sustainability

Article

Indoor Distance Measurement System COPS (COVID-19 Prevention System)

Minchan Shin and Nammee Moon *

Department of Computer Science and Engineering, Hoseo University, Asan 31499, Korea; shinmc0322@gmail.com
* Correspondence: nammee.moon@gmail.com

Abstract: With the rapid spread of coronavirus disease 2019 (COVID-19), measures are needed to detect social distancing and prevent further infection. In this paper, we propose a system that detects social distancing in indoor environments and identifies the movement path and contact objects according to the presence or absence of an infected person. This system detects objects through frames of video data collected from a closed-circuit television using You Only Look Once (v. 4) and assigns and tracks object IDs using DeepSORT, a multiple object tracking algorithm. Next, the coordinates of the detected object are transformed by image warping the area designated by the top angle composition in the original frame. The converted coordinates are matched with the actual map to measure the distance between objects and detect the social distance. If an infected person is present, the object that violates the movement path and social distancing of the infected person is detected using the ID assigned to each object. The proposed system can be used to prevent the rapid spread of infection by detecting social distancing and detecting and tracking objects according to the presence of infected persons.

Keywords: COVID-19; social distancing; indoor environment; YOLOv4; multiple object tracking

check for updates

Citation: Shin, M.; Moon, N. Indoor Distance Measurement System COPS (COVID-19 Prevention System). *Sustainability* **2021**, *13*, 4738. https://doi.org/10.3390/su13094738

Academic Editor: Amir Mosavi

Received: 9 February 2021
Accepted: 14 April 2021
Published: 23 April 2021

1. Introduction

After discovering the coronavirus disease 2019 (COVID-19), a severe acute respiratory syndrome, in Wuhan, Hubei Province, China, in December 2019, it has been reported worldwide, including all over China [1]. On 12 February 2020, the World Health Organization (WHO) designated the disease as COVID-19 and delivered a message about the risk of spreading it [2]. To date, the situation has caused the spread of the virus, and measures are sought to prevent the spread of the virus worldwide.

The WHO has proposed and recommended various infection control measures to prevent the spread of COVID-19. One of the currently actively recommended infection control measures is social distancing. Social distancing is a method of minimizing the mortality rate and slowing the disease spread by setting 2 m as the standard distance to reduce the possibility of contact between infected and uninfected people [3–6]. However, in indoor environments, such as department stores, cafes, and other stores, which are confined spaces, it is difficult to maintain social distancing because the population density is relatively higher than for outdoor environments. Environmental health scientists report that 3.8 million people worldwide die prematurely from diseases caused by indoor air pollution [7]. In addition, based on the results of studies showing that the probability of a secondary infection in indoor spaces due to COVID-19 is high, a high risk exists of becoming infected with COVID-19 in an indoor environment [8–10].

The spread of COVID-19 correlates with the time of contact with an infected person. According to a study published in January 2020, an increase in the number of infections among healthcare professionals occurred after the COVID-19 outbreak. In a study of 138 hospitalized patients infected with COVID-19 in China, 57 were infected in hospitals in indoor environments. It was announced that 40 of these were medical professionals working in hospitals [11,12]. Based on the research results, it is possible to determine the correlation with the contact time with the infected person.

Based on the above, a high risk exists for the spread of COVID-19 in indoor environments. Thus, it is important to maintain social distancing to prevent the spread of COVID-19 in indoor environments. However, it is important to determine whether social distancing is well maintained as a result of the difficulty of social distancing due to the characteristics of the indoor environment. Because the standard for social distancing is set at 2 m, if social distancing is outside the standard radius, it should be judged safe, and if it is included within the standard radius, it should be judged dangerous. In addition, to prevent the spread of infection when an infected person is present in an indoor space, it is necessary to quickly identify the movement path of the infected person and the people who have contacted this person. This is because additional confirmed cases may occur due to the movement path and contact with the infected person. However, because the distance between people is not always clearly judged, it is difficult to determine the path of an infected person [13]. Currently, no system exists to understand the indoor movement of people infected with COVID-19. Therefore, a method to prevent further diffusion is needed.

Therefore, this paper presents the design and construction of a system to detect social distancing in an indoor environment and identify the movement path and contact objects of a person infected with COVID-19. In this context, the method for determining the infected person is to assume that a random object is an infected object and proceed with the experiment.

The system in this paper detects objects by applying You Only Look Once (YOLOv4) based on the image data collected through closed-circuit television (CCTV) and uses DeepSORT, a multiple object tracking (MOT) algorithm, to assign IDs and track objects. Next, we derive the weight of the transformation matrix to obtain the object coordinates through image warping of the original frame. The center point of the bounding box for the object is transformed to fit the shape of the transformed frame using the derived transform matrix weight. The converted center point is matched with an actual map to convert the distance between objects in an indoor space into an actual distance through the CCTV. Then, the actual distance is measured using the Euclidean distance measurement formula. The object risk is determined by checking whether it is within 2 m of the social distancing standard set through the measured distance to the object. If an infected person is present, tracking the infected person through the details related to the object ID can be performed to determine the movement path of the infected person and the contact object. The proposed system detects social distancing in an indoor space and, if an infected person is present, tracks the path of the infected person and identifies the contacted objects to prevent infection.

2. Related Work

2.1. Object Detection

Object detection is a type of computer technology closely related to computer vision and image processing. Video data for object detection are collected through various devices, such as webcams, CCTV, and Microsoft Kinect. Many studies have been conducted to detect an object using video data collected through such devices. To detect objects, studies using feature extraction algorithms, such as the scale-invariant feature transform and speeded up robust feature, have been conducted [14,15]. However, several deep learning models have been studied based on the convolutional neural network (CNN), which has the advantage of high reliability of feature extraction and matching. Representatively, CNN-based object detection deep learning models include the single-shot multibox detector [16], faster region-based convolutional network [17], and YOLO [18].

Among the representative CNN-based object detection deep learning models, YOLOv4 [19], which was extended from the existing YOLO model suitable for real-time object detection, has been widely used. Prior studies applying this model include real-time multiclass object detection and location estimation, real-time behavior recognition and prediction, and real-time single and multiple object detection using drones [19–22]. Based on the results of each study, we confirmed the excellent performance of YOLOv4 in deriving fast detection speed and accuracy.

2.2. Multiple Object Tracking

The purpose of object tracking is to segment the region of interest from the video input image and continuously track motion, positioning, and occlusion situations. Object detection and tracking is a technology that can be applied to video detection systems, robot vision, traffic monitoring, video inpainting, and animation. The framework frequently used in MOT is simple online and real-time tracking (SORT) [23], which focuses on inter-frame prediction and association. This framework achieves high performance in terms of speed and accuracy and can handle long-term occlusion. It also focuses on efficiency to facilitate real-time tracking and facilitate more use in such applications as autonomous vehicle and pedestrian tracking.

The DeepSORT [24] algorithm that integrates shape information is frequently used to improve SORT performance. DeepSORT is a tracking method that can effectively reduce the number of ID switches by tracking objects through occlusion for a longer period and by learning through deep association metrics for a large personal reidentification dataset.

Many studies have been conducted with the MOT method. Hou improved low-reliability tracking filtering using the DeepSORT algorithm to reduce the influence of unreliable detection on vehicle tracking [25]. Kapania demonstrated high performance by combining YOLOv3 and RetinaNet based on the DeepSORT algorithm [26]. Wang achieved stable driver-face detection and tracking based on the DeepSORT algorithm [27]. Based on the results of each study, MOT exhibited high performance in tracking.

2.3. Indoor Object Distance Measurement

Object positioning should be performed to measure the distance between objects indoors. A technique frequently used for object positioning indoors is fingerprinting using received signal strength indication (RSSI) values with a Bluetooth low-energy beacon and a Wi-Fi device [28,29]. In this technique, the indoor space is divided into cells, and a radio map is constructed by collecting the media access control addresses and RSSI values of the access point for each cell. Through this method, the cell information with a value most similar to the access point value at the user's current location is returned to determine the location [29]. Recently, as a method for positioning a precise location, research has been conducted by incorporating such technologies as the support vector machine, random forest, and cloudlet computing using the RSSI value of the transmitting device [29–32].

However, the problem with this method is that it is designed for measuring the distance by positioning an object, and the error rate of the distance measurement is large. Therefore, to solve the error rate problem in distance measurement, a method for precise distance measurement is used via an algorithm for distance measurement between an object in the video frame of the camera and an actual object. Rahim constructed a precise system capable of determining risk factors by measuring the social distances of objects through a camera [33]. Yan built a precision system using cameras to detect pedestrians and warn them using distance detection [34]. Based on these studies, it is used as a method for precise distance measurement using a camera for measuring the distance of an object.

3. System Overview

Figure 1 is the overall system structure comprising seven parts. The first part is learning the image data and labeled data of the MS COCO dataset [35] through YOLOv4 to derive the learning weight for object detection. The second part is specifying the person class and bounding box by detecting an object in real time based on the YOLOv4 model using the extracted learning weight and the video frame extracted through the CCTV for object detection. The third part is assigning IDs to objects in real time by assigning the IDs and bounding boxes of objects using the DeepSORT algorithm suitable for MOT for object tracking and enabling tracking. The fourth part is designating four coordinates of the space in which a person can move in the video frame and transforming the frame corresponding to the assigned coordinates. Fifth, the lower center point of the bounding box of the detected objects is transformed using the weight of the transform matrix derived when transforming the frame, which is measuring the distance between objects through the Euclidean distance formula. The sixth part is identifying the overall objects using the results of the third and fifth processes and storing the information for each object ID in the database so that the identified objects can detect and track the social distance standard (2 m). The seventh process is detecting the movement path of the infected object and objects in contact with it (if an infected object is present in the indoor environment) and storing the path in the database.

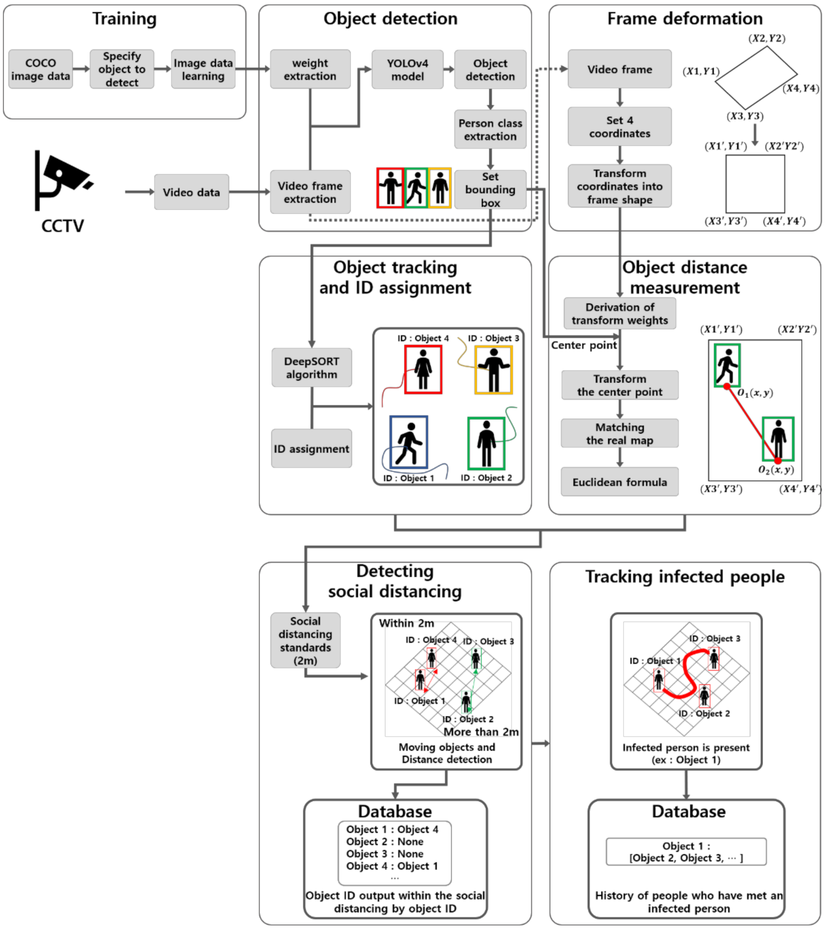


Figure 1. Overall system structure.

The distance between objects in an indoor space is identified through the above process, and information on the objects included in the social distancing criteria is extracted to detect social distancing. If an infected object is present, a system is built that judges even the objects in contact with the path of the infected object.

This study aims to detect social distancing by conducting real-time object detection, object tracking, and infected object tracking and determining the objects in contact with the infected person's movement path when present. As presented in Table 1, the differentiation of the proposed system is illustrated by comparing it with other systems.

Table 1. System comparison table with the proposed system.

Key Element Paper Author	Real-Time	Object Detection	Object Tracking	Infected Object Tracking
Yang [34]	Yes	Yes	No	No
Punn [36]	Yes	Yes	No	No
Gupta [37]	Yes	Yes	No	No
Cristani [38]	Yes	Yes	No	No
Bhambani [39]	Yes	Yes	Yes	No
Proposed System	Yes	Yes	Yes	Yes

4. System Implementation Method

4.1. Image Data Training

Learning is conducted using the image data and dataset labeled with the object in the image data to detect an object. Object detection should be performed by deriving the learning weights for the images. Therefore, in this paper, MS COCO dataset is trained by YOLOv4 model as shown in Figure 2. Moreover, YOLOv4 is a model based on YOLOv3 and the applied backbone (CSPDarknet53), neck (spatial pyramid pooling (SPP) and path aggregation network (PAN)), bag of freebies (BoF), and bag of specials (BoS) [19]. The training was conducted by dividing the training and testing ratio in the MS COCO 2017 dataset into 8:2 using this YOLOv4 model. The weight derived after training is used for real-time object detection.

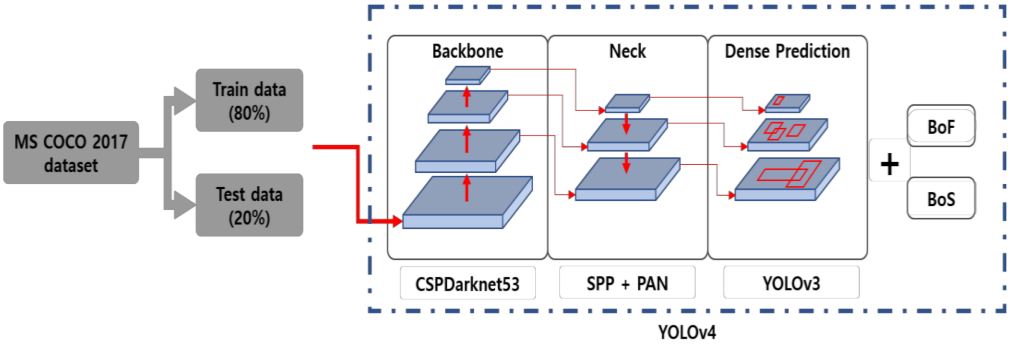


Figure 2. Data training process using the MS COCO dataset and YOLOv4 model.

Table 2 is a table comparing the YOLOv3 model used by Punn [35] and the YOLOv4 model used in this paper. To check the performance of the model, the weights derived from the training results of each model are used. The learning weight of each model was compared by measuring mean average precision (mAP) based on 5000 test images and measuring frames per second (FPS) based on video. As a result of comparing the YOLOv3 model and the YOLOv4 model, it can be seen that the YOLOv4 model has 38.4% higher mAP and about twice the speed. Based on the results, YOLOv4 is used as a method for fast and accurate object detection.

Table 2. System comparison table with the proposed system.

Evaluation Factor Model	mAP	FPS
YOLOv3	30.4	30.3
YOLOv4	42.1	60.2

4.2. Object Detection

Figure 3 is an example of a sequence that detects YOLOv4 in the field. The sequence of the figure above designates the video frame extracted from the CCTV as the input data of the YOLOv4 model using the training weights derived through training of the image data of the MS COCO dataset in Section 4.1. Afterward, objects are detected, and only the person class is extracted among them, and the bounding box is set. Through this process, the person object of the video frame is detected in real time.

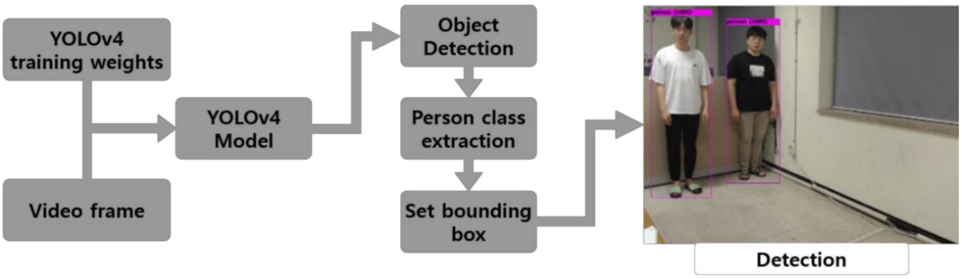


Figure 3. Example of object detection using YOLOv4.

4.3. Multiple Object Tracking and ID Assignment

The DeepSORT algorithm is used as a method for tracking multiple objects and allocating IDs for each object. Figure 4 illustrates the DeepSORT process. The DeepSORT algorithm performs each process by dividing it into two parts: the previous and current object tracking frames. In the previous frame, the ID is assigned to the object found through the object detector, and information on the position and velocity of the object is included. Even in the current frame, the part that allocates the ID to the object found through the object detector is the same, but it also includes deep cosine metric learning to extract the feature information of the object. In this way, object tracking is performed through a series of processes using the previous and current frames through DeepSORT.

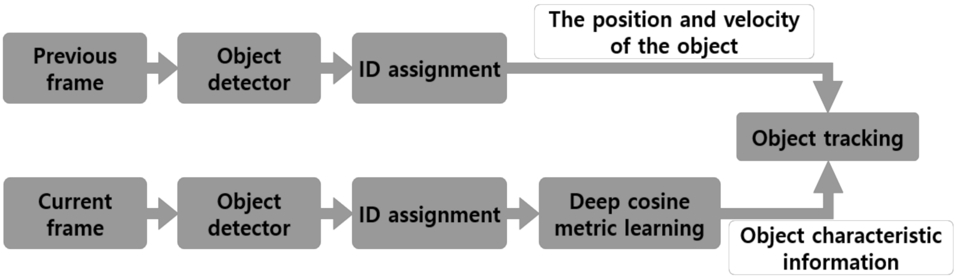

Previous frame
Object detector
ID assignment
The position and velocity of the object
Object tracking
Current frame
Object detector
ID assignment
Deep cosine metric learning
Object characteristic information

Figure 4. Tracking multiple objects and assignment IDs using DeepSORT.

4.4. Video Frame Transformation

The figure at the top left of Figure 5 is an example of one frame among video frames captured through the CCTV. Four coordinates are designated on the ground, and four coordinates of the frame to be converted are set to compose a frame from the original frame to the top angle composition. The weight of the transformation matrix is derived from transforming the specified coordinates of the original frame into the coordinates of the frame to be transformed. Next, image warping, a kind of geometric transformation, is performed using the derived transformation matrix weight. The figure at the bottom left is an example of the conversion of the four designated coordinate areas of the original frame to the top angle structure from above. Preliminary work is performed through the above process to map the actual distance to the designated area of the video frame.

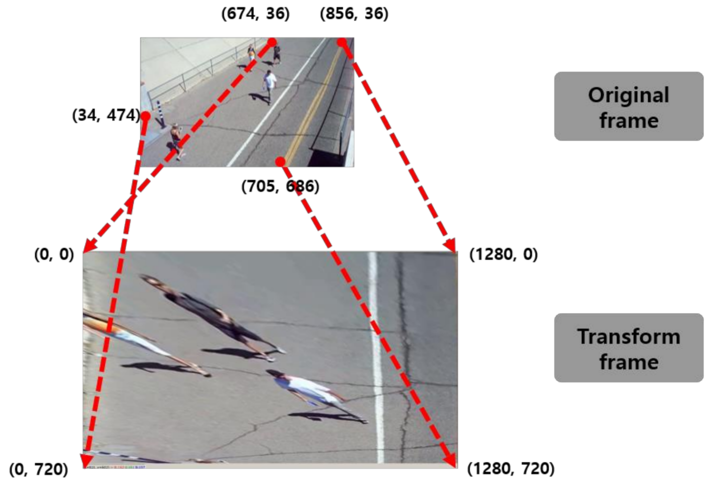


Figure 5. Example of transforming the coordinates specified in the original frame.

4.5. Object Distance Measurement

To measure the distance between objects, the coordinates of the individual objects are necessary. To obtain the coordinates of each object, the bounding box of the object detected in the video frame is used. Figure 6 is the bounding box area of the detected object. The O_{center_point}, which is the center point of the lower bounding box of the object, can be determined through Equation (1):

$$O_{center\_point} = \left(\left(\frac{x' - x}{2} \right) + x,\ y' \right) \tag{1}$$

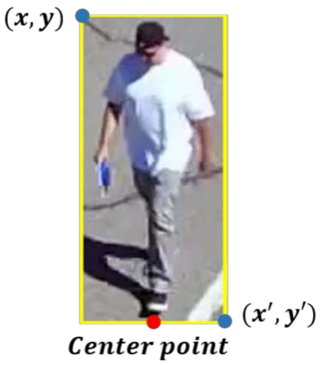


Figure 6. Example of the bounding box area of the detected object.

After obtaining the object coordinates, matching the actual coordinates is necessary. Before matching, the original frame was transformed to set the same depth as the position of the objects through coordinates, as in Section 4.4. Therefore, the object's center coordinate must be multiplied by the weight of the transformation matrix derived through the frame transformation. The transformed object center coordinate $O_{center\_transform}$ is obtained by multiplying the boundary coordinates of the object by the weight of the transformation matrix through Equation (2):

$$O_{center\_transform} = O_{center\_point} * w_{frame\_weight} \tag{2}$$

To map the center coordinate $O_{center\_transform}$ of the converted object with the coordinates of the actual distance, the converted frame and actual distance are mapped. Through

Equations (3) and (4), the x coordinate of the actual object, $O_{real\_center\_x}$, and the y coordinate, $O_{real\_center\_y}$, are obtained:

$$O_{real\_center\_x} = \left(\frac{W_{real\_width} * O_{center\_transform\_x}}{W_{frame\_width}} \right) \tag{3}$$

$$O_{real\_center\_y} = \left(\frac{H_{real\_height} * O_{center\_transform\_y}}{H_{frame\_height}} \right) \tag{4}$$

When the actual x and y coordinates of the object are derived, the distance between each object can be calculated. To measure the distance between objects, we use the Euclidean distance formula. By applying the two-dimensional Euclidean distance formula for measuring the distance between objects through Equation (5), $d_{n,m}$, which is the distance between objects n and m, is obtained:

$$d_{n,\ m} = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2} \tag{5}$$

4.6. Detecting Social Distancing and Infected Person Tracking

Figure 7 is a flow chart for detecting social distancing and tracking the presence of an infected person. First, the social distancing detection stores detailed information on the detected object in a database. Next, whether the distance between the detected objects (2 m) has been violated is determined. If not, it does not indicate risk. If so, a list is designated by ID for objects within 2 m. Then, information on object IDs within 2 m, the detected location, and the time are stored in the database. In addition, risks can be identified by marking a risk bounding box on the object. If an infected person is present, the ID of the infected object can be extracted from the database, and the tracking of the infected object and contact objects can be identified based on the contents of the movement path, the contact object IDs, and the contact time. Table 3 is an example table of details stored for information related to the object ID, detected time, detection end time, and movement path based on detailed information on the detected object.

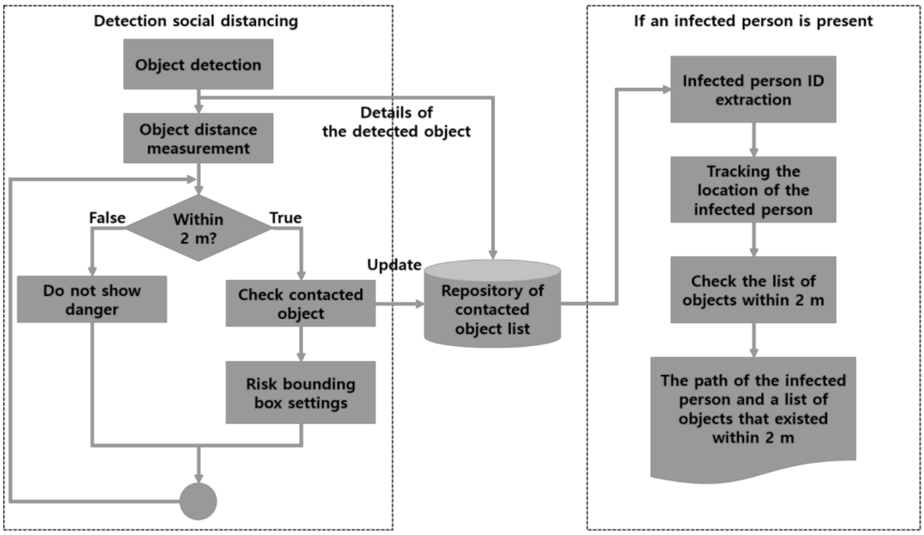

Detection social distancing
Object detection
Object distance measurement
Details of the detected object
False
Within 2 m?
True
Do not show danger
Check contacted object
Risk bounding box settings
Update
Repository of contacted object list
If an infected person is present
Infected person ID extraction
Tracking the location of the infected person
Check the list of objects within 2 m
The path of the infected person and a list of objects that existed within 2 m

Figure 7. Detecting the social distancing and infected person tracking flow.

Table 4 is an example table listing the object ID of an object that violated the social distancing (2 m) of the detected objects, the object ID within 2 m, the detected location, and the details stored for the contact time.

Table 3. Details of the detected object.

Object ID	Start Time	End Time	Movement Path
Object$_1$	2021-04-10 13:13:02	2021-04-10 13:14:30	(210, 220), (205, 216) $\cdots$
Object$_2$	2021-04-10 13:13:10	2021-04-10 13:13:52	(130, 150), (135, 155) $\cdots$
Object$_3$	2021-04-11 08:15:10	2021-04-10 08:16:10	(480, 540), (490, 530) $\cdots$
$\cdots$	$\cdots$	$\cdots$	$\cdots$
Object $_n$	2021-04-11 08:15:02	2021-04-11 08:16:35	(504, 402), (545, 395) $\cdots$

Table 4. Social distancing detection list.

Object ID	Object ID within 2 m	Detected Location	Time (Sec)
Object$_1$	Object$_4$	(145, 165), (139, 155) $\cdots$	15
Object$_1$	Object$_7$	(195, 185), (180, 175) $\cdots$	20
Object$_3$	Object $_n$	(480, 540), (490, 530) $\cdots$	10
$\cdots$	$\cdots$	$\cdots$	$\cdots$
Object $_n$	Object$_3$	(504, 402), (545, 395) $\cdots$	10

5. Experiment

5.1. Environment

Table 5 lists the detailed experimental environment of the system proposed in this paper.

Table 5. Environment.

Type	Contents
Server OS	Ubuntu 18.04.5 LTS
Server CPU	Intel i9-9900k
Server GPU	Nvidia Titan RTX
CCTV Device	Logitech C922 Pro HD Stream Webcam
Python version	3.6.9
Tensorflow version	1.14.0
Keras version	2.4.1
OpenCV version	4.2.0

The figure in the left of Figure 8 is an indoor space set up as an actual experiment environment. The figure on the right is set to 8 m × 12 m grid as a space for social distance detection and infected person tracking.

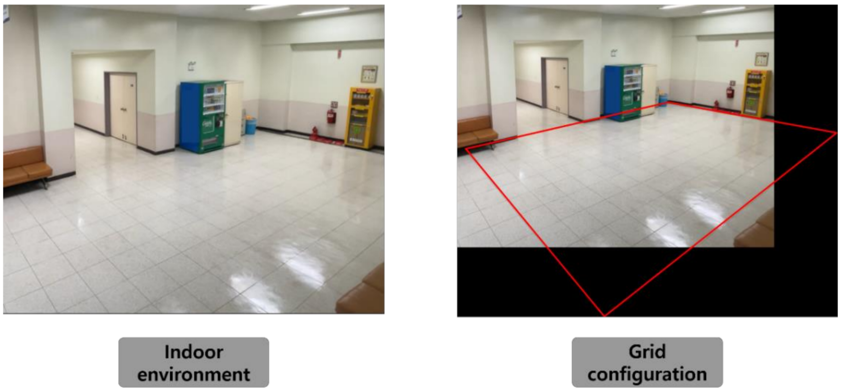


Figure 8. Indoor test environment.

5.2. Experiment

Figure 9 shows the original frame to transformed frame conversion. Through im-age warping, the specified area of the real space is transformed as shown in the right side of the figure, and the weights of the transform matrix are derived. Through this derived weights, object coordinates are converted into coordinates of the transformed frame.

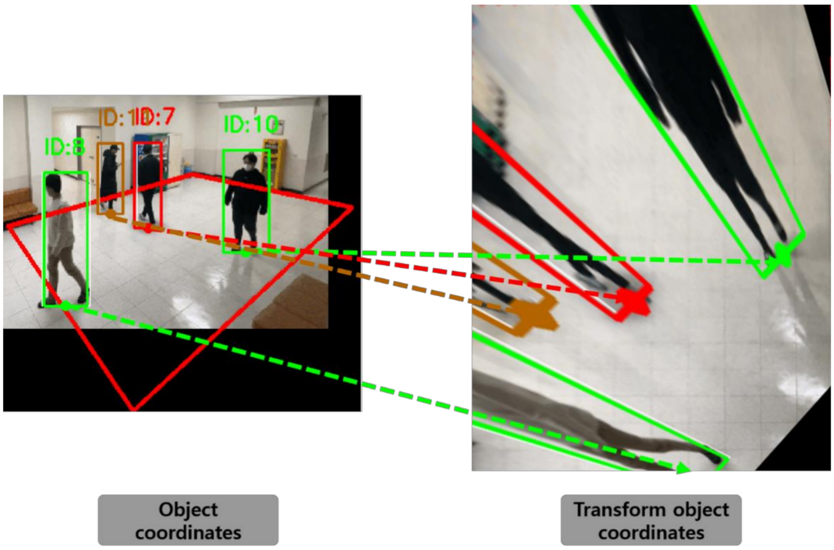


Figure 9. Convert object coordinates to transformed frame coordinates.

The YOLOv4 model is used to detect an object in real space, and the ID assignment and object tracking are performed on the detected object through the DeepSORT algorithm. Then, the coordinates of each object are changed using the weight of the transformation matrix. The distance is detected through the process in Section 4.5 via the coordinates of the transformed object.

In this experiment, when tracking an object through DeepSORT, it was confirmed that a frequent ID switching problem occurred due to an object occlusion problem. As a solution to the frequent ID switching problem, four FPSs are compared. For comparison, there are 15 objects in the video of this experiment, and the performance is compared by comparing the number of object detections in each FPS.

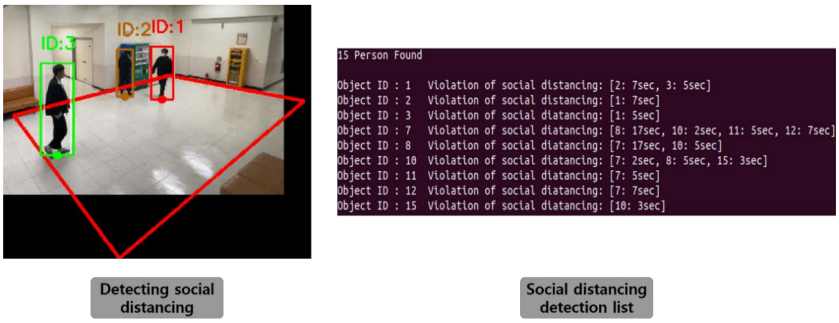
Through this, 24FPS derives the highest accuracy, as shown in Table 6 below. Therefore, in this paper, we conducted an experiment using video of 24FPS based on the results in Table 6.

Table 6. FPS comparison to solve ID switching problem.

FPS Element	10	24	35	50
Actual number of objects	15	15	15	15
Number of objects detected	24	16	19	18
Accuracy	62.5	**93.7**	78.9	83.3

The figure in the left of Figure 10 is a part of the test conducted by moving objects in various directions through the designated area in the actual space, which was taken by a webcam for 1 min. In the left part of the picture, objects can be observed moving in real space. Objects are detected in real time through the YOLOv4 model and numerical ID assignment and tracking using DeepSORT algorithm. The listing on the right is a screenshot of the details of objects violating social distancing (2 m) by object ID through an actual experiment like that in the picture on the left.

In the experiment of this paper, the object with ID 7 is set as the infected object and the experiment is carried out. Figure 11 shows the experiment after setting the object with ID 7 as an infected object. Infected objects are indicated by a red bounding box. Objects that violate social distancing from infected objects are marked with a brown bounding box to indicate danger. Other objects are classified in green, so that dangerous situations can be easily identified. To determine the movement path of the infected object (ID 7) in practice, the movement coordinates are derived based on the ID-related information obtained through social distancing detection as in Section 4.6.



```
15 Person Found

Object ID : 1   Violation of social distancing: [2: 7sec, 3: 5sec]
Object ID : 2   Violation of social distancing: [1: 7sec]
Object ID : 3   Violation of social diatancing: [1: 5sec]
Object ID : 7   Violation of social diatancing: [8: 17sec, 10: 2sec, 11: 5sec, 12: 7sec]
Object ID : 8   Violation of social diatancing: [7: 17sec, 10: 5sec]
Object ID : 10  Violation of social diatancing: [7: 2sec, 8: 5sec, 15: 3sec]
Object ID : 11  Violation of social diatancing: [7: 5sec]
Object ID : 12  Violation of social diatancing: [7: 7sec]
Object ID : 15  Violation of social diatancing: [10: 3sec]
```

Figure 10. Detecting social distancing in real space (left) and details of contacted objects by ID (right).

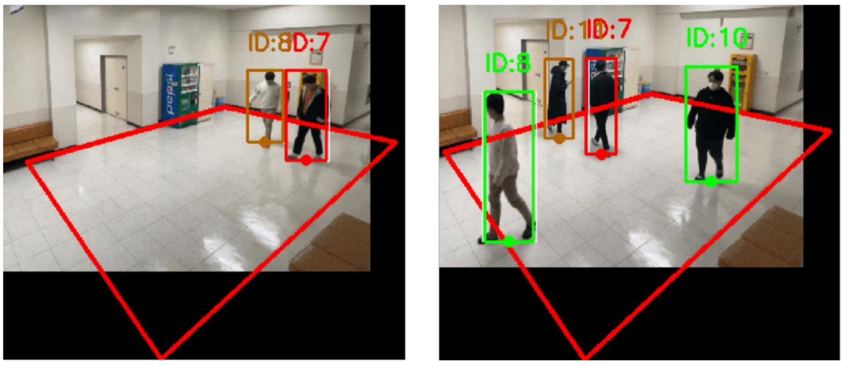


Figure 11. Infected object setup and experiment.

The left graph of Figure 12 is a graph showing the movement path of an infected object (ID 7) in an actual environment. In the graph, the movement path of the infected object (ID 7) shows the path at 8 m × 12 m of the actual experimental environment. The graph on the right shows infected objects (ID 7) and those who violate social distancing. In addition, it is possible to derive an index that can determine the risk of each object by indicating the time in violation of social distancing.

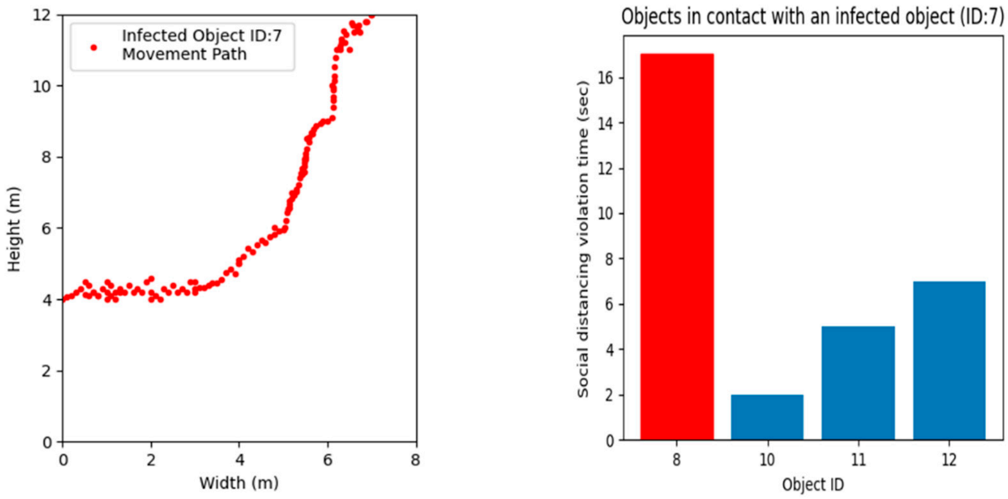


Figure 12. Movement path of infected object (left); Objects in contact with infected objects (right).

6. Conclusions

The COVID-19 pandemic is currently underway. Consequently, social distancing remains an important way to prevent the spread of the virus in the current situation. However, it is difficult to practice social distancing in an indoor environment because the space is narrow and crowded compared to the outdoor environment. In addition, as the range of people's movement paths in indoor space narrows, the number of people who come into contact with an infected person increases considerably if an infected person is present. Thus, a method is necessary to detect social distancing and, if an infected person is present, identify those who have come into contact with the path of the infected person to prevent the spread. Therefore, we implemented a system to prevent the rapid spread of infection by detecting social distancing and detecting and tracking objects according to the presence of infected persons. Because the current pandemic is changing the pattern of life worldwide, we plan to expand the system to solve the current situation.

Author Contributions: Conceptualization, M.S.; methodology, M.S.; software, M.S.; validation, M.S.; formal analysis, M.S.; investigation, M.S.; resources, M.S.; data curation, M.S.; writing—original draft preparation, M.S.; writing—review and editing, M.S. and N.M.; visualization, M.S.; supervision, N.M.; project administration, N.M.; funding acquisition, N.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A2C2011966).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from Minchan shin. But restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available. However, the data can be requested again by the author and used with the permission of Minchan Shin.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Velavan, T.P.; Meyer, C.G. The COVID-19 epidemic. *Trop. Med. Int. Health* **2020**, *25*, 278. [CrossRef]
2. Zu, Z.Y.; Di Jiang, M.; Xu, P.P.; Chen, W.; Ni, Q.Q.; Lu, G.M.; Zhang, L.J. Coronavirus Disease 2019 (COVID-19): A Perspective from China. *Radiology* **2020**, *296*, E15–E25. [CrossRef] [PubMed]
3. Lewnard, A.J.; Lo, N.C. Scientific and ethical basis for social-distancing interventions against COVID-19. *Lancet Infect. Dis.* **2020**, *20*, 631–633. [CrossRef]
4. Thu, T.P.B.; Ngoc, P.N.H.; Hai, N.M.; Tuan, L.A. Effect of the social distancing measures on the spread of COVID-19 in 10 highly infected countries. *Sci. Total. Environ.* **2020**, *742*, 140430. [CrossRef] [PubMed]
5. McGrail, D.J.; Dai, J.; McAndrews, K.M.; Kalluri, R. Enacting national social distancing policies corresponds with dramatic reduction in COVID19 infection rates. *PLoS ONE* **2020**, *15*, e0236619. [CrossRef] [PubMed]
6. Mittal, R.; Meneveau, C.; Wu, W. A mathematical framework for estimating risk of airborne transmission of COVID-19 with application to face mask use and social distancing. *Phys. Fluids* **2020**, *32*, 101903. [CrossRef]
7. Du, W.; Wang, G. Indoor Air Pollution was Nonnegligible during COVID-19 Lockdown. *Aerosol Air Qual. Res.* **2020**, *20*, 1851–1855. [CrossRef]
8. Sun, C.; Zhai, Z. The efficacy of social distance and ventilation effectiveness in preventing COVID-19 transmission. *Sustain. Cities Soc.* **2020**, *62*, 102390. [CrossRef] [PubMed]
9. Nwanaji-Enwerem, J.C.; Allen, J.G.; Beamer, P.I. Another invisible enemy indoors: COVID-19, human health, the home, and United States indoor air policy. *J. Expo. Sci. Environ. Epidemiol.* **2020**, *30*, 773–775. [CrossRef] [PubMed]
10. Nishiura, H.; Oshitani, H.; Kobayashi, T.; Saito, T.; Sunagawa, T.; Matsui, T.; Wakita, T.; MHLW COVID-19 Response Team; Suzuki, M. Closed environments facilitate secondary transmission of coronavirus disease 2019 (COVID-19). *MedRxiv* **2020**. [CrossRef]
11. Wang, D.; Hu, B.; Hu, C.; Zhu, F.; Liu, X.; Zhang, J.; Wang, B.; Xiang, H.; Cheng, Z.; Xiong, Y.; et al. Clinical Characteristics of 138 Hospitalized Patients With 2019 Novel Coronavirus–Infected Pneumonia in Wuhan, China. *JAMA* **2020**, *323*, 1061–1069. [CrossRef] [PubMed]
12. Guan, W.J.; Ni, Z.Y.; Hu, Y.; Liang, W.H.; Ou, C.Q.; He, J.X.; Liu, L.; Shan, H.; Lei, C.L.; Hui, D.S.C.; et al. Clinical Characteristics of Coronavirus Disease 2019 in China. *N. Engl. J. Med.* **2020**, *382*, 1708–1720. [CrossRef] [PubMed]

13. Benreguia, B.; Moumen, H.; Merzoug, M.A. Tracking COVID-19 by Tracking Infectious Trajectories. *IEEE Access* **2020**, *8*, 145242–145255. [CrossRef]
14. Truong, M.T.N.; Kim, S. A tracking-by-detection system for pedestrian tracking using deep learning technique and color in-formation. *J. Inf. Process. Syst.* **2019**, *15*, 1017–1028.
15. Gupta, S.; Thakur, K.; Kumar, M. 2D-human face recognition using SIFT and SURF descriptors of face's feature regions. *Vis. Comput.* **2021**, *37*, 447–456. [CrossRef]
16. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision 2016, Amsterdam, The Netherlands, 8–16 October 2016.
17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef]
18. Uijlings, J.R.R.; Van De Sande, K.E.A.; Gevers, T.; Smeulders, A.W.M. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
19. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
20. Jiang, M.; Rao, Y.; Zhang, J.; Shen, Y. Automatic behavior recognition of group-housed goats using deep learning. *Comput. Electron. Agric.* **2020**, *177*, 105706. [CrossRef]
21. Juang, J.-G.; Tu, G.-T.; Liao, Y.-H.; Huang, T.-H.; Chang, S.-I. Drone patrol using thermal imaging for object detection. Infrared Sensors, Devices, and Applications X.; SPIE-Intl Soc Optical Eng. *Sensors* **2020**, *11503*, 115030U.
22. Wu, D.; Lv, S.; Jiang, M.; Song, H. Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and ac-curate detection of apple flowers in natural environments. *Comput. Electron. Agric.* **2020**, *178*, 105742. [CrossRef]
23. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468.
24. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017.
25. Hou, X.; Wang, Y.; Chau, L.-P. Vehicle Tracking Using Deep SORT with Low Confidence Track Filtering. In Proceedings of the 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Taipei, Taiwan, 18–21 September 2019; pp. 1–6.
26. Kapania, S.; Saini, D.; Goyal, S.; Thakur, N.; Jain, R.; Nagrath, P. Multi Object Tracking with UAVs using Deep SORT and YOLOv3 RetinaNet Detection Framework. In Proceedings of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems, Association for Computing Machinery (ACM), Bangalore, India, 1–22 November 2020; pp. 1–6.
27. Wang, Q.; Cao, L.; Xia, J.; Zhang, Y. *MTCNN-KCF-deepSORT: Driver Face Detection and Tracking Algorithm Based on Cascaded Kernel Correlation Filtering and Deep SORT*; SAE International: Troy, MI, USA, 2020.
28. Cheng, R.-S.; Hong, W.-J.; Wang, J.-S.; Lin, K.W. Seamless Guidance System Combining GPS, BLE Beacon, and NFC Technologies. *Sensors* **2016**, *2016*, 1–12. [CrossRef]
29. Zhang, S.; Guo, J.; Luo, N.; Wang, L.; Wang, W.; Wen, K. Improving Wi-Fi fingerprint positioning with a pose recognition-assisted SVM algorithm. *Remote Sens.* **2019**, *11*, 652. [CrossRef]
30. Lee, S.; Kim, J.; Moon, N. Random forest and WiFi fingerprint-based indoor location recognition system using smart watch. *Hum.-Cent. Comput. Inf. Sci.* **2019**, *9*, 6. [CrossRef]
31. Yang, Q.; Zheng, S.; Liu, M.; Zhang, Y. Research on Wi-Fi indoor positioning in a smart exhibition hall based on received signal strength indication. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 1–13. [CrossRef]
32. Khanh, T.T.; Nguyen, V.; Pham, X.-Q.; Huh, E.-N. Wi-Fi indoor positioning and navigation: A cloudlet-based cloud computing approach. *Human-centric Comput. Inf. Sci.* **2020**, *10*, 1–26. [CrossRef]
33. Rahim, A.; Maqbool, A.; Rana, T. Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera. *PLoS ONE* **2021**, *16*, e0247440. [CrossRef] [PubMed]
34. Yang, D.; Yurtsever, E.; Renganathan, V.; Redmill, K.A.; Özgüner, Ü.A. vision-based social distancing and critical density detection system for covid-19. *arXiv* **2020**, arXiv:2007.03578.
35. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6 September 2014, pp. 740–755.
36. Punn, N.S.; Sonbhadra, S.K.; Agarwal, S. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. *arXiv* **2020**, arXiv:2005.01385.
37. Gupta, S.; Kapil, R.; Kanahasabai, G.; Joshi, S.S.; Joshi, A.S. SD-Measure: A Social Distancing Detector. In Proceedings of the 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), Institute of Electrical and Electronics Engineers (IEEE), Bhimtal, India, 25–26 September 2020; pp. 306–311.
38. Cristani, M.; Del Bue, A.; Murino, V.; Setti, F.; Vinciarelli, A. The Visual Social Distancing Problem. *IEEE Access* **2020**, *8*, 126876–126886. [CrossRef]
39. Bhambani, K.; Jain, T.; Sultanpure, K.A. Real-time Face Mask and Social Distancing Violation Detection System using YOLO. In Proceedings of the 2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC), Institute of Electrical and Electronics Engineers (IEEE), Bangalore, India, 8–10 October 2020; pp. 1–6.