Int. J. Nonlinear Anal. Appl. 13 (2022) 2, 2865–2880
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.27431.3598

Social distance in object detection: Survey based on cutting-edge deep learning approach

Hiyam Hashem Saeed*, Abdulbasit Alazzawi

Department of Computer Science, College of Science, University of Diyala, Baqubah, Iraq

(Communicated by Madjid Eshaghi Gordji)

Abstract

Many health systems face extraordinary problems because of the ongoing COVID-19 outbreak and new variations. Multiple regulatory authorities have made it mandatory to maintain a safe distance, particularly in public settings where large groups of people are likely to come into contact, such as sports arenas, public transportation, workplaces as well as shopping malls. Nevertheless, keeping a safe distance (two meters), adjusting multiple model detection errors or accuracy as well as deployment prerequisites, great number of people, facial expression, view angle, low-resolution images, detection model deployment on computers having restricted processing power, and the shortage of a real-world dataset have all made compliance and adherence to proper distancing social difficult. As a result, this survey examines and contrasts the most important past deep learning (DL)-based social distance research. Here, the survey presents a new fine-grained taxonomy that classifies the present state-of-the-art DL-based object detection for detecting distance in terms of several dimensions, such as detection, input data, evaluation methodologies, as well as testing, based on a thorough review. Each facet is then divided into categories based on a variety of factors. In addition, this survey analyses and evaluates the associated experimental techniques suggested as DL-based object detection. Finally, this survey examines DL's role in social distance, object detection datasets impact, as well as the proposed approaches efficacy by assessing the experimental research. The results show that more work is needed to enhance the existing state-of-the-art. Ultimately, open research difficulties are recognized, as well as prospective DL research areas are suggested for future research.

Keywords: Deep learning, COVID19, Convolutional neural network, Social distancing, Object detection
2020 MSC: 68T07

1 Introduction

In many health systems around the world, the emergency of new coronavirus (COVID-19) poses extraordinary obstacles. Due to the epidemic continuing to spread and decimate the population, particularly in weak countries, the World Health Organization (WHO) declared a public health emergency in March 2020 [16]. To fight the pandemic, countries reinforced infection and prevention measures such as enacting a national lockdown, travel ban, isolation, temperature checking, sanitizing routine hand washing, quarantine suspected, including positive cases, wearing face masks, as well as social distancing. Furthermore, some countries have imposed COVID-19 limitations like curfews, national lockdown, the closure of public areas, travel restrictions, border closures, as well as physical separation [30].

*Corresponding author
Email addresses: scicompms2130@uodiyala.edu.iq (Hiyam Hashem Saeed), dr.abdulbasit@uodiyala.edu.iq (Abdulbasit Alazzawi)

Received: February 2022 *Accepted:* April 2022

In developing countries, where infrastructure was degraded, health services were overburdened, money was insufficient, and public health surveillance was limited, these limits posed serious problems. Large-scale restrictions are difficult to establish and to be complied with; as a result, challenging to be practiced and maintained, resulting in sporadic public compliance, particularly when they have a considerable effect on political as well as social norms and the economy [7], as well as the affected population's psychological wellbeing. Following the successful development of vaccines, the focus is now clearly shifting to population immunization. Emerging COVID-19 variations, regular movement of informal traders, permeable borders, as well as the selling of bogus vaccination certificates, nevertheless, continue to jeopardize some countries' efforts toward virus containment. Surveillance systems, for example, could give personal protection against infection by reducing the space between people. Surveillance systems are fairly priced, simple to operate, and effective, despite continuous studies into their effectiveness. Presently, using face masks is essential for reducing transmission chains [32]. Despite the vaccine's efficacy, regulatory authorities have mandated the wearing of face masks in public venues in which large crowds are likely to congregate, such as sporting arenas, public transportation, workplaces, as well as retail malls [47]. However, the masked face protection is not a unique approach that is suggested. Still, another significate approach is social distancing, as recommended by WHO to guarantee health protection for people and avoid the emerging COVID-19 chains from spreading. As a result, all of these are being carried out to prevent re-infection as well, as the COVID-19 spread as countries re-open in the new normal. Social distancing rules have been established in countries including South Africa, the United States, France, China, India, Uzbekistan, Kenya, Spain, Brazil, Lebanon, as well as Qatar [36]. Nevertheless, keeping the safety distance has proven to be a tough procedure due to a variety of causes such as inadequate sanitation, informal settlements [27], social instability, socio-economic problems [28], ignorance, as well as lack of understanding. In particular, developing technologies have been used to combat COVID-19 in multiple environments. Apart from that, the Internet of Things, such as Wi-Fi or Bluetooth, geographic information systems, 5G technology, as well as big data are instances of these technologies [29]. Scientists, technologists, researchers have recently used artificial intelligence (AI) models to level up the pandemic screening process, discover as well as map hotspots and movement patterns in real-time, thermal imaging, model, monitor, predict, screen, as well as diagnose COVID-19 suspected cases [51].

Furthermore, [1] conducted a comprehensive assessment of deep learning (DL) as well as machine learning strategies in detecting COVID-19. Moreover, [17] performed a quick evaluation of AI models used in clinical care for COVID-19 diagnostic and screening. A thorough DL review for COVID-19 prognosis was undertaken by [41]. AI models employed to detect close distancing for people are not examined in any of the published reviews. Aside from that, AI models must be used to detect COVID-19 face masks, thanks to advances in AI approaches. In the areas of picture classification and object detection, AI algorithms have shown promise[43]. As a result, these models must be used to identify distance. As a result, the goal of this research was to offer a complete assessment of AI approaches used to monitor social distance in order to assure COVID-19 social distance compliance and adherence. The following are the primary contributions of this paper:

1. To determine and analyze the state-of-the-art AI (machine learning and DL) models implemented to detect social distance ~~detection~~.
2. To identify and deliberate prediction limitations as well as the accuracy of AI models implemented to monitor the distance.

This work taxonomy and reviews existing works for social distance in object detection, object detection approaches and the most popular. This paper is separated into the following topics to present its significant contributions:

Section 2 discusses the method used to demonstrate the social distance object detection field and the approach used to detect it. Also, the common dataset object detection is explained. Next, section 3 describes a taxonomy for social distance in object detection and includes input data, testing, detection technology, some of the popular object detection issues, and their utilized performance metrics. Next, Section 4 portrays summarization and discussion for current social distance detection to present current issues and trends future.

2 Background of Social Distancing Detection (SDD)

It has been suggested that people retain a minimum of two meters or six feet distance from each other for optimal social distancing against COVID-19. Conventionally, government policy declarations such as the closing of public venues and the prohibition of public meetings and events, including wedding ceremonies and funerals, have been used to achieve social distance. These state-of-the-art tactics, on the other hand, cause not only individuals discomfort but also have a negative influence on their social lives and sources of income. Various technology-based solutions have been developed in this regard to automatically detect individuals who may have come into contact with COVID-19

infected individuals or are in danger of spreading the virus. GPS, Surveillance cameras, Wi-Fi, Computer Vision, deep learning (DL), Bluetooth, Positioning or localization algorithms as well as smartphones are some of the essential technologies that might be used to assure the right and constant use of social distancing [3].

Figure 1 shows the model is formed from three stages or steps to detect the distance between the persons. These steps are explained as follows:

- **Pre-processing Data step**: It processes the input image that may also be a video frame and annotate the actual box for the labeling dataset.
- **Classification Object step**: An object detection algorithm is then employed to detect these images.
- **Localization Object step**: The people or class is localized by the DL model by drawing green boxes on these images once the DL algorithm has been trained.
- **Distance calibration Calculation Step**: Measure the distance between the prediction boxes' top left vertices as the distance between persons. Hence, if the distance exceeds a particular value (we assume 100), the two people are considered to be too close.
- **Alarm System Step**: Finally, a Red Line is created between the prediction boxes' top left vertices that are too close together, signaling that the social distance involving the two people is threatening.

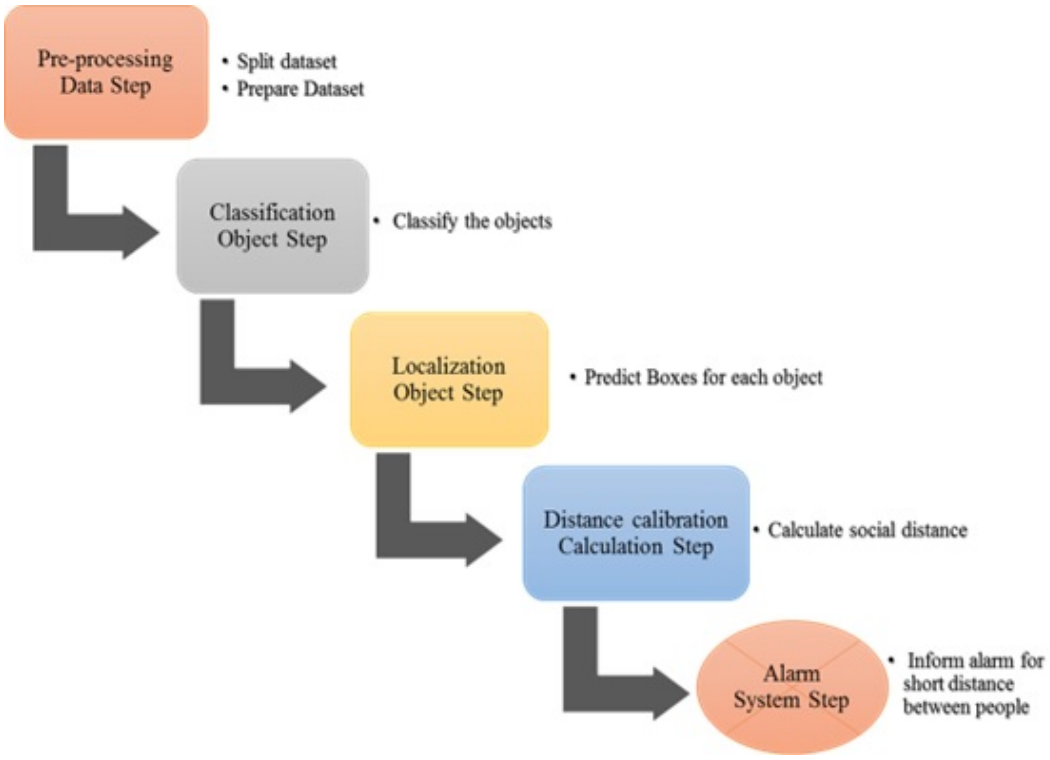

Figure 1: Social Distance Detection Model

3 Taxonomy of a Deep Learning-Based SDD

On the contrary, this survey presents a fine-grained taxonomy that divides the state-of-the-art deep learning (DL)-based social distance in object recognition into four categories. They comprise testing strategy, input data, approaches or technologies strategy, as well as evaluation strategy. In addition, as illustrated in Figure 2, each aspect is categorized according to a set of criteria.

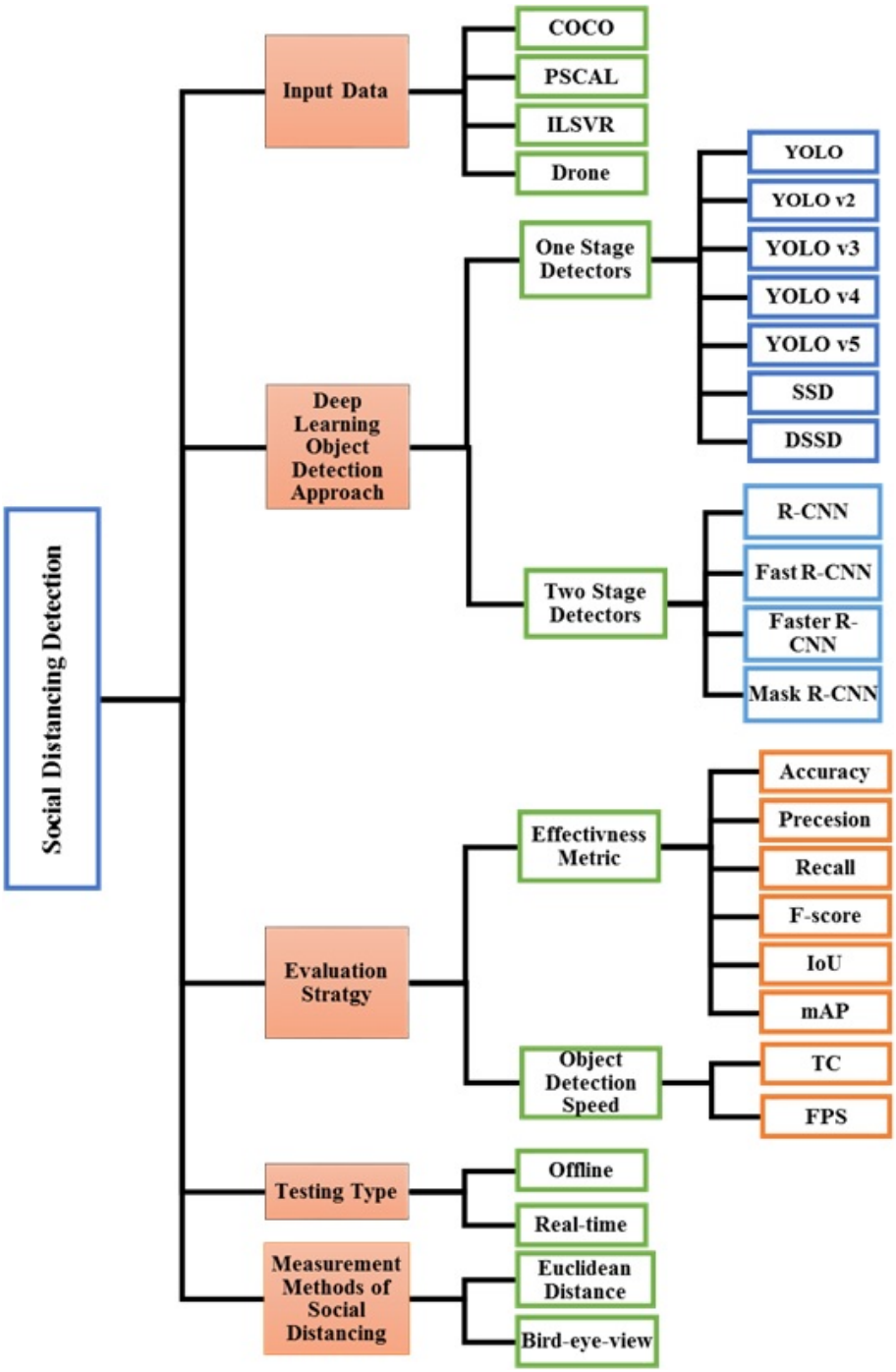

Figure 2: Social Distance Detection Taxonomy

3.1 Input Dataset

An object must be detected by describing that it falls under a particular class as well as finding it in the image. A bounding box is typically employed to depict an object's location. Various research fields gained an advantage from using tough datasets as benchmarks since they allow for a consistent comparison of different algorithms and the solution goals setting. Face detection was the focus of early algorithms, which utilized a variety of ad hoc datasets. Face detection datasets that were more realistic and difficult were later generated. Detecting pedestrians is another popular challenge for which various datasets have been developed. There are 350,000 tagged examples with bounding boxes in the Caltech Pedestrian Dataset [8]. Object detection datasets such as PASCAL VOC [9], MS COCO [24], ImageNet-loc [44] are the most often used yardsticks.

1. PASCAL VOC DATASET
   A multi-year effort was given to the compilation as well as series maintenance of widely adopted benchmark datasets for the basic item types detection from 2005 to 2012. The PASCAL VOC datasets [9] comprise 11,000 images with 20 item categories (in VOC2007, these include bicycles, people, bottles, birds, dogs, and so on). Vehicles, animals, household things, and humans are the four primary branches among the 20 categories. Some of them, such as motorcycles and automobiles, enhance the semantic specificity of the output, yet they do not appear alike. Furthermore, visually similar classes, such as "cat" vs. "dog," make it more difficult to distinguish between them. Over 27,000 bounding boxes for object instances have been tagged, with nearly 7,000 having extensive segmentations. In the VOC2007 dataset, there are unbalanced datasets, with the class "human" being roughly 20 times larger compared to the smallest class "sheep" in the training set. It is worth noting here that the issue is common in the neighboring environment, and how can detectors effectively address it? The subsequent challenge is that the detectors must evaluate different views individually, such as the back, front, right, left, and unspecified.
2. MS COCO DATASET
   There are 91 common object categories in the Microsoft Common Objects in Context (MS COCO) dataset [24] for recognizing as well as segmenting objects seen in ordinary life in their innate surroundings, with 82 of them having more than 5,000 labeled examples. The 20 categories in the PASCAL VOC dataset are covered by these categories. In total, there are 2,500,000 labeled examples in 328,000 images in the dataset. MS COCO collection also considers many points of view, and all of the objects are found in natural settings, providing us with a wealth of contextual information. COCO includes fewer categories but more cases per category than the popular ImageNet dataset [44]. The dataset also has a substantially higher number of cases per category (average of 27k) as opposed to the PASCAL VOC datasets [9] (approximately ten times less compared to the MS COCO dataset) and the ImageNet object detection dataset (1k) [44]. In comparison to PASCAL VOC (2.3) and ImageNet (3.0), MS COCO has significantly more object instances per image (7.7). Moreover, the MS COCO dataset has 3.5 categories per image, as opposed to PASCAL (1.4) as well as ImageNet (1.7), respectively. Furthermore, 10% of images in MS COCO possess only one category, whereas more than 60% of images in ImageNet and PASCAL VOC possess only a single object category. Small objects, as we all know, require more contextual reasoning to recognize. Contextual information abounds in the MS COCO dataset's images. The largest class is "person," which has about 800,000 occurrences, whereas the smallest class is "hair driver," which has just approximately 600 instances in the entire dataset. Apart from that, another minor class, "hair brush," has roughly 800. Except for 20 classes having a large or small number of examples, the other 71 categories have nearly the same number of examples.
3. ImageNet DATASET
   Vision tasks and practical applications might advance with the help of challenging datasets. The ImageNet dataset [44] is another significant large-scale benchmark dataset. Object detection is an ILSVRC job that tests an algorithm's capacity to name and locate all target objects instances in an image. There are almost 450k training images, 20k validation images, as well as 40k test images in ILSVRC2014, which includes 200 object classes.
4. ILSVRC-2017
   ILSVRC-2017 detection dataset [44]. There are 200 object types in this dataset. In addition, there are 456,567 images and 478,807 bounding box annotations near object instances in the dataset for training. The validation dataset includes 20,121 images that are fully labeled with 200 object categories and 55,502 object instances. In addition, [44] defines a category hierarchy, with some objects having many parents. We designate a single parent for every category because we also test the strategy for meta-class detection as well as labeling.
5. VisDrone2018
   A new dataset, VisDrone2018 [55], is a large-scale visual object detection as well as tracking benchmark dataset consisting of photographs and movies recorded by drones in the last two years. On the drone platform, this dataset attempts to improve visual understanding tasks. The benchmark images and video sequences were taken in 14 distinct cities across China, from north to south, in various urban and suburban environments. VisDrone2018 is made up of 263 video clips as well as 10,209 images (no overlap with video clips) having detailed annotations such as truncation ratios, occlusion, object categories, object bounding boxes, and more. In 179,264 images/video frames, this benchmark has over 2.5 million annotated examples. The benchmark allows for broad examination and investigation of visual analysis algorithms on the drone platform because its size is the largest of its kind ever released. The enormous number of small items in VisDrone2018, for instance, bicycles, pedestrians, as well as dense cars, will make certain categories challenging to detect. Furthermore, 82.4% of the training images feature more than 20 objects per image, and the average number of objects per image in the

training set is 54 among 6471 images. Because this dataset covers dark night scenes, the images' brightness is lower during the day. As a result, it is quite challenging to recognize dense and small objects correctly.

6. OPEN IMAGES V5 DATASET
Open Images [23] refers to a 9.2 M images collection that includes object bounding boxes, image-level labels, visual relationships, as well as to object segmentation masks. Moreover, Open Images V5 has 16M bounding boxes for 600 object types on 1.9M images, making it the biggest collection where object position annotations are currently available. To begin, expert annotators (Google-internal annotators) manually drew the boxes in this dataset to ensure correctness and uniformity. Second, the images are extremely varied, with the majority of them containing complicated settings containing multiple items (8.3 per image on average). Next, this information includes visual relationship annotations, such as "woman playing the guitar" and "beer on the table." With 391,073 samples, it has 329 relationship triplets in all. Fourth, segmentation masks are available for 2.8 M object instances across 350 classes in V5. Segmentation masks identify the objects' outline, allowing for a considerably more detailed description of their spatial extent. Finally, 36.5M image-level labels covering 19,969 classes have been added to the dataset.

3.2 Deep Learning Object Detection Approach

DL-based approaches for object detection have recently earned state-of-the-art performance. In addition, these approaches may be divided into two categories: two-stage and one-stage detectors. The former, for instance, You Only Look Once (YOLO), executes image detection without the need for a proposal stage. Faster region-based Convolutional Neural Networks (Faster R-CNN), on the other hand, suggest a group of potential areas first and later use a classifier to produce the final prediction. These two types of detectors have varying detection performance due to changes in processing algorithms. In terms of processing speed, one-stage detectors win out, whereas two-stage detectors win out in terms of categorization and positioning accuracy.

3.2.1 One-Stage Detectors

Through a single-stage, one-stage detectors can directly generate the categorization probability as well as to object position coordinates. Single Shot MultiBox Detector (SSD), YOLO, and others are examples of these detectors. YOLO is available in a variety of forms at the moment. In an end-to-end neural network, the original YOLO model, also known as YOLOv1, anticipated the bounding box coordinates while classifying the objects. The YOLOv2 is an improved version of the YOLOv1 that preserves the speed advantage while adding anchor boxes, batch normalization, and a high-resolution classifier. With 53 convolutional layers in addition trained to ImageNet in YOLOv3, an improved feature extractor was introduced. The YOLOv3 is more accurate than the YOLOv2, but it is slower because of the additional layers. In addition, YOLOv4 and YOLOv5 have been added to achieve cutting-edge detection accuracy. However, because of the more complex network, these methods' detection speed was substantially slowed. Although YOLO made real-time object identification possible, it was still challenging to detect small objects, as well as the bounding box coordinates inaccuracy was significant. To increase detection accuracy, SSD was presented, which incorporated reference boxes and detected the item on multi-scale feature maps. As a result, in Deconvolutional Single Shot Detector (DSSD), the deep residual network (ResNet) was used as the backbone network. Although numerous better SSD methods have been presented, we are unaware of any existing study on YOLO series method enhancements.

3.2.2 YOLOv1

YOLO represents the most recent up-to-date real-time object detection model, which is a single CNN that concurrently forecasts various bounding boxes and classes concerning a single scan of the full image. This extremely fast model was created by Joseph Redmon et al. [38] in the year 2016, and the architecture of the network was inspired via the model of GoogLeNet for classifying images. There are 24 convolutional layers in this network, preceded by two fully linked layers. 1×1 reduction layers were utilized in YOLO succeeded by convolutional layers of 3×3 [18]. Since every grid cell may only anticipate only one class and two boxes, YOLO puts stern spatial limits on bounding box predictions. The amount of surrounding items that the model may anticipate is restricted by this geographic limitation. In addition, the YOLO model also has trouble with little objects that appear in groups, for instance, flocks of birds. The model experienced a challenging time to generalize to objects having unique or new configurations or aspect ratios because it learns to estimate bounding boxes from data. Since this model's design comprises numerous downsampling layers from the input image, it employs rather coarse features for predicting bounding boxes [38].

3.2.3 YOLOv2

In 2017, Joseph Redmon and Ali Farhadi [39] developed the YOLOv2 approach, which has higher accuracy and speed than the YOLOv1 method. The input image is divided into SS grids by the YOLOv2 object detection algorithm. K bounding boxes are predicted by each grid. Each bounding box's class-specific confidence is:

$$Pr(Class_i|Object) * Pr(Object) * IoU_{pred}^{truth} = Pr(Class_i) * IoU_{pred}^{truth},$$

where $Pr(Object) * IoU_{pred}^{truth}$ denotes the confidence that the bounding box contains objects; IoU_{pred}^{truth} resembles the Intersection-over- Union between the predictions as well as the ground truth; $Pr(Class_i|Object)$ refers to the object's conditional probabilities, which fall under C classes. Thus, the YOLOv2 predictions are encoded as an $S \times S \times (K \times (5+C))$ tensor. Also, the YOLOv2 backbone network extracts the object features by the down-sampling convolutional structure, which is equivalent to the VGG network. If CNN propagates forward, the relationship involving the lth layer and the $l-1th$ layer is a function as written below:

$$x^l = f(y^l) = f(x^{l-1} * w^l + b^l).$$

The CNN's lth layer's input is denoted by the letter x^l. $f(.)$ refers to the activation function. $y^l = x^{l-1} * w^l + b^l$ denotes the intermediate variable, in which w^l denotes the convolution kernel's weight, b^l is the bias parameter and $*$ indicates convolution. The gradient of the loss function for a CNN propagating backwards is:

$$\delta^{l-1} = \frac{\partial L}{\partial y^{l-1}} = \frac{\partial L}{\partial y^l} \cdot \frac{\partial y^l}{\partial y^{l-1}} = \delta^l * rot180(w^l) \bigodot \acute{f}(x^{l-2} * w^{l-1} + b^{l-1}),$$

where $rot180(.)$ denotes the weight parameter matrix's 180° counterclockwise rotation. $\bigodot$ represents the Hadamard product while $L(.)$ resembles the loss function. Next, the gradient indicated by the activation functions derivative product and the weight parameters will decrease as the gradient propagates layer by layer in the network. For instance, the Sigmoid activation function's derivative is $\acute{f}(y^{l-1})_{Sigmoid}| \leq 1/4$, and the initialization weights are normally less than 1; the gradient will diminish as the network propagates backward. Finally, the vanishing-gradient issue appears, resulting in reduced detection accuracy [39]. To extract the features, YOLOv2 employs DarkNet-19 as a backbone, which comprises 19 convolutional layers as well as 5 max-pooling layers, as shown in Figure 2. By using multi-scale training and K-means, DarkNet-19 may improve the input image's resolution, eliminate the completely connected layer, as well as learn better boxes for object detection. Darknet19, on the other hand, has a poor feature extraction performance and does not fully exploit multi-scale area features, which restricts future detection accuracy improvements.

3.2.4 YOLOv3

YOLOv3 is the YOLO series' third generation and was originally developed by Joseph Redmon and Ali Farhadi [40]. Instead of employing a recommendation area-based recognition network, it turns target detection into a regression problem that may be positioned and classed simultaneously, allowing for real-time detection. Multi-scale prediction, feature extraction, class prediction as well as bounding box prediction are the YOLOv3 model's four steps. The darknet-53 feature-extraction network, which has 53 convolutional layers, and the YOLO prediction layer, which has low computational costs, where both included in YOLOv3. To store the target characteristics and improve the loss function calculation, YOLOv3 includes a large number of convolution layers 3×3 and 1×1. To correct the flaw in YOLOv1 as well as YOLOv2's previous generation, it incorporates the residual network. It contributes to the Leaky ReLU layers as well as batch normalization (BN) after each convolution layer. Darknet-53's architecture can be seen in Figure 4. Darknet accepts 416×416 pixels as a default input. Outputting 13×13, 26×26, and 52×52-pixel feature maps allows for the detection of multi-scale targets. Downsampling durations are reduced, and smaller targets can be recognized by enhancing the pixel grid density. This model, on the other hand, is significantly more complicated than the previous one. By adjusting the size of the model structure, speed and accuracy can be balanced.

3.2.5 YOLOv4

When contrasted to YOLOv3, YOLOv4 has significantly improved speed and accuracy. However, YOLOv4 simply integrates the methodologies suggested for other models in recent years with YOLO [6]. CSP Darknet53, which joins the cross-stage partial network (CSPNet) paradigm with the residual block in Darknet53, serves as the YOLOv4 network's backbone. In addition, due to its continuous, smooth, non-monotonic, self-regularized properties, the

convolution block's activation function is shifted from Leaky ReLU to Mish. The path aggregation network (PAN), as well as spatial pyramid pooling (SPP), are employed in the neck region of the YOLOv4 network. SPP has the ability to dramatically expand the receptive field while also separating the most important contextual elements. PAN can extract the image's multi-scale characteristics repeatedly. PAN has more flexible ROI pooling than FPN and reduces the fusion path between feature maps in low-level layers and high-level layers [54]. Furthermore, the YOLOv3 head is used in the new model for the detection head, with the goal of predicting objects at numerous scales. When there are three object classes in total, for instance. Therefore, the total number of filters is equal to $(\text{classes} + 5) \times 3 = 24$. PANet, SPP block, CSP Darknet-53, as well as the prediction head are the primary components of YOLOv4's skeleton. To increase the variety of the learned features inside distinct layers, CSP Darknet-53 combines Darknet-53 with CSPNet, which contains the partial transition layer as well as the partial dense block. Aside from that, the SPP block is employed to widen the receptive field and separate the most important context elements without slowing down inference. The detecting head, just like YOLOv3, has three scales. The detecting head in YOLOv4 has the $64 \times 64 \times 24, 32 \times 32 \times 24$, and $16 \times 16 \times 24$ when the inputs are 512×512.

3.2.6 YOLOv5

YOLOv5 is the most recent version of the YOLO family proposed by Jocher Glenn et al. [3] with a number of enhancements, including the use of the Pytorch DL framework, distributed training, mixed precision, trunk network optimization, and so on. It is worth noting that the training time was cut in half, and the detection accuracy increased. YOLOv5 consists mostly of four models: YOLOv5m, YOLOv5s, YOLOv5s, as well as YOLOv5l. Next, the key difference is the amount of feature extraction modules and convolution kernels at certain network sites. In comparison to other models, YOLOv5s possess the smallest network width and depth, the fastest detection speed, as well as the quickest training time. The input, Backbone, Neck, and Head modules make up YOLOv5's network architecture. To extract the features of the destroyed house from the input image, the YOLOv5 Backbone module is used. It is based on the CSPNet, SPP, as well as Focus structures. Convolutional processing is referred to as Conv. CBL is made up of three layers: a batch normalization (BN) layer, a convolutional (Conv) layer, as well as an activation layer utilizing a Leaky ReLU.Meanwhile, Concat is a feature combiner. CSP is an abbreviation for CSPNet [3]. The distinction between the two sorts of structures in CSPNet is the repeated ResUnit numbers. CSP2 possesses more layers, which allows it to extract more details. By dividing the channels equally, the input feature layer is divided into two portions. One portion is converted to a cross-stage hierarchical structure before being concatenated having another to construct the next convolutional layer. Therefore, this structure not only minimizes the model's computations but also delivers a richer features combination as well as increases the detection accuracy and speed. Apart from that, the receptive field can be increased by using the SPP module. A broad receptive field can detect the target data and distinguish essential context features from the input feature layer. Thus, the main goal of the Neck module is to produce the feature pyramid, which allows the model to detect objects at various scales. YOLOv5 employs the Path Aggregation Network (PANet), which facilitates the transfer of low-level features to high-level features via a bottom-up approach based on the Feature Pyramid Network (FPN). Moreover, the Head module has the same function as YOLOv3 and YOLOv4, which may forecast target class probabilities and bounding boxes on three scales $(13 \times 13, 26 \times 26, 52 \times 52)$ [19].

3.2.7 SSD

Wei Liu et al. [25] proposed the SSD model in the year 2015. SSD can address the drawbacks of previous techniques, such as the difficulty of detecting small-scale objects and their inaccurate location. SSD has two key contributions and conducts both operations in a single shot. To begin, feature maps of various sizes ($10 \times 10, 5 \times 5, 3 \times 3$, and so on) are retrieved from the picture, with small-scale feature maps employed to detect large objects, whereas large-scale feature maps employed to detect little ones (multi-reference detection). Second, anchor boxes' aspect ratios and alternative scales are created (multi-resolution detection). Another major benefit of SSD is that it improves speed in low-resolution photos, allowing us to avoid using expensive sensors. In SSD, the VGG-16 network is utilized as a feature extraction network. Therefore, six extra convolution layers are added to the VGG-16 network for detection.When the SSD is given an image and a collection of truth or object labels, it generates a series of feature maps of various scales, preceded by a 3×3 convolution filter on each feature map to generate default bounding boxes. Apart from that, the bounding box offset and class probabilities for each box are predicted simultaneously as the image processes. To obtain the best-predicted bounding box and label, SSD only executes detection at the top layer [14]. SSD reached state-of-the-art performance on several test datasets having few additional data augmentation as well as hard negative mining approaches. SSD, on the other hand, struggled with little objects because of thin layers that lacked deep semantic information.

3.2.8 DSSD

By utilizing a broader network, DSSD was able to increase SSD. ResNet-101, DSSD's deep and robust backbone network, surpassed the VGG network. To offer extra context information, a deconvolutional module was included. More crucially, the deconvolutional layers might be trained throughout the training process, allowing DSSD to be more flexible and achieve better results. K-means clustering is used to aggregate training boxes with squared root box areas as the distance measurement to increase the anchor scales and ratios' accuracy. Although DSSD increased SSD accuracy, particularly for small objects, balancing precision and real-time remains a concern [10].

3.3 Two-Stage Detectors

Detecting objects in two passes is what two-stage detectors imply. The various stages produce a sparse list of regions of interest (RoIs) and classify each one using a network. The two-stage detectors are essentially a collection of R-CNN algorithms that must first produce candidate boxes before classifying and regressing them using CNN. Even Faster R-CNN may only run at a frame rate of 7 Frames per Second (FPS), making real-time detection difficult [35]. R-CNN series algorithms offer great accuracy, but due to the issue of excessive calculation, they can only run at a frame rate of 7 Frames per Second (FPS). To simplify region proposal generation's generation process, the Fast R-CNN and Faster R-CNN methods use the selective search strategy and the region proposal network accordingly.

3.3.1 R-CNN

Driven by CNN's success in image classification, Ross Girshick et al. [12] presented R-CNN, a three-module method that used CNN to extract rich features in object identification tasks for the first time, achieving state-of-the-art performance. It first creates a set of object-independent object proposals in the form of regions in the input image, then uses CNN to extract fixed-length deep features from the image's processed regions. Finally, these features are fed into a series of linear Support Vector Machines (SVMs) that determine the object type [12]. By classifying object proposals using a deep neural network, RCNN provides outstanding object identification accuracy. On the contrary, R-CNN has significant downsides. It is slow in nature since it requires a long time to train the network due to the large number of region proposals that are input into the CNN. Furthermore, the selective search technique is widely regarded as time-consuming.

3.3.2 Fast R-CNN

The Spatial Pyramid Pooling Convolutional Network (SPPNet) is formed on the classic CNN structure as well as adds regions of interest (ROI) pooling layer before the SVM classifier, allowing the network input picture to be any size while the output size remains constant. In comparison to R-CNN, the SPPNet requires only a single operation and has a faster speed and map. The Fast R-CNN, postulated by Ross Girshick [11], is a clean and quick update of R-CNN and SPPnet. Fast R-CNN is predicated upon SPPNet, and instead of using SVM classifiers, it employs neural network classification. This allows it to train box regression, judgment category, neural network as well as feature extraction, all at the same time. A single input image and many ROIs are fed into a fully convolutional network in the rapid R-CNN architecture. Fully connected layers combine each ROI into a fixed-size feature map, which is subsequently transferred to a feature vector (FCs). Softmax probabilities and per-class bounding-box regression offsets are the two output vectors per ROI generated by the network. A multi-task loss is used to train the architecture end-to-end [11]. Fast R-CNN detects small objects with a larger feature map; however, because the larger feature map has fewer convolutional layers and there exists an issue with insufficient feature extraction, the technique is insensitive to small objects.

3.3.3 Faster R-CNN

The detection accuracy of Faster R-CNN is enhanced on R-CNN's basis [42]. Since the four steps of target detection are given over to deep neural networks and executed on a GPU, Faster R-CNN is extremely fast. This has the potential to dramatically improve operational efficiency. RPN network and Fast R-CNN are the two sections of the model. To get the feature map of the image, the input image is first subjected to convolution and pooling operations via the basic feature extraction network, and then the feature map is passed to the RPN network, which performs preliminary border regression and classification judgment. Whether the candidate frame is the backdrop or the object to be recognized is used to classify it. The candidate frame's position and score information is output by the RPN network and then sent to the Fast R-CNN network for final processing by the fully connected layer. They are the frame's final regression and the item to be recognized's unique classification. To be clear, RPN is used by Faster R-CNN to identify

as well as to detect targets in proposals. RPN's main idea is to employ CNN to create regional recommendation candidates directly. RPN generates the region proposal based on the convolution feature map output by the last shared convolution layer using the sliding window approach. In addition, the RPN network maps the export features to the shared fraction of R-final CNN's output layer. That is, a sliding window operation on the feature map matrix using a 3×3 pixels' window is utilized to get a candidate frame, which is referred to as an anchor frame here. For these candidate frames, the RPN network first receives the target's score as well as the background. The candidate frame is regressed based on the score [50].

3.3.4 Mask R-CNN

By inserting a branch to the Faster R-CNN structure to predict the object mask, Kaiming He et al. in [15] introduced an accurate object identification and segmentation approach dubbed Mask R-CNN. The ResNet is paired with the feature pyramid network (FPN) as the feature extraction network in this approach, which strengthens the basic network. It presents the ROIAlign technique to improve ROI Pooling and thereby solve the misalignment problem. In the meantime, the network incorporates the concept of fully convolutional networks (FCN) and adds a mask module to complete the identified object's instance segmentation. The addition of this new branch added a minor amount of computational complexity, but it can create pixel-to-pixel alignment between the network's input and output, resulting in good results. The Figure 12 in [53] depicts the Mask R-CNN workflow, whereas the Figure 13 in [53] depicts the structures of the FPN network and the ROIAlign implementation method.

3.4 Testing Types

- **Offline**: The dataset can be recorded by a special camera with a special default setting for training and testing the object detection algorithms.
- **Real-Time(Online)**: Refer to the real measurement of the performance of DL in the object detection systems. The testing is done by cameras to detect the objects in real life.

3.5 Evaluation Strategy

DL in object detection is the same as any algorithms other domain as security, medical. So, the evaluation is an important part of measuring the effectiveness and process speed for DL object detection algorithms.

3.6 Evaluation Metrics

The AP is the most popular metric used to quantify the accuracy of detections among various annotated datasets employed by object detection competitions and the scientific community. Before we look at the many types of AP, let's go over some common notions. The ones defined below are the most basic:

- **True positive (TP)**: Detection of a ground-truth bounding box that is accurate.
- **False-positive (FP)**: A non-existent object is detected incorrectly, or an existing object is detected incorrectly.
- **False-negative (FN)**: A ground-truth bounding box that has not been detected yet. It is worth noting that a true negative (TN) result is not applicable in the context of object detection due to an endless number of bounding boxes that should not be detected in any given image
- **Precision** refers to a model's ability to recognize only important points. It is the percentage of optimistic forecasts that are correct
- **Recall** denotes a model's capacity to locate all relevant cases (all ground-truth bounding boxes).
- **mean AP (mAP)** represents a metric for evaluating object detector accuracy across all classes in a database.
- **Confidence score** refers to the chance that an anchor box includes an object. A classifier is often used to forecast it.
- **Intersection over Union (IoU)** refers to the intersection area, which is divided by the area of the union of a predicted bounding box (Bp) and a ground-truth box (Bgt) to get Intersection over Union (IoU). Note that the parameters for establishing whether a detection is a true positive or a false positive are both confidence score and IoU.

3.7 Object Detection Speed

- **Frame Per Second (FPS)**: The videos we watch are formed by a series of still images. Since the difference between each still image is very small. "Frame Per Second" or the so-called "fps" means how many still images frames in the per-second video. As a result, FPS specifies how quickly an object detection model processes video and produces the necessary output.
- **Time Complexity (TC)**: It is the length of time it takes for an algorithm to run as a function of the input length. It measures how long every code statement in an algorithm takes to execute.

3.8 Measurement Methods of Social Distancing

4 Discussion and Findings

Since the start of the decade, researchers have been researching deep learning (DL) approaches for detecting social distances. Articles published throughout the years 2017 and 2022 are considered in this study. For every type of architecture, Figure 3 depicts the discussed DL-based object detection distribution across years. We observed that the researched designs use CNN methods in their research. The sections that follow go into many features of the proposed remedies as well as their outcomes. We address the suggested SDD's deep learning algorithms and datasets, as well as the efficacy and efficiency of methodology-based solutions.

4.1 Deep Learning approach in SDD

DL approaches have been shown to be effective in detecting complicated interactions within raw data at several abstraction levels without the need for human intervention. For the object detection challenge, deep learning algorithms were applied. One group of discussed solutions used two stages of DL called Faster-RCNN [48, 52], the other group of solutions more used the YOLOv1, YOLOv3 and SDD algorithms [4, 20, 21, 26, 33, 45, 46, 52] as illustrated in Figure 2. The most common Yolov3 is used to detect the distance between the people than other algorithms in both stages of SDD. Yolov4 is achieved the best accuracy 100 rates with the coco dataset in social distance detection, while Faster-RCNN is a good level accuracy 0.96 and 0.94 for accuracy and mAP, respectively. Also, a lower level of accuracy is achieved by the Yolov3 algorithm of one stage.

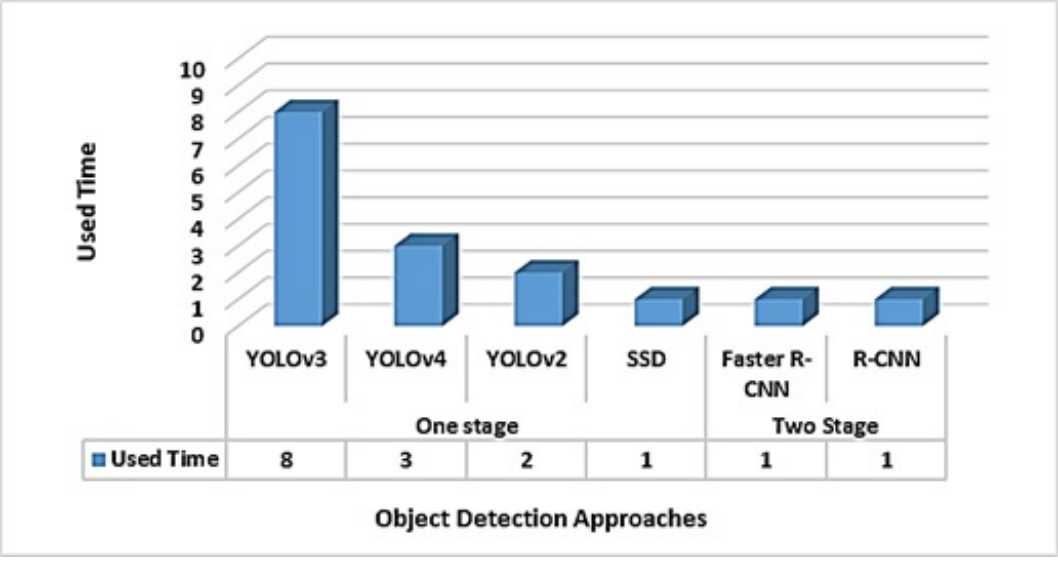

Figure 3: Distribution of Object Detection Algorithms for SDD

4.2 Social Distance Measurement Methods

The social distances are measured by different methods such as Euclidean, neighbors, and the bird's eye view method. Euclidean used most methods to calculate the distance between the people, while the birds-eye view method is less used for measuring the distance between pedestrians the as top-down view by drone as illustrated in Figure 4.

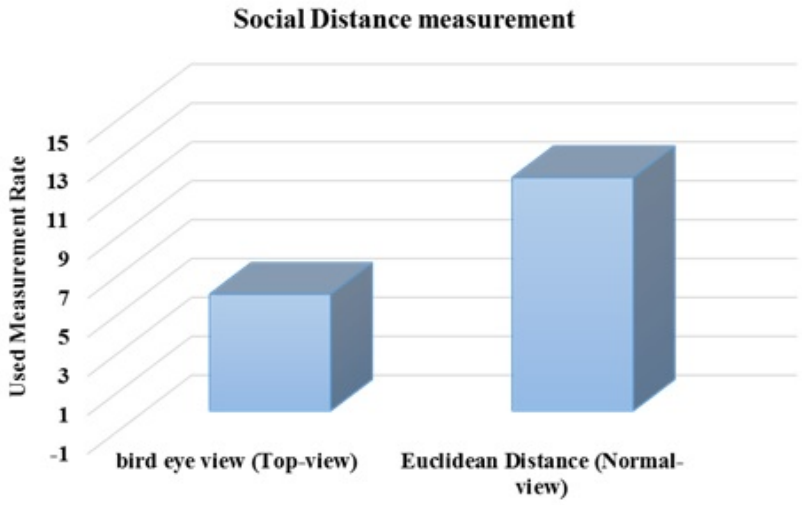

Figure 4: Frequency of Social Distance Measurement Methods

4.3 Dataset and Testing Strategy

The dataset is an important factor for training and testing that can affect SDD's effectiveness as well as efficiency. PASCAL-VOC and its upgraded coco version were utilized in the bulk of the proposed SDD. As indicated in Table. 1, the majority of proposed solutions employed benchmark datasets, whereas only a few works used their own datasets from simulated or real-world settings. As a result, due to dataset restrictions, current SDD solutions do not give a sufficiently general model for real-world use. For evaluating DL in object detection, the type of testing is critical. Offline testing was considered in several studies by dividing the dataset into training and testing sections, with 70% and 30% testing, accordingly—other SDD models proposed benchmarking datasets for training and online (real-time) to test their SDD models.

4.4 Challenges and Future Directions

Following are some of the insights gained and the most relevant study directions for upcoming research according to the findings from Section 4:

4.4.1 Ineffective methodologies

The use of ensemble and hybrid CNN architectures in object detection is yet underexplored, and more research is needed.

4.4.2 Inaccurate Distance

the evaluation metrics are used to measure the ability of surveillance system detection, and each object detection dataset has an IoU value to match the predicted area (Box) and bounding box for the object. With respect to effectiveness as well as efficiency, multiple DL algorithms' evaluation undertaken in isolation does not offer a fair comparison. This is owing to the fact that there are a wide range of (1) the used dataset, (2) the dataset portion that is adopted, (3) pre-processing, (4) deep network configuration, as well as (5) hardware platforms. In order to acquire a fair comparison result, further comparative experimental research through a unified computing platform and shared impacting factors for diverse DL architectures are needed. Furthermore, A few works mentioned the IoU values other metrics used for measuring the effectiveness of CNNs versions. The mAP was proposed to evaluate CNN's performance in object detection. Few studies evaluated the SDD model-based mAP, FPS, training and testing time. As a result, the detected distance by SDD model between people may be incorrect, as established by the World Health Organization.

4.4.3 Unrealistic Dataset

For starters, the benchmark datasets were created two years ago and did not reflect contemporary conditions such as aerial camera angles. Moreover, the benchmark datasets do not possess real-time challenge features. Nevertheless,

because of their availability and the difficulty of getting genuine system traffic or developing simulated environments, PASCAL is still used as a training dataset by the research community.

Table 1: Survey of Social Distance Detection Models

Ref.	Methodology	Social Distance measurement	Dataset	Testing Strategy		Evaluation Metrics						
				Offline	Real-time	Accuracy	Recall	F-score	mAP	FPS	Training time (Sec)	Testing time
[2]	Faster-RCNN	bird eye view	Data Private	video	×	0.96	0.92	0.94	0.95	×	×	×
[34]	YOLOv3	Euclidean distance	Image Net (Normal view)	×	Video	×	×	×	0.84	23	5659s	×
[48]	YOLOv3	bird eye view	Private Data (Top-view)	×	video	×	×	×	×	×	×	×
[52]	Faster R-CNN, YOLOv4	bird-eye-view	Oxford Town Center (Top-view)	×	video	0.92	0.95	×	0.95	×	×	×
[26]	YOLO v3	bird eye view	Oxford Town Center (Top-view)	video	×	×	×	×	×	51	×	0.24
[33]	YOLO v3	bird eye view	Private dataset (Top-view)t	×	video	×	×	×	0.85	×	×	×
[21]	YOLO v3	Euclidean Distance	PASCAL-VOC, COCO, (Normal view)	×	video	×	×	×	0.84	22	×	×
[20]	CNN	Euclidean Distance	Auxiliary Data (Normal view)	video	×	×	×	×	0.92	×	×	×
[45]	YOLOv2	Euclidean distance	Private Data (Normal view)	×	video	0.96	0.96	×	0.95	27	×	×
[46]	PeleeNet	bird eye view	Private Data (Normal view)	×	video	×	×	×	0.88	76	×	×
[4]	CNN	Euclidean distance	Private Data (Normal view)	images	×	0.98	×	×	×	×	34.34	1.34
[5]	YOLOv4	Euclidean distance	Private Data (Normal view)	×	video	×	×	×	0.94	38	×	×
[49]	YOLO v3	DBSCAN	Private Data (Normal view)	×	video	0.91	×	×	0.90	41	5.42	1.92
[22]	YOLO v3	Euclidean distance	Private Data (Normal view)	×	video	×	×	×	×	×	×	×
[35]	SSD	Euclidean distance	Private Data (Normal view)	image	×	×	×	×	0.88	×	×	×
[31]	YOLO v3	Euclidean distance	Private Data (Normal view)	video	×	×	×	×	×	×	×	×
[13]	R-CNN	Euclidean distance	MS COCO, Private Data (Normal view)	×	video	0.94	0.83	×	0.86	×	×	×
[37]	YOLO v4	Euclidean Distance	COCO (Normal view)	image	×	×	0.73	×	1.00	×	×	×

5 Conclusion

Object identification technology based on deep learning (DL) has been rapidly evolved for various applications as powerful computing equipment has been upgraded. Monitoring for social distancing appears to be a new sector of object detection based on DL algorithms, according to this study. The study in this area is still at a very nascent stage, and very few studies have been done to explore the use and adoption to achieve their objectives. Most studies are not used benchmarking testing for detecting the distance between people, especially in real-time. The final purpose of this assignment is to construct a succession of directions, such as Faster-CNN, CNN's, as well as Yolov3 algorithms, to gain efficiency as well as accuracy detectors by extracting rich information and utilizing good representations. Furthermore, euclidean distance is more commonly employed than bird's-eye-view methods to determine the distance between people. Despite the fact that this domain has recently been successful, there is still a lot of potential for improvement. Future studies will focus on improving object detection algorithms to resolve negative data imbalances and enhance localization accuracy, which might be utilized to improve reliable and accurate distance measurement. With the increasingly vital need to keep the distance social in a big area, we recommend using camera technology of drone or CCTV for monitoring and detecting the abnormal distance for a pedestrian in the big area as stadiums or markets in real-time.

References

[1] M. Abboah-Offei, Y. Salifu, B. Adewale, J. Bayuo, R. Ofosu-Poku and E.B.A. Opare-Lokko, *A rapid review of the use of face mask in preventing the spread of COVID-19*, Int. J. Nurs. Stud. Adv. **3** (2021), 100013.

[2] I. Ahmed, M. Ahmad, J.J.P.C. Rodrigues, G. Jeon and S. Din, *A deep learning-based social distance monitoring framework for COVID-19*, Sustain. Cities Soc. **65** (2021).

[3] U.R. Alo, F.O. Nkwo, H.F. Nweke, I.I. Achi and H.A. Okemiri, *Non-pharmaceutical interventions against COVID-19 pandemic: Review of contact tracing and social distancing technologies, protocols, apps, security and open research directions*, Sensors **22** (2022), no. 1, 280.

[4] M.A. Ansari and D.K. Singh, *Monitoring social distancing through human detection for preventing/reducing COVID spread*, Int. J. Inf. Technol. 13 (2021), no. 3, 1255–1264.

[5] K. Bhambani, T. Jain and K.A. Sultanpure, *Real-time face mask and social distancing violation detection system using YOLO*, Proc. B-HTC 2020 - 1st IEEE Bangalore Humanit. Technol. Conf., 2020.

[6] A. Bochkovskiy, C.-Y. Wang and H.-Y.M. Liao, *Yolov4: Optimal speed and accuracy of object detection*, ArXiv Prepr. arXiv2004.10934, (2020).

[7] N. Byrd and M. Białek, *Your health vs. my liberty: Philosophical beliefs dominated reflection and identifiable victim effects when predicting public health recommendation compliance during the COVID-19 pandemic*, Cognition **212** (2021), 104649.

[8] P. Dollar, C. Wojek, B. Schiele and P. Perona, *Pedestrian detection: An evaluation of the state of the art*, IEEE Trans. Pattern Anal. Mach. Intell. **34** (2011), no. 4, 743–761.

[9] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn and A. Zisserman, *The pascal visual object classes (voc) challenge*, Int. J. Comput. Vis. **88** (2010), no. 2, 303–338.

[10] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi and A.C. Berg, *Dssd: Deconvolutional single shot detector*, arXiv Prepr. arXiv1701.06659, (2017).

[11] R. Girshick, *Fast R-CNN*, in Proceedings of the IEEE international conference on computer vision, (2015), 1440–1448.

[12] R. Girshick, J. Donahue, T. Darrell and J. Malik, *Rich feature hierarchies for accurate object detection and semantic segmentation*, Proc. IEEE Conf. Comput. Vision Pattern Recog., 2014, pp. 580–587.

[13] S. Gupta, R. Kapil, G. Kanahasabai, S.S. Joshi and A.S. Joshi, *SD-measure: A social distancing detector*, Proc. 12th Int. Conf. Comput. Intell. Commun. Networks, CICN, 2020, pp. 306–311.

[14] M. Haris and A. Glowacz, *Road object detection: a comparative study of deep learning-based algorithms*, Electron. **10** (2021), no. 16, 1932.

[15] K. He, G. Gkioxari, P. Doll and R. Girshick, *Mask R-CNN*, Proc. IEEE Int. Conf. Comput. Vision, 2017, pp. 2961–2969.

[16] A.N. Ikwu, *The impact of covid-19 pandemic on Africa's healthcare system and psychosocial life*, Eur. J. Nat. Sci. Med. **4** (2021), no. 1, 39–50.

[17] M.M. Islam, F. Karray, R. Alhajj and J. Zeng, *A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19)*, IEEE Access **9** (2021), 30551–30572.

[18] Y. Jamtsho, P. Riyamongkol and R. Waranusast, *Real-time license plate detection for non-helmeted motorcyclist using YOLO*, Ict Express **7** (2021), no. 1, 104–109.

[19] Y. Jing, Y. Ren, Y. Liu, D. Wang and L. Yu, *Automatic extraction of damaged houses by earthquake based on improved YOLOv5: A case study in Yangbi*, Remote Sens. **14** (2022), no. 2, 382.

[20] R. Keniya and N. Mehendale, *Real-time social distancing detector using socialdistancingNet-19 deep learning network*, Available at SSRN 3669311, (2020).

[21] M.Z. Khan, M.U.G. Khan, T. Saba, I. Razzak, A. Rehman and S.A. Bahaj, *Hot-spot zone detection to tackle Covid19 spread by fusing the traditional machine learning and deep learning approaches of computer vision*, IEEE

Access **9** (2021), 100040–100049.

[22] G.S. Kumar and S.D. Shetty, *Application development for mask detection and social distancing violation detection using convolutional neural networks*, ICEIS 2021–23rd Int. Conf. Enterprise Info. Syst. **1** (2021), 760–767.

[23] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov and T. Duerig, *The open images dataset v4*, Int. J. Comput. Vision **128** (2020), no. 7, 1956–1981.

[24] 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*, Eur. Conf. Comput. Vision, 2014, pp. 740–755.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu and A.C. Berg, *Ssd: Single shot multibox detector*, Eur. Conf. Comput. Vision, 2016, pp. 21–37.

[26] R. Magoo, H. Singh, N. Jindal, N. Hooda and P.S. Rana, *Deep learning-based bird eye view social distancing monitoring using surveillance video for curbing the COVID-19 spread*, Neural Comput. Appl. **33** (2021), no. 22, 15807–15814.

[27] E. Mbunge, *Integrating emerging technologies into COVID-19 contact tracing: Opportunities, challenges and pitfalls*, Diabetes Metab. Syndr. Clin. Res. Rev. **14** (2020), no. 6, 1631–1636.

[28] E. Mbunge, *Effects of COVID-19 in South African health system and society: An explanatory study*, Diabetes Metab. Syndr. Clin. Res. Rev. **14** (2020), no. 6, 1809–1814.

[29] E. Mbunge, B. Akinnuwesi, S.G. Fashoto, A.S. Metfula and P. Mashwama, *A critical review of emerging technologies for tackling COVID-19 pandemic*, Hum. Behav. Emerg. Technol. **3** (2021), no. 1, 25–39.

[30] E. Mbunge, S.G. Fashoto, B. Akinnuwesi, A. Metfula, S. Simelane and N. Ndumiso, *Ethics for integrating emerging technologies to contain COVID-19 in Zimbabwe*, Hum. Behav. Emerg. Technol. **3** (2021), no. 5, 876–890.

[31] S. Meivel, N. Sindhwani, R. Anand, D. Pandey, A.A. Alnuaim, A.S. Altheneyan, M.Y. Jabarulla and M.E. Lelisho, *Mask detection and social distance identification using internet of things and faster R-CNN algorithm*, Comput. Intell. Neurosci. **2022** (2022).

[32] K. Ng, B.H. Poon, T.H. Kiat Puar, J.L. Shan Quah, W.J. Loh, Y.J. Wong, T.Y. Tan and J. Raghuram, *COVID-19 and the risk to health care workers: A case report*, Ann. Int. Med. **172** (2020), no. 11, 766–767.

[33] D. Pandit, S. Chougule, H. Fatepurwala, A. Kulkarni, N. Kakade and A. Sundge, *Generalized method to validate social distancing using median angle proximity methodology*, Proc. 3rd Int. Conf. Intell. Sustain. Syst. ICISS, 2020, pp. 279–284.

[34] 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*, arXiv preprint arXiv:2005.01385, (2020) 1–10.

[35] J. Qin and N. Xu, *Reaserch and implementation of social distancing monitoring technology based on SSD*, Procedia Comput. Sci. **183** (2021), 768–775.

[36] S. Rab, M. Javaid, A. Haleem and R. Vaishya, *Face masks are new normal after COVID-19 pandemic*, Diabetes Metab. Syndr. Clin. Res. Rev. **14** (2020), no. 6, 1617–1619.

[37] A. Rahim, A. Maqbool and T. Rana, *Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera*, PLoS One **16** (2021), no. 2, 1–19.

[38] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, *You only look once: Unified, real-time object detection*, Proc. IEEE Conf. Comput. Vision Pattern Recogn., 2016, pp. 779–788.

[39] J. Redmon and A. Farhadi, *YOLO9000: Better, faster, stronger*, Proc. IEEE Conf. Comput. Vision Pattern Recogn., 2017, pp. 7263–7271.

[40] J. Redmon and A. Farhadi, *Yolov3: An incremental improvement*, arXiv Prepr. arXiv1804.02767, (2018).

[41] H.S. Rekha, H.S. Behera, J. Nayak and B. Naik, *Deep learning for covid-19 prognosis: A systematic review*, Intell. Comput. Control Commun. (2021), 667–687.

[42] S. Ren, K. He, R. Girshick and J. Sun, *Faster R-CNN: Towards real-time object detection with region proposal networks*, Adv. Neural Inf. Process. Syst. **28** (2015).

[43] B. Roy, S. Nandy, D. Ghosh, D. Dutta, P. Biswas and T. Das, *MOXA: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks*, Trans. Indian Natl. Acad. Eng. **5** (2020), no. 3, 509–518.

[44] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein and A.C. Berg, *Imagenet large scale visual recognition challenge*, Int. J. Comput. Vis. **115** (2015), no. 3, 211–252.

[45] S. Saponara, A. Elhanashi and A. Gagliardi, *Implementing a real-time, AI-based, people detection and social distancing measuring system for Covid-19*, J. Real-Time Image Process. **18** (2021), no. 6, 1937–1947.

[46] Z. Shao, G. Cheng, J. Ma, Z. Wang, J. Wang and D. Li, *Real-time and accurate UAV pedestrian detection for social distancing monitoring in COVID-19 pandemic*, IEEE Trans. Multimed. **24** (2021), 1–16.

[47] S.W. Sim, K.S.P. Moey and N.C. Tan, *The use of facemasks to prevent respiratory infection: A literature review in the context of the health belief model*, Singapore Med. J. **55** (2014), no. 3, 160.

[48] M. Sriharsha, S. Jindam, A. Gandla and L.S. Allani, *Social distancing detector using deep learning*, Int. J. Recent Technol. Eng. **10** (2022), no. 5, 146–149.

[49] S. Srinivasan, R. Rujula Singh, R.R. Biradar and S.A. Revathi, *COVID-19 monitoring system using social distancing and face mask detection on surveillance video datasets*, Int. Conf. Emerg. Smart Comput. Informatics, ESCI, 2021, pp. 449–455.

[50] Y. Su, D. Li and X. Chen, *Lung nodule detection based on faster R-CNN framework*, Comput. Methods Programs Biomed. **200** (2021), 105866.

[51] R. Vaishya, M. Javaid, I.H. Khan and A. Haleem, *Artificial intelligence (AI) applications for COVID-19 pandemic*, Diabetes Metab. Syndr. Clin. Res. Rev. **14** (2020), no. 4, 337–339.

[52] 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*, Sensors (Basel) **21** (2021), no. 13.

[53] C. Yu, Z. Hu, R. Li, X. Xia, Y. Zhao, X. Fan and Y. Bai, *Segmentation and density statistics of mariculture cages from remote sensing images using mask R-CNN*, Inf. Process. Agric. In Press, (2021).

[54] Z. Zhang, Y. Li, W. Wu, H. Chen, L. Cheng and S. Wang, *Tumor detection using deep learning method in automated breast ultrasound*, Biomed. Signal Process. Control **68** (2021), 102677.

[55] P. Zhu, L. Wen, X. Bian, H. Ling and Q. Hu, *Vision meets drones: A challenge*, arXiv Prepr. arXiv1804.07437, (2018).