The conclusion section draws the present work's conclusion and future in improvising the present investigation.

II. Methods

Figure 1 illustrates the methodology flowchart for the research. This research is proposed to assist the authority in observing social distancing during the pandemic. YOLOv4, Faster R-CNN, and SSD deep neural network algorithm are being applied for person detection with MS COCO dataset and the results are being analyzed to find the most suitable algorithm for the social distancing system.

A. Data Preparation

MS COCO image dataset [8] is used to evaluate the performance of the proposed system. It is large-scale object detection, segmentation, and captioning dataset. MS COCO dataset consists of 330,000 images with 80 object categories, including 64,115 images for the person category. For this research, 10,000 images were randomly selected from COCO person images. The images were downloaded using COCO Application Programming Interface (API) with a filtered category (person) to get the images. COCO API assists in loading, parsing, and visualizing annotations in its dataset. Figure 2 shows examples taken from the MS COCO dataset for the person category.

The image in the dataset has its annotations provided that can be extracted with COCO API. For Faster R-CNN and SSD, the annotations were converted into PASCAL VOC format, while YOLOv4 model, the annotations need to be converted to YOLO format to fit into the model. The annotation type used for the model is bounding boxes. In COCO format, the bounding box was

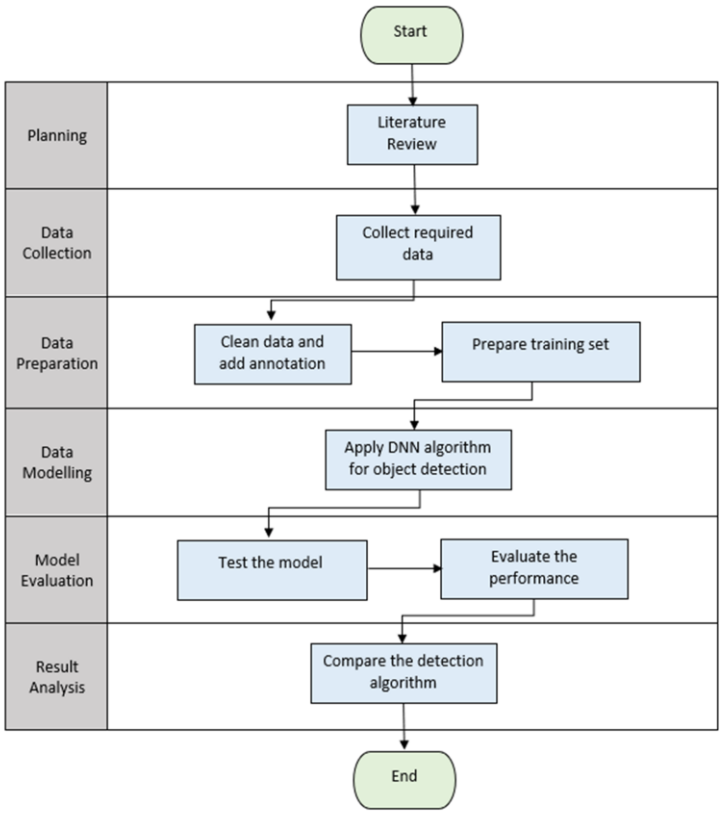

Fig. 1. Methodology flowchart based on machine learning process

Fig. 2. Examples of COCO person images dataset

Fig. 3. Examples of bounding boxes annotations

displayed as [x, y, width, height], where x and y are the top-left edges of the bounding box, followed by its width and height. While YOLO format is displayed as <object-class> <x> <y> <width> <height> where x and y are the center of the bounding boxes followed by its width and height. Figure 3 depicts the annotations example of the bounding boxes for the person images.

B. Data Modeling

The main idea of R-CNN is composed of two steps. Girshick *et al.* (2014) proposed using selective search to extract the regions in the image to identify the region of interest (ROI) and extract the features from each region for classification [9]. Girshick *et al.* (2015) proposed a new improvement of R-CNN called Fast R-CNN after determining some drawbacks from the previous R-CNN. The approach is similar to R-CNN, but instead of feeding the region proposals to CNN, the input image was fed to the CNN to generate a convolutional feature map and identify the region proposals [10].

Both R-CNN and Fast R-CNN use selective search to find the region proposals [9][10]. Therefore, Ren *et al.* (2016) [11] eliminate the selective search process and let the network learn the region proposals. Faster R-CNN is the improvement of Fast R-CNN comprising two modules. Based on Figure 4, the first module is a feature extraction network consisting of deep convolutional layers and the second is a Fast R-CNN detector based on the proposed regions in the first module. The second module contains two subnetworks: Region Proposal Network (RPN) and classifier. Using RPN in Faster R-CNN has improved the efficiency of the detection. It is a fully convolutional network that is trained, and predicts object boundaries and scores for each detection. In short, the second module is to generate object proposals followed by the classifier to predict the actual class of the object [11].

Despite using two modules in Faster R-CNN, SSD has no delegated region proposal network, and it predicts the classes directly from feature maps and uses small convolutional filters to predict. SSD is designed for real-time object detection. It applies multiscale features and default boxes to improve accuracy. Figure 5 represents that the VGG16 network is used to extract feature maps from the input image and applies 3×3 convolution filters for each cell to make predictions. Six additional convolutional layers, then follow it after the VGG16. Five of them are used for object detection, and six predictions are made using six layers [12].

Instead of selecting parts of an image for prediction, YOLO predicts classes and bounding boxes for the whole image in one run of the algorithm and is mainly used for real-time object detection. YOLO predicts the object based on the bounding boxes and class probabilities for the boxes that define whether an object is present or not. The general YOLO system consists of three steps. First, get the input image and divide it into grids. Second, run the convolutional network on the image to predict the bounding boxes and their class probabilities. Finally, it applies non-max suppression where it cleans the multiple detections by selecting the highest probability [13]. YOLOv2 was introduced to improve the initial YOLO detection by altering the layers in YOLO [14]. YOLOv3 is built on YOLOv2 with several improvements. On the other hand, it makes detections at three scales that give input dimensions by 32, 16 and 8. In YOLOv3, the detection is done by applying 1×1 detection kernels generated by the convolutional network on feature maps of three different sizes at three different places in the network [15].

The latest YOLO, YOLOv4 as shown in Figure 6, is an improved architecture of the previous YOLO version consisting of four blocks: Backbone, Neck, Dense Prediction, and Sparse Prediction. Backbone block refers to feature extraction architecture, and the Neck adds extra layers between

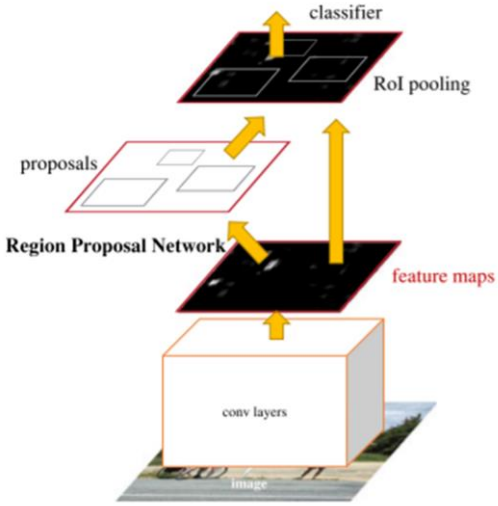

Fig. 4. Faster R-CNN model architecture [11]

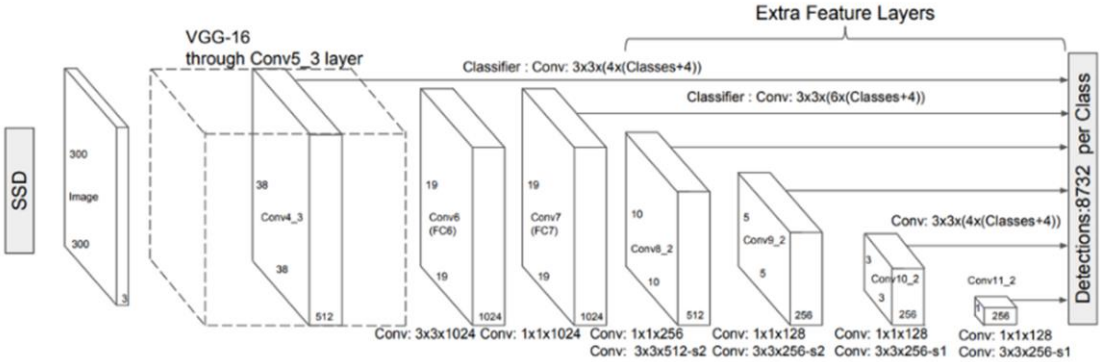

Fig. 5. SSD model architecture [12]

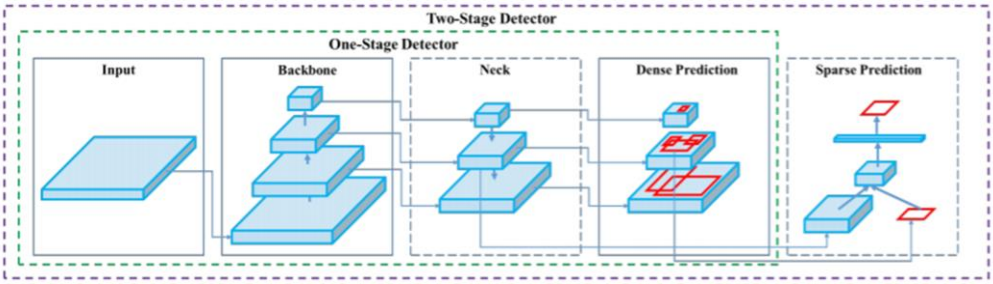

Fig. 6. YOLOv4 model architecture [16]

Table 1. The mAP and FPS for different object detection model [16]

Model	mAP	FPS	GPU
Faster R-CNN	59.2%	9.4	Pascal
SSD300	43.1%	43	Maxwell
SSD500	48.5%	22	Maxwell
YOLOv2	48.1%	40	Maxwell
YOLOv3	57.9%	20	Maxwell
YOLOv4	65.7%	23	Maxwell
YOLOV4	65.7%	33	Pascal

blocks. Head comprises Dense Prediction and Sparse Prediction to locate bounding boxes and classify what is inside each box [16].

A comparison of the speed and accuracy of object detectors on the MS COCO dataset is shown in Table 1 based on Bochkovskiy *et al*. [16]. Faster R-CNN has shown a good mAP value but with the lowest speed compared to other detectors. However, SSD has the fastest speed for 300×300 image resolutions with a higher mAP value than Faster R-CNN. The emergence of YOLOv4 with the highest mAP value with balanced speed makes it a better detector than others.

The algorithm used for person detection is YOLOv4, where the architecture of the network is imported from Darknet for model training. The platform for the training is Google Colab which has the below specifications:

- CPU: Intel(R) Xeon(R) CPU @ 2.20GHz
- GPU: Tesla T4
- RAM: 12GB

The model was taken from Tensorflow Object Detection API for transfer learning for Faster R-CNN and SSD. Faster R-CNN was trained using Inceptionv2 as the backbone. Inceptionv2 is the improvement of Inceptionv1 wherein the Inceptionv2 architecture and the two 3×3 convolutions replace the 5×5 convolution. This decreases computational time and thus increases computational speed because a 5×5 convolution is 2.78 more expensive than a 3×3 convolution. To sum up, using two 3×3 layers instead of 5×5 increases the performance of architecture [17].

However, SSD was trained using MobileNetv2 as the backbone for the algorithm. MobileNet is a streamlined architecture that uses depthwise separable convolutions to construct lightweight deep convolutional neural networks and provides an efficient model for mobile and embedded vision applications. As a lightweight deep neural network, MobileNet has fewer parameters and higher classification accuracy [18].

For YOLOv4, CSPDarknet53 serves as the backbone for this model [16]. It is a CNN and foundation for object detection that employs DarkNet-53. It divides the feature map of the base layer into two pieces using a Cross-Stage-Partial-connections network (CSPNet) technique [19] and then combines them using a cross-stage hierarchy. A split and merge method provides more gradient flow over the network. Hyperparameters were tuned based on the model, machine memory, and capability shown in Table 2.

Table 2. Hyperparameter tuning

Model	YOLOv4	Faster RCNN	SSD
Backbone	CSPDarknet53	Inceptionv2	MobileNetv2
Batch	64	1	24
Learning rate	0.001	0.0002	0.04
Iterations	6000	200000	160000
Momentum	0.949	0.9	0.9
Size	416×416	600×600	300×300

After the person detection training, the distance between two persons is calculated using the Euclidean Distance formula (Equation 1) to determine whether the minimum distance has been followed as in SOP guidelines. The points taken from the center of each bounding box detect the person.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

where d is the distance, x, y represent two points in Euclidean n-space, x_i, y_i determine the Euclidean vectors, starting from the origin of the space (initial point), and n defines the n-space

The system will detect whether there will be more than one person in the frame and calculate the distance between them. The social distance threshold is set at 40.0 pixels, equivalent to approximately 1 meter, assuming the relative scale ratio is 1:4000. The risk percentage is shown at the bottom left of the frame using Equation 2.

$$Risk = \frac{Total\ violated\ person\ (Red\ box)}{Total\ person\ detected} \times 100 \quad (2)$$

III. Results and Discussion

The results of models have been analyzed with the Intersection over Union (IoU) threshold of 0.5, following the standard requirement set by MS COCO Benchmark Challenge [8]. Based on Figure 7, IoU is calculated by dividing the intersection area with the union area between ground truth and predicted bounding boxes. For object detection, the precision and recall are calculated using IoU. If IoU is bigger or equal to 0.5, the object is classified as True Positive (TP). If IoU is lower than 0.5, it is considered a False Positive (FP). False Negative (FN) is classified when the ground truth is present, but the model failed to detect the object [20].

The model is evaluated by calculating the Precision (3), Recall (4), F1-score (5), and mAP for accuracy and FPS for model performance. The general definition of AP is finding the area under the precision-recall curve, which can be calculated using (6). The mAP score is calculated by taking the average of AP over all classes for an IoU threshold of 0.5. Since this research contains only one

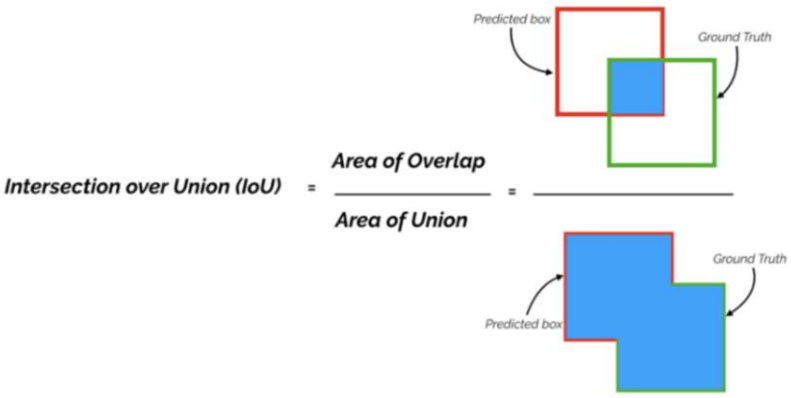

Fig. 7. Calculation of IoU

class (Person), the AP and mAP will be the same.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (3)$$

$$Recall = \frac{True\ Positive}{Ture\ Positive + False\ Negative} \quad (4)$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

$$AP = \frac{1}{11} \sum_{r \in \{0.0,\ldots,1.0\}} p_{interp}{}^{(r)} \quad (6)$$

In model testing, YOLOv4 has achieved a mAP score of 82.47% for 2,000 testing images, while Faster R-CNN is 66.10%, and SSD is 41.34%. The performance is tested on the video, and YOLOv4 can detect around 14~17 FPS while Faster R-CNN is 7~8 FPS and SSD is 49~54 FPS. Table 3 shows the model performance for person detection using different deep learning algorithms for the person COCO dataset.

A good detector for object detection should give the best balance of speed and accuracy needed for the application [21]. Based on the result, Faster RCNN has the lowest FPS and a better mAP score than SSD. However, SSD has the best speed compared to the other models. To conclude, YOLOv4 has been proven to be the best detection model, which it shows a balance of accuracy and speed for detection and has been applied to the monitoring system.

The model is then tested on the test video from the Oxford Town Centre dataset [22]. The sample

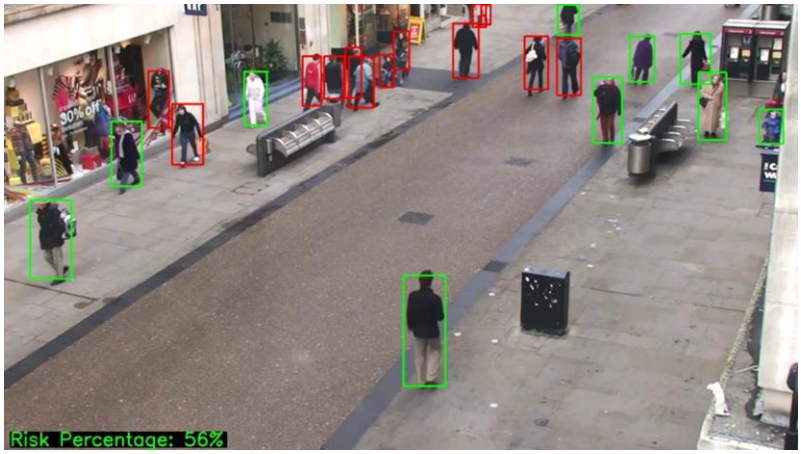

Fig. 8. Social distancing detection for the test video

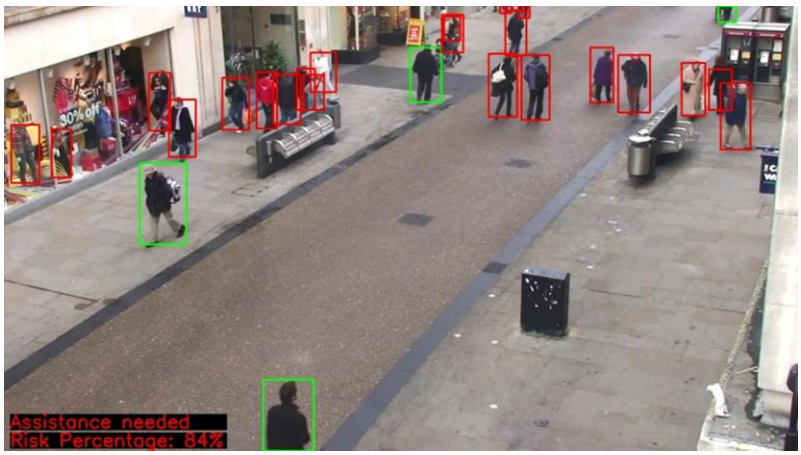

Fig. 9. Social distancing detection for the test video

Table 3. Performance metrics for person detection

Model	YOLOv4	Faster RCNN	SSD
Precision	0.77	0.72	0.49
Recall	0.79	0.75	0.89
F1-score	0.78	0.73	0.31
mAP	82.47%	66.10%	41.34%
FPS	14~17	7~8	49~54

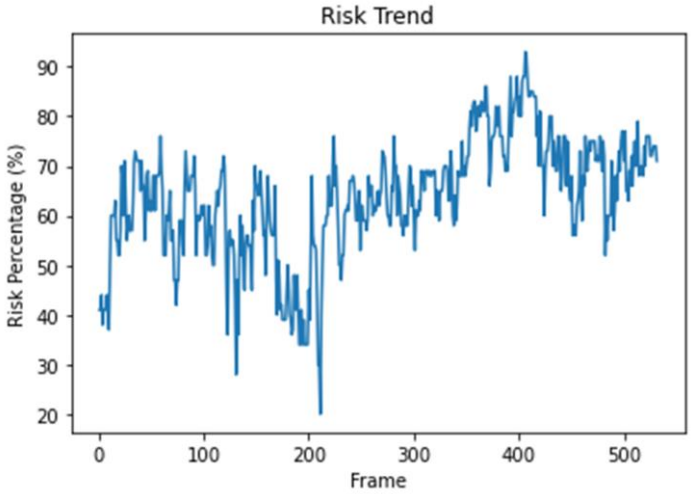

Fig. 10. Risk Percentage vs Frame for the test video

was taken from the video for 21 seconds and 531 frames. The distance is calculated using Equation 1, and the color of the bounding box is determined whether it has satisfied the conditions of social distancing. The red box is set for a risky person with less than 1 meter, and the green box indicates that the distance is more than one meter between each detection.

The risk percentage is shown at the bottom left of the frame using Equation 2. Based on Figure 8, 10 green boxes and 13 red boxes were detected, resulting in a risk percentage of 56%. For Figure 9, there were 4 green and 22 red boxes were detected, giving 84% of the risk percentage. The percentage was captured during the testing and displayed in the graph shown in Figure 10 to show the trend of the level of compliance by the citizen. This data can be taken into consideration by the authorities for them to improve future inspection efficiency.

IV. Conclusions

In conclusion, the research has investigated the reliability of the detection algorithms for social distancing inspection. Three deep learning models are studied to determine the best social distancing algorithms. Experimental results showed that YOLOv4 achieved the highest performance of mAP compared to other detection models with a balance speed. Despite the highest performance, the calculation of social distancing detection did not use a proper camera calibration, and the distance is based on the assumption and may lead to inaccuracy for the social distance. Therefore, future work can be extended to include a proper camera calibration and alert system to improve the social distance monitoring system.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no known conflict of financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission information are available at http://journal2.um.ac.id/index.php/keds.

Publisher's Note: Department of Electrical Engineering - Universitas Negeri Malang remains neutral with regard to jurisdictional claims and institutional affiliations.

References

[1] World Health Organization: Coronavirus disease (COVID-19): How is it transmitted? https://www.who.int/news-room/q-a-detail/coronavirus-disease-covid-19-how-is-it-transmitted (2020). Accessed 5 Apr 2021.

[2] I. Leong, PKP: Kadar pematuhan 92 peratus tapi penjarakan sosial gagal dipatuhi - Ismail Sabri. Astro Awani. https://www.astroawani.com/berita-malaysia/pkp-kadar-pematuhan-92-peratus-tapi-penjarakan-sosial-gagal-dipatuhi-ismail-sabri-234881 (2020). Accessed 5 Nov 2020.

[3] A. Povera: Significant increase in the number of SOP flouters. New Straits Times. https://www.nst.com.my/news/nation/2020/07/611744/significant-increase-number-sop-flouters (2020). Accessed 5 Nov 2020.

[4] I. Hilmy: Eight slapped with RM1k compound each for breaching recovery MCO in Penang. The Star. https://www.thestar.com.my/news/nation/2020/07/25/eight-slapped-with-rm1k-compound-each-for-breaching-recovery-mco-in-penang (2020). Accessed 5 Nov 2020.

[5] W. Lan, J. Dang, Y. Wang and S. Wang, "Pedestrian Detection Based on YOLO Network Model," *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2018, pp. 1547-1551.

[6] N. S. Punn, S. K. Sonbhadra, S. Agarwal, and G. Rai, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques," May 2020.

[7] D. Yang, E. Yurtsever, V. Renganathan, K. A. Redmill, and Ü. Özgüner, "A Vision-based Social Distancing and Critical Density Detection System for COVID-19," Jul. 2020.

[8] 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: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) *Computer Vision – ECCV 2014, Lecture Notes in Computer Science*, vol. 8693, pp. 740–755. Springer, Cham (2014).

[9] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580-587.

[10] R. Girshick, "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440-1448.

[11] 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, 1 June 2017.

[12] W. Liu *et al*., "SSD: Single Shot MultiBox Detector," 2016, pp. 21–37.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788.

[14] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517-6525.

[15] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 2018.

[16] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020.

[17] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818-2826.

[18] A. G. Howard *et al*., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017.

[19] C. -Y. Wang, H. -Y. Mark Liao, Y. -H. Wu, P. -Y. Chen, J. -W. Hsieh and I. -H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 1571-1580.

[20] K. E. Koech: Confusion matrix for object detection. Towards Data Science. https://towardsdatascience.com/confusion-matrix-and-object-detection-f0cbcb634157 (2020). Accessed 29 Apr 2021.

[21] N.-D. Nguyen, T. Do, T. D. Ngo, and D.-D. Le, "An Evaluation of Deep Learning Methods for Small Object Detection," *J. Electr. Comput. Eng.*, vol. 2020, pp. 1–18, Apr. 2020.

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