OXFORD CDE

Journal of Computational Design and Engineering, 2022, 9(1), 187–200

https://doi.org/10.1093/jcde/qwab075
Journal homepage: www.jcde.org

RESEARCH ARTICLE

Anomaly detection with vision-based deep learning for epidemic prevention and control

Hooman Samani[1], Chan-Yun Yang[2], Chunxu Li[1,3,*], Chia-Ling Chung[2] and Shaoxiang Li[3]

[1]School of Engineering, Computing and Mathematics, University of Plymouth, Plymouth, Devon, PL48AA, UK; [2]Department of Electrical Engineering, National Taipei University, Sanxia, New Taipei City, 23741, Taiwan and [3]College of Environment and Safety Engineering, Qingdao University of Science and Technology, Qingdao, Shandong, 266000, China

*Corresponding author. E-mail: chunxu.li@plymouth.ac.uk

Abstract

During the COVID-19 pandemic, people were advised to keep a social distance from others. People's behaviors will also be noticed, such as lying down because of illness, regarded as abnormal conditions. This paper proposes a visual anomaly analysis system based on deep learning to identify individuals with various anomaly types. In the study, two types of anomaly detections are concerned. The first is monitoring the anomaly in the case of falling in an open public area. The second is measuring the social distance of people in the area to warn the individuals under a short distance. By implementing a deep model named You Only Look Once, the related anomaly can be identified accurately in a wide range of open spaces. Experimental results show that the detection accuracy of the proposed method is 91%. In the social distance, the actual social distance is calculated by calculating the plane distance to ensure that everyone can meet the specification. Integrating the two functions and implementing the environmental monitoring system will make it easier to monitor and manage the disease-related abnormalities on the site.

Keywords: robotics for pandemics; anomaly detection; social distance; deep learning; computer vision; epidemic prevention and control

1. Introduction

The outbreak of COVID-19 has become a pandemic, affecting almost all continents. At the same time, the number of confirmed cases in Asia and other European countries has increased sharply. As of 3 June 2021, more than 170 million people have been infected, and nearly 3.7 million have died (Dong *et al.*, 2020). Although vaccines are available in all countries, the mortality rate has not slowed down. Therefore, global efforts are needed to break the transmission chain of the virus. In this context, the demand for epidemic prevention and control is more important. In recent years, deep learning has developed rapidly. It can build and simulate the human brain's neural network for analysis and learning and input the perceptual data into the deep neural network. The input data are classified, grouped, translated, marked, and designed for model recognition. Learning a deep nonlinear network structure can achieve the approximation of complex functions, represent the distributed representation of input data, and have the strong ability to learn the basic features of datasets from a small number of sample sets. In the past, some robots have been used by auxiliary medical personnel, such as Fig. 1, to carry the automated external defibrillator and coronavirus detection kits (Samani & Zhu, 2016). Doctors can communicate in two-way through robots equipped with

Received: 28 July 2021; **Revised:** 27 October 2021; **Accepted:** 9 November 2021

Figure 1: Ambulance robot (Samani & Zhu, 2016).

audiovisual channels, provide assistance in inpatient medical care, and avoid doctor–patient contact, and they can even operate robots remotely. In addition, deep learning in computer vision and speech recognition and more social applications also have massive improvement. Artificial intelligence (AI) can effectively reduce economic damage and use AI to create intelligent epidemic prevention measures in the face of the influx of viruses. Among them, AI and robots will play an important role during the pandemic.

As the epidemic is more difficult to control, the health risk of medical personnel who contact the infected people is also increasing. Various AI-based technologies are hence introduced to manage and reduce the risk, e.g. unmanned aerial vehicles with a camera that support facial recognition from the air (Hsu & Chen, 2017), fusing face recognition and temperature measuring in a thermal infrared camera (Xie *et al.*, 2017), and numerous regionally official apps in tracing out the possible contact with an infected individual (Ferretti *et al.*, 2020). However, AI technologies provide a variety of advanced functions in helping to identify the suspicious hot points (Bragazzi *et al.*, 2020) with the disease and to find a new treatment method to track the spread of the disease (Chung *et al.*, 2020).

However, AI provides more advanced technologies to help identify coronavirus symptoms and find new treatment methods to track the spread of the disease. At the same time, robots make it easier to interact with patients and treat patients. Therefore, the deep learning combined with anomaly detection has been employed, which can be carried on a monitoring system or robot to assist human screening (Li *et al.*, 2017). It can effectively avoid the problem and reduce the infection rate of direct contact between medical staff and patients, as the first line of defense of epidemic-prevention mechanism. Based on the above, YOLO (You Only Look Once) has been chosen as the first anomaly detection system. The feature of YOLO is that convolutional neural networks (CNNs) can judge the category and position of objects in it only by looking at the picture once, which greatly improves the identification speed. The advantage of YOLO is that the single network design can determine the position of the bounding box and each bounding box. The whole network design is end-to-end, easy to train, and fast.

Although there are people on the scene to monitor and control the spread of the epidemic, it is still difficult to check every abnormal activity of everyone on the scene because people are easily fatigued after long-term work. Moreover, the blind spot of the sight makes it easy to ignore some activities. Therefore, the study aims to substitute the role of human beings with a robotic agent not only in reducing the need for human resources but also in lessening the false detections caused by the negligence of human beings (Chen *et al.*, 2016). Avoiding direct contact with the suspect infected case using the robotic surveillance vision is also a good feature of the developed system to prevent epidemic spread. Indeed, the evolution of deep learning technology could be a good tool for implementation to fulfil the purpose. Additionally, a social distance measurement function to conform to the safety distances, which all the individuals on the site stand for, can be carried out along with the developed system. Summing all the characteristics, the system design can be featured out as follows:

1. Real-time monitoring of abnormal behaviors: A deep learning method is used to detect abnormal features, and an online version of the method is implemented to show the anomaly immediately.
2. Measuring social distances among individuals: By positioning people in the venue, distances among people could be calculated to check the safety along with the social distance suggestion.
3. Saving manpower by introducing robotic agents: An automated monitoring system could take over the role of personnel patrol.
4. Controlling and preventing the epidemic: By avoiding the spread of the virus caused by human contact, the automatic monitoring system can effectively reduce the risk of infection and the need for medical resources.
5. Potential for anytime and anywhere deployment: The robotic application could be maintained as a long-lasting monitor system to substitute mankind, especially suitable to establish as a fundamental infrastructure to block the loopholes in epidemic prevention.

2. Related Works

2.1 Anomaly detection based on machine learning

Deep anomaly detection (DAD) technology has exceptionally been proposed for the performance challenge in response to the rampant performance that a traditional algorithm has satisfied. DAD can learn and distinguish features automatically from data with a great reduction in calculations. This automatic feature extraction in DAD gets a big improvement from that manually handled. Today, DAD has been introduced in many practical life applications, such as video surveillance (Kiran *et al.*, 2018), health care (Schlegl *et al.*, 2017), social networks (Kwon *et al.*, 2019), and sensor networks (Mohammadi *et al.*, 2018). According to the classifications in its fundamental methodology, DAD frameworks can be divided into three categories: supervised, unsupervised, and hybrid learning models. DAD techniques based on learning targets were categorized mainly into two categories, deep hybrid model (DHM) and one-class neural network (OC-NN; Fig. 2). The method of DHM uses a deep neural network as a feature extractor to input the autoencoder hidden represented features into the traditional anomaly detection algorithm (Andrews *et al.*, 2016). By contrast, the OC-NN method combines the deep neural network for extracting features from abundant data to represent single class targets. The innovation of the OC-NN method is that the data representation behind the hidden layers is driven by the OC-NN target so that the anomaly detection can be customized (Krizhevsky & Hinton, 2009).

Cui *et al.* (2011) presented a new method to detect abnormal behaviors in human groups. The approach effectively modelled group activities based on social behavior analysis. An interaction energy potential function was proposed to represent the

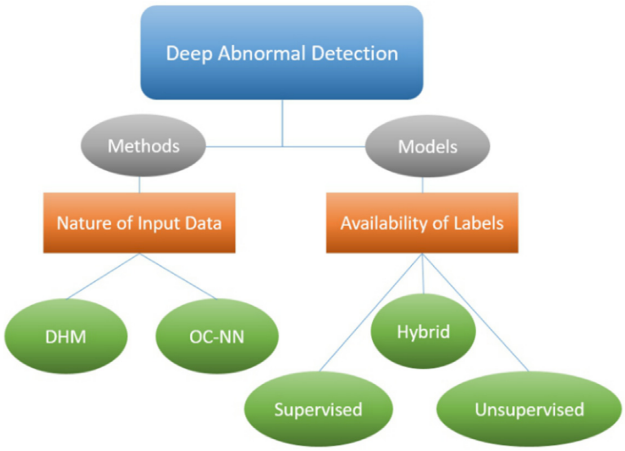

Figure 2: Various DAD methods and models.

behavior state of a subject, and velocity is applied as its actions. A fuzzy c-means algorithm was adopted by Cui *et al.* (2014) to cluster the source and sink points of trajectories that were deemed critical points into several groups. Then, the trajectory clusters can be acquired. The statistical feature histogram, which contained the motion information for each trajectory cluster, was built after being refined with Hausdorff distances. Eventually, the local motion coherence between test trajectories and refined trajectory clusters was used to judge whether they were abnormal. For the falling detection of the human body, utilizing an image pre-processing in which three triangular-mass-central points were included to extract the features, Juang *et al.* (Juang & Wu, 2015) presented a preliminary design to identify fall-down movements from body posture. The experimental result showed that the proposed method could properly extract the feature values, and the accuracy could reach up to 90% for a single body posture. Zhang *et al.* (2016) have used a wearable tri-axial accelerometer to capture the movement data of the human body and proposed a novel fall detection method based on a one-class support vector machine (SVM). The positive samples trained the one-class SVM model from the falls of young volunteers, and the outliers from the non-fall daily activities of the young volunteers and samples from the elderly volunteers formed a dummy set to contrast the positive samples.

In our proposed abnormal event detection, a detect–tag–judge framework to learn from a collection of falling-down samples has been developed. The steps of the sequential framework are scratching out the human from the scene by the object detection, assigning tags to scratched frames, and then checking the movement and measuring the distance among the tagged frames. The proposal of abnormal events is generated by giving the abnormal action tags and setting the raft values as the basis for distinguishing normal and abnormal. With the proposal, the abnormal events contained in the proposal are finally classified. Several deep learning models were considered to develop the application of the epidemic control topic. By schedule, the methods for abnormal analysis and social distance measurement will be explained in the following.

2.2 Particular deep models used for anomaly detection

Any deep learning framework and adopted system for anomaly detection need to be able to detect anomalies in a short time and maintain a satisfactory level of accuracy. One-stage method is to detect and identify the position of objects in one step. That is, a neural network can detect the position of objects at the same time and identify objects. One stage can save a lot of computing time. Single Shot multibox Detector (SSD; Liu *et al.*, 2016) and YOLO are the most commonly used one-stage methods. SSD, namely a target detection algorithm proposed by Liu (Liu *et al.*, 2016), is one of the main frameworks among the detection technologies. Compared with YOLO, SSD is advantageous in mean Average Precision. SSD based on the ImageNet dataset is architected and deployed by a backbone network VGG-16 model. The SSD structure uses convolution layers to replace the full connection layer, removes the dropout layer, and uses an expanded convolution layer to replace the last maximum pooling layer (Gu *et al.*, 2017). SSD adopts a pyramid structure, which uses feature maps of different sizes to perform soft-max classification and location regression on multiple feature maps simultaneously. Instead of using K-means to discover aspect ratios, SSD defines a set of aspect ratios to be used in the bounding box of each grid cell. SSD is very sensitive to the size of the bounding box, so it has poor performance in small object detection tasks (Gu *et al.*, 2017).

By contrast, the YOLO model employs only one CNN for judging the object category and identifying the object's position in the scene. Relative to SSD, it greatly improves the identification speed of YOLO. The former module used a modified GoogleNet as the backbone network (Bochkovskiy *et al.*, 2020). Thanks to the backbone, a model named darknet-19 was created, which followed the general design of a 3×3 filter to double the number of channels in each pooling step. A 1×1 filter was applied afterward in the whole network to compress features periodically. The model is first trained as an image classifier and then adjusted for the detection task.

Along with its development, there are four versions of YOLO. Since using higher resolution images at the end of pre-training can improve the detection performance, this principle was adopted in the second version of YOLO. The first version of YOLO was to predict all four values that describe a bounding box directly. The *X* and *Y* coordinates of each bounding box are defined relative to the upper left corner of each cell and are normalized according to the cell size so that these coordinates range from 0 to 1. The width and height of the box are defined so that the model predicts the square root width and height. By defining the width and height of the box in the form of the square root value, the difference between the large values is not as significant as the difference between the small values. In the third version, this is changed to a standard feature pyramid network. Alternating between the output of a prediction result and the upsampling feature map helps detect small targets in the image. The fourth version of YOLO introduces a bigger new model, CSPDarknet-53, with better performance (Sultana *et al.*, 2020).

The main difference between the two architectures of SSD and YOLO is that YOLO architecture uses two fully connected layers, while SSD network uses convolution layers of different sizes. YOLO model predicts the probability of a target when there is a target and then predicts the probability of each category. SSD model tries to directly predict the probability of a category in a given target box. In the early stage, SSD was better than YOLO in speed and accuracy, but after improving the third version of YOLO, the speed was significantly ahead. In particular, the fourth version of YOLO has surpassed SSD inaccuracy (Sultana *et al.*, 2020). To make the network operate fast and reduce the amount of computation, the backbone of the structure uses CSP-Darknet53, which is famous for its excellent speed and accuracy. The portion around the network neck has relatively expanded

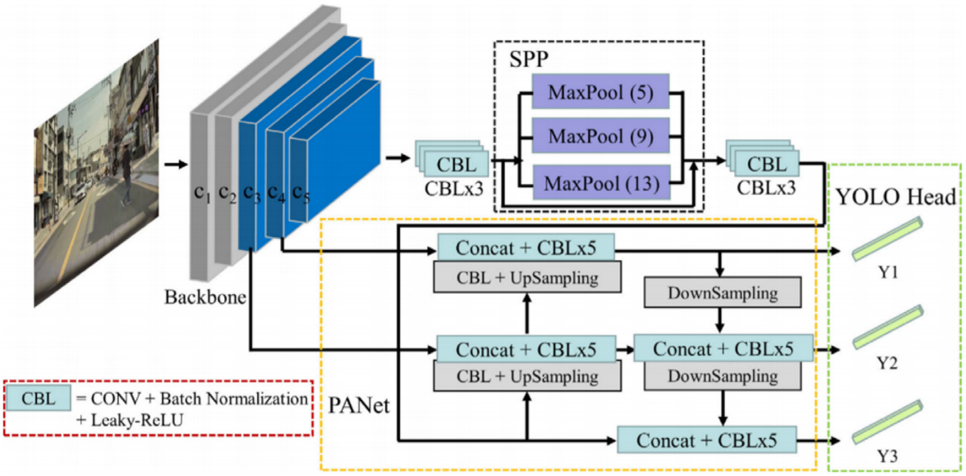

Figure 3: The overall structure of YOLOv4, including a CSPDarknet backbone, a neck including SPPNet and PANet, and a YOLOv3 head (Bochkovskiy *et al.*, 2020).

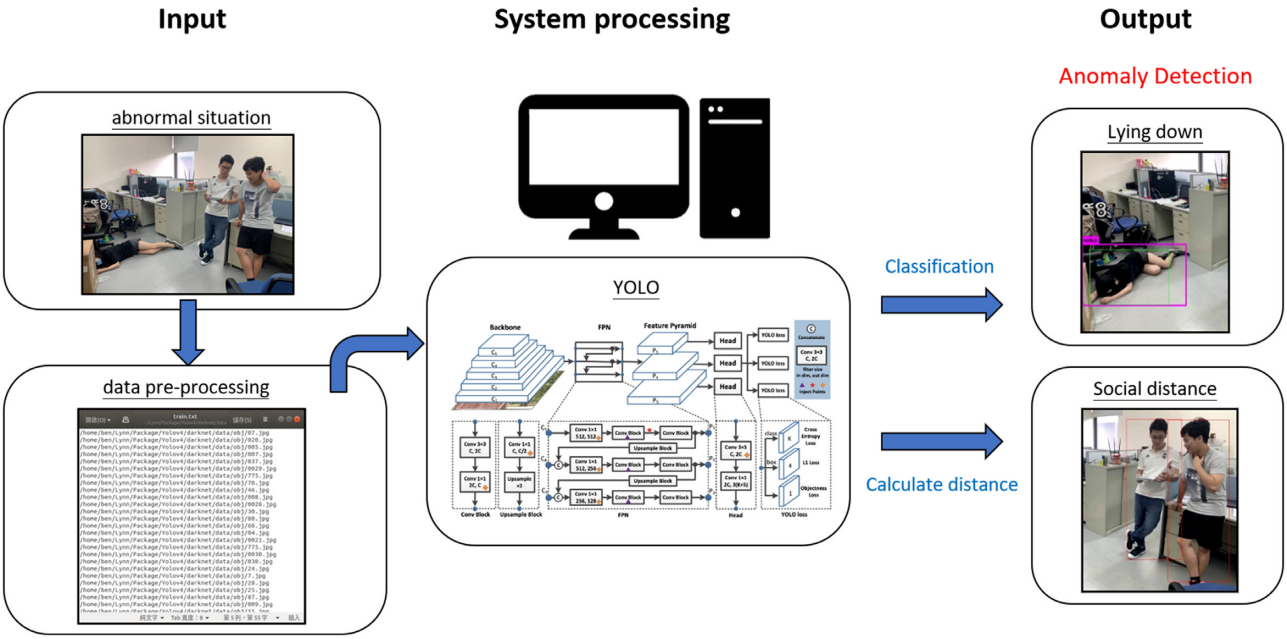

Figure 4: The overall structure of the proposed system.

the emersion field of the detected regions and get a better fuse to different scale features by using PANet (Liu *et al.*, 2018) and SPPNet (He *et al.*, 2015). The last head portion uses the conventional YOLOv3 without changes. The evolutionally constructed architecture guaranteed YOLOv4 the fastest and most accurate detection currently. Hence, in this paper, YOLOv4 is employed, whose overall structure is shown in Fig. 3.

3. Methodology

In this section, various kinds of detection of anomalies that may occur during the epidemic are described in detail. As the ultimate goal is to break the transmission chain and relax the need for personnel patrol, two anomalies are included, as shown in the architecture in Fig. 4. The first anomaly is related to human actions, and the other one is the distances among people. The process of each part is individually divided into three steps. In the first part, the fall detection is considered as an anomaly detection problem. YOLO is introduced to tackle the problem of having a panoramic monitor of the human's motion and continuously identify the anomaly in an area. Therefore, the methodological steps, including data collection, data labeling, and parameter tuning, will be followed up in this section. After the anomaly detection, the theme turns to measure the distances among people. In this second part of abnormal detection, a simple 2D measurement instead of a three-dimensional one is used to respond more quickly when people get crowded in the area. The steps to accomplish the task are also explained, including human identification, actual distance calculation, and threshold to activate the warning.

3.1. Accidental fall as an anomaly

Coronavirus mainly attacks the lungs, so that it may cause sudden shock in infected people. In addition, patients with a certain degree of discomfort may also collapse and feel headaches and even dizziness (Pal & Sankarasubbu, 2021). To make the model accurately detect abnormal motion, many

Table 1: Abnormal behavior dataset.

Class name	Description	No. of images
Notice	Human fall dataset from Shutterstock (Shutterstock 2021)	500
	Self-collected data of human falls	153
	Self-collected data of human fall movements (50% body cover)	25
	Self-collected data of human fall movements (30% body cover)	22

Table 2: Label feature of the dataset with category ''notice.''

Class name	Description
Notice	People lying down in a space People fall onto the ground

abnormal motion data must be prepared first. Such as falling on the ground on a hot day or lying on the ground in public. Although COCO data (Lin *et al.*, 2014) and ImageNet (Deng *et al.*, 2009) are the first choices for most people when training models, there are few images of human motion. Finally, over 500 pictures of abnormal behaviors on Shutterstock (Shutterstock 2021) have been searched to formulate a training set, a total of about 700 photos. Two hundred fall data were further subdivided into 50%, 30%, and 0% of body obstructed, a summary can be found in Table 1. To improve the accuracy, the image size was adjusted to $1024 * 1024$.

After normalizing the image size of the dataset, the image data are converted into text files. In this procedure, the Label image tool (GitHub, 2021b) is used to mark these anomalies in the image manually. In this research, the anomaly is defined as people lying down or falling onto the ground in the scene. With this labeling tool, the set of training images is individually labelled by scratching out the frames of anomaly. The manual procedure is regarded as "notice" displayed on the screen when the frame scratch has been done, shown in Table 2. The images are then stored in a folder that the training model can trace during the training procedure.

A set of feature data is converted and saved into a separated text file with the labeling task. The feature data include five values. The first value is the category, while our category 0 represents the abnormal action. The second and third values are x and y, respectively, representing the ratio of the central coordinates of the bounding box to the width and height of the picture. In contrast, the fourth and fifth values w and h, respectively, represent the ratio of the bounding box's width and height to the input image's width and height. The latter four values are the normalized coordinates of the bounding box, as a sample illustrated in Table 3.

To set up the YOLO model for training purposes, it is necessary to set the parameters. This research aims to detect abnormal human movements, which is a binary classification problem. The final applicable model is often optimized by adjusting the parameters. Many adjustable parameters exist in the adopted YOLO model where Table 4 presents the usage of the parameters with regards to the best module performance.

The parameter "Batch" represents how many samples the network has accumulated before performing a forward propagation, which can be set to 64, 32, 16, 8, 4, 2, and 1. In general, the larger the Batch value is, the better the training effect will reach. As the training of the deep model is performed on GPU, the Batch size should be too large to balance the computation cost. The parameter "Subdivisions" is used to divide a batch of samples into subtimes to complete the forward transmission of the network. The larger the Subdivisions, the smaller the memory allocated for training. To achieve a balance between the training efficiency and the burden of memory usage, both the parameters Batch and Subdivisions are properly settled to the identical value of 64 after plenty of attempts. In other words, in each iteration, 64 samples will be randomly selected from the training set to participate in the training. These batch samples will be divided into 64 individual subdivisions, i.e. one sample in one subdivision, 64 times and sent to the network for training to reduce the pressure of memory occupation. "Max_ batches" is a stop criterion parameter. That is, when the batch iterations reach 4000, the learning process will be stopped. One formula to calculate Max_batches is the number of classes $*$ 2000, but the base cannot be less than 4000 (Bochkovskiy *et al.*, 2020). In the research, Max_batches is set to 4000. The function of Steps is to adjust the learning rate according to the Batch parameter. Its formula and parameter values are set to be 80% and 90% of the Batch parameter, i.e. 3200 and 3600 for formula and parameter values, respectively. The input image size is $416 * 416$. Finally, CSPDarknet-53 is used to specify the training set path and testing set path.

3.2. Social distance detection

One of the best ways to avoid the virus is to wear a mask, which can protect the person wearing a mask and others. However, some occasions only need to maintain social distance to avoid wearing masks, such as outdoor environments like parks, playgrounds, or areas with fewer people. Some special circumstances require fewer restrictions, such as while exercising or eating in a restaurant. Therefore, it is necessary to keep social distance as much as possible to protect the transmission of viruses. As for the definition of social distance, each country has different norms. According to Taiwan, the social distance is 1.5 meters, which is approximately equivalent to the distance between two hands of one person. To sum up, if someone does not wear a mask, someone should keep social distance to monitor their health.

First, the tag data in YOLO have been used to check and detect anyone in the picture. Through the identification data that YOLO has trained, the image and track the position of these people

Table 3: Each data after labeling image.

Category number	Object center in x	Object center in y	Weight (w)	Height (h)
0	0.527644	0.804087	0.545673	0.233173

Table 4: Data after labeling image.

Class	Batch	Subdivisions	Max_batches	Steps	Height	Weight
1	64	64	4000	3200, 3600	416	416

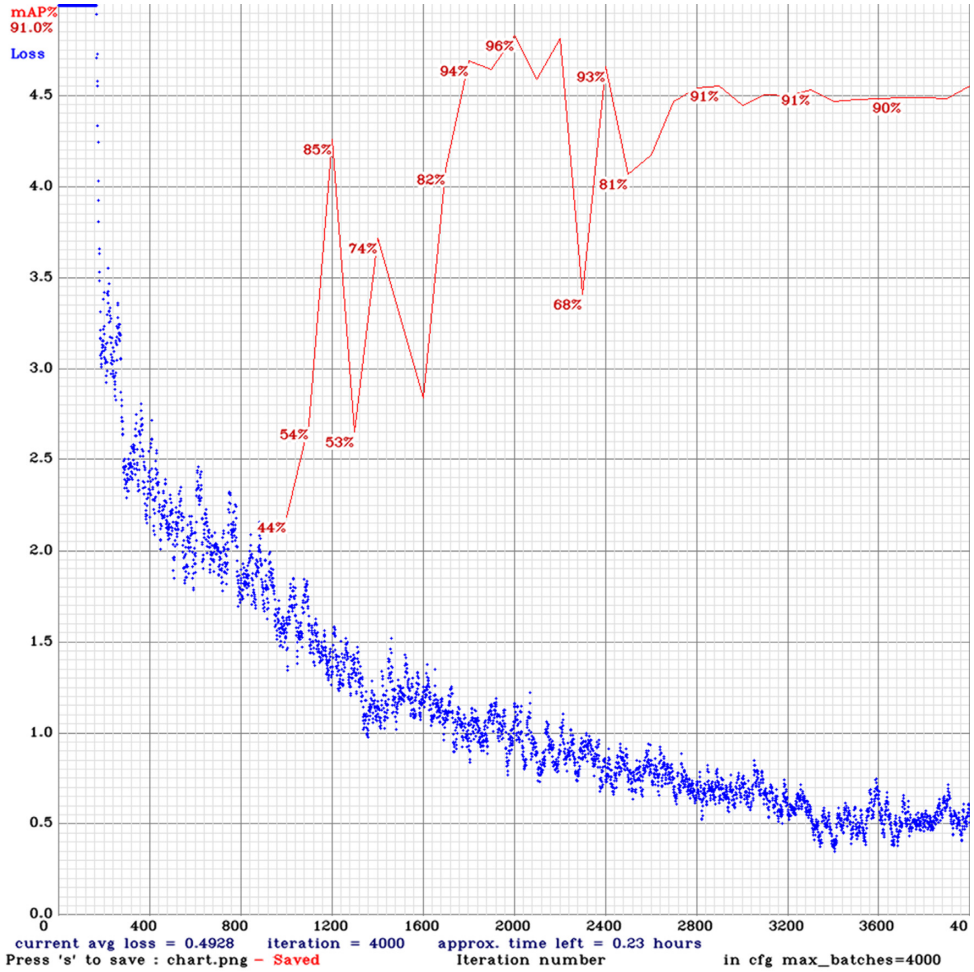

Figure 5: Training result.

Figure 6: Detection with/without anomaly in the venue.

can be directly detected. If a person exists in the image, the system draws a box around them and gets the coordinates of these bounding boxes. Then, the centroid position of the bounding box is collected. Of course, at least two people must be detected before calculating the distance. After obtaining the centroid position, the Euclidean formula is used to calculate the linear distance between people (Natanael *et al.*, 2018). The pixel value is converted into the distance from the point to the nearest background point, and whether the distance between two people is less than a threshold value (*N* pixels) is detected (Natanael *et al.*, 2018). The safe distance is detected in that case. Otherwise, it is not. This distance measurement method will be further promoted onto a depth camera model to implement into a 3D scenario by collecting the depth information of a given pixel.

In the formula, d is the distance, x_1 is the centroid *x_value* of the first box, y_1 is the centroid *y_value* of the first box, x_2 is the

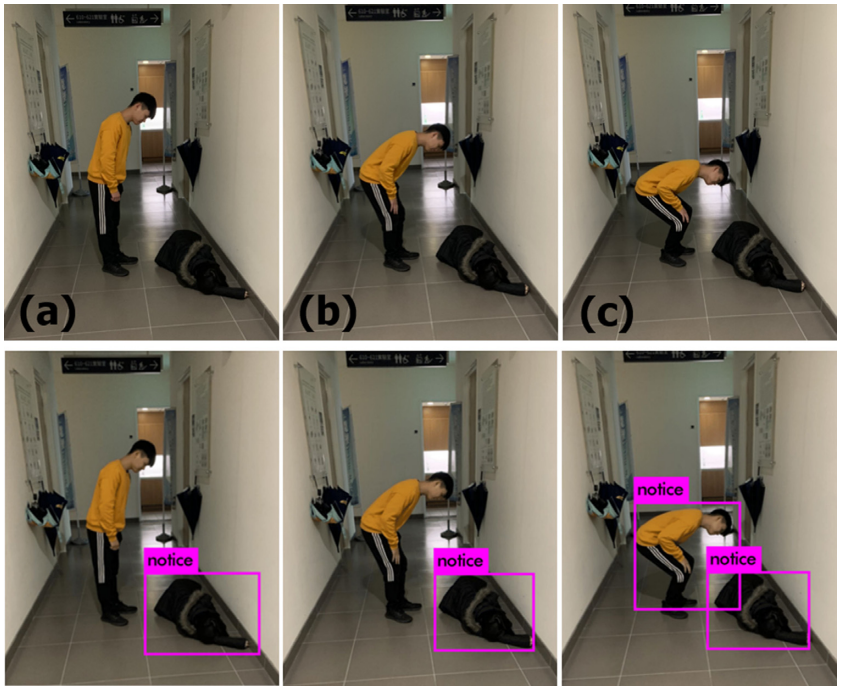

Figure 7: Anomaly detection with different degrees of body bending. (a) 15, (b) 45, and (c) 90 degrees.

Figure 8: Anomaly detection with squatting posture.

centroid *x_value* of the second box, and x_2 is the centroid *y_value* of the second box. To calculate the distance between the boxes, how many pixels are on how many bounding boxes are to be "noticed" and then the counting numbers will be calibrated using a scaler with known dimensions. Here, it is worth noticing that no matter how to zoom in/out an image, the pixel values of the original image won't change. Meanwhile, the scale factor won't change either. Hence, the invariant scaling capability can be satisfied.

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

Generally speaking, the social distance to be kept during the pandemic is that the distance between people must be 1 meter indoors and 1.5 meters outdoors. Since the research is focused on developing an online monitoring system, a plane fixed-angle video shooting was adopted, rather than a three-dimensional depth camera, for saving calculation time and providing the measurements. In reality, the distance among people is calculated with the center of two feet side by side as the reference point and the straight-line distance. The actual distance of three-dimensional space is converted by plane adjustment. The social distance threshold is set to 1.5 and check whether it meets the conditions. Finally, the bounding box is set to two output colors. While red means people who are in danger, green means those protected. The detailed experimental results will be presented in the following section.

4. Results and Discussions

In this section, series of experimental studies have been conducted to validate the proposed method in real time and also to discuss in depth the behavior of the proposed algorithm compared with various case studies.

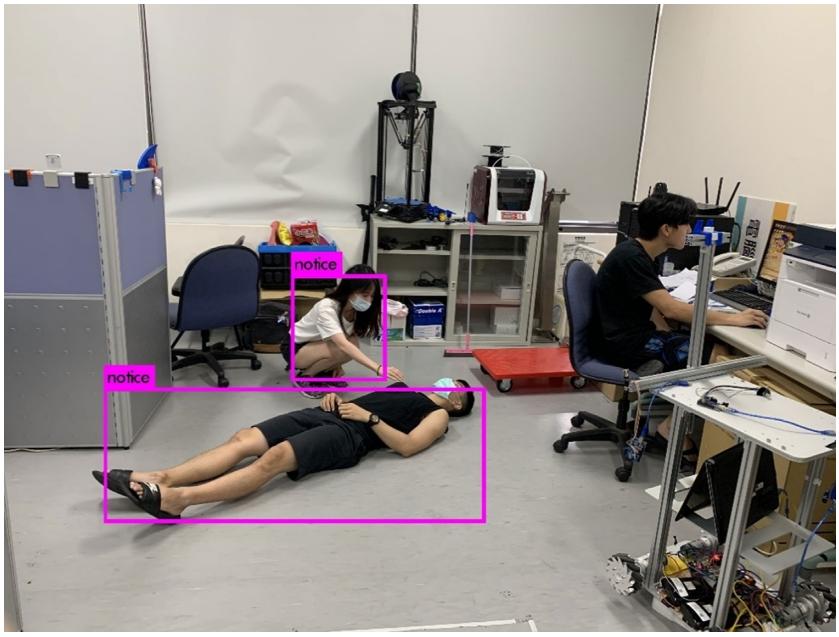

Figure 9: Anomaly detection with lying down and squatting posture.

4.1. Model training convergency

Figure 5 shows the training result of abnormal behavior. The blue line is the loss function, and the loss function is the "residual between the actual value and the predicted value." The minimization of the loss function is an important key to learn a model. Generally, the smaller the loss function is, the more the fitness of the model increases. The red line is the validation accuracy of judging abnormalities based on the arrival loss. As expected, the validation accuracy would increase in the underfitted region and then come to a relatively saturated and stable region to be mature. The overfitting problem would have occurred if the model had been overtrained. The principle to choose the model is, in general, based on the expectation on the two curves. At the beginning of training, while the blue loss curve continuously decreases, the observation of the red accuracy curve continuously increases. A typical underfitting happened to the stage. The red curve fluctuated and was unstable during the 1200 to 2400, succussing to the underfitted region. The fluctuation represented a kind of transition before saturation. In this region, the blue curve kept its trend in the decreasing down fashion. Because of the instability, the accuracy is still unacceptable. With the iterations passed, the loss function slowed down its decreasing rate and dropped to the position close to 0.5, and the corresponding accuracy got stable and saturated. The iterations of 2800 were chosen for the final model with an accuracy of 91%.

4.2. Static model validation

In reality, there are many types of behaviors of people be regarded as abnormal. In this research, the abnormal behavior of "falling"' is focused particularly. The discussion in this section is thus focused on the general outcomes of the abnormality detection with different conditions, e.g. the conditions of multiple subjects in the same scene, the body subject to different scales of bending, and the body subject to various occlusions. To verify whether the system correctly identifies the difference between normal people and abnormal conditions, a scene was specially arranged where a person sat together and lay down on the ground, as shown in Fig. 6. As detected by the proposed model, it can be seen that the obvious abnormal behavior has been highlighted with a pink frame on the left-hand side of the photo, and the person without any abnormality on the right-hand side is free from the highlighted frame, i.e. has not been detected. With the evidence, the system function can be roughly assured for the abnormality detection.

Differentiation from varieties of body bending: because the system can accurately distinguish between normal and abnormal people. Then, an experiment has been designed, as shown in Fig. 7, to know how bent a person's body can be detected as abnormal. The scene is in the corridor of a public place, with a person lying on the floor on the right. For people on the left-hand side, abnormalities according to the degree of body bending are detected.

The results show that if a person's movement is only slightly bent, it will not be detected, but it will be detected as abnormal when it is bent to nearly 90 degrees. Because normal people do not walk like this, if a person bends too much while standing, it may be a sign of heatstroke or dizziness, which is abnormal.

For a normal person sitting in a chair, the system was expected to decline the anomaly detection correctly. To show the system's capability, a case that a person squatted on the ground, who will be judged as having a problem, is also correctly detected with the highlighted frame (Fig. 8). Finally, Fig. 9 shows the correct detection of chair sitting and bending, and lying down. According to the above results, when a person bends nearly 90 degrees or even more than 90 degrees, squatting or lying down, the system will detect the person as abnormal. Therefore, the cases of sitting on a chair can be differentiated accordingly.

Detecting with various occlusions: In this anomaly category, the person might be covered by obstacles such as chairs, so people nearby cannot detect it in time. Therefore, the abnormal movement is detected according to different degrees the object (body) has been occluded to ensure the effectiveness of the developed system. Four types of occluded anomaly, including 0% (exposing the whole body), 30% (exposing the most part below chest),

Figure 10: Anomaly detection with various occlusions. (a) 70%, (b) 50%, (c) 30%, and (d) 0%.

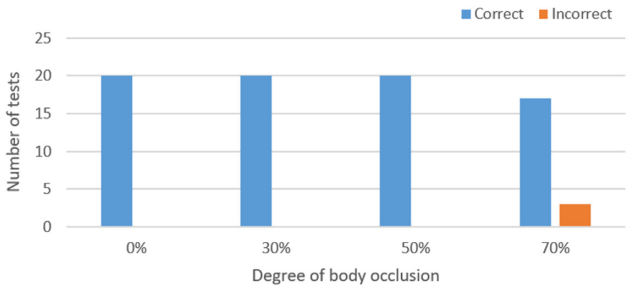

Figure 11: Actual accuracies of different occlusion rates.

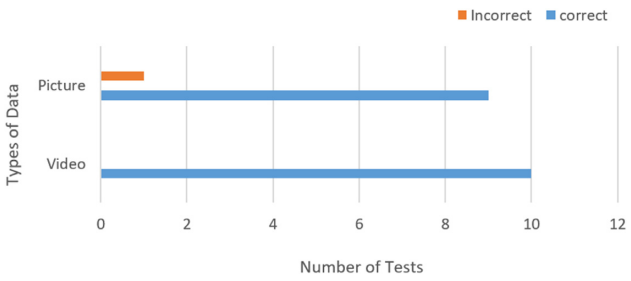

Figure 12: Difference between static and dynamic recognitions.

50% (exposing the half body), and 70% (exposing only the legs), were thus categorized for detection. With the cases of occluded detection, specification limitation of the developed system can be known to avoid improper applications. Intuitively, the developed system should know that the fallen object is a human and detect the condition of the falling degree at least accurately. One demonstrative example of the occlusion experiment is illustrated in Fig. 10. The fortunate example shows all the falling downs under the various occlusions are successfully detected.

To investigate the accuracy of the occlusions and the case experiment above, a large-scale experiment with many repetitions for accuracy validation has been arranged. The content of the experiment is: a female tester simulates 20 times according to different degrees of body occlusion (0%, 30%, 50%, and 70%, respectively), as shown in Fig. 11. In the case of 0%, 30%, and 50% occlusions, the system can clearly and correctly detect the situation, while in the case of 70% or more body occlusion, there will be three times when the situation is not detected.

4.3. The recognition in dynamic

In the static part of anomaly recognition, only the result of people lying down can be detected. Static recognition cannot identify the transition from normal to abnormal. Hence, investigating the transition is important if people seek anomaly detection and form a dynamic recognition system. Here, dynamic

Figure 13: Fall detection of different subjects in the same scene for cases a, b, and c.

Table 5: Accuracy of the fall test process for different people.

Fall down	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Case 1	48.66	59.16	71	78.16	80.41	81.16	83.83	83.58	86.83	91.91
Case 2	56.33	73.33	69.08	71.25	74.66	79.83	84.25	86.08	89.66	92.66
Case 3	48.83	74.08	74.58	74.91	79.08	81.33	81.91	84.91	85.91	89.91

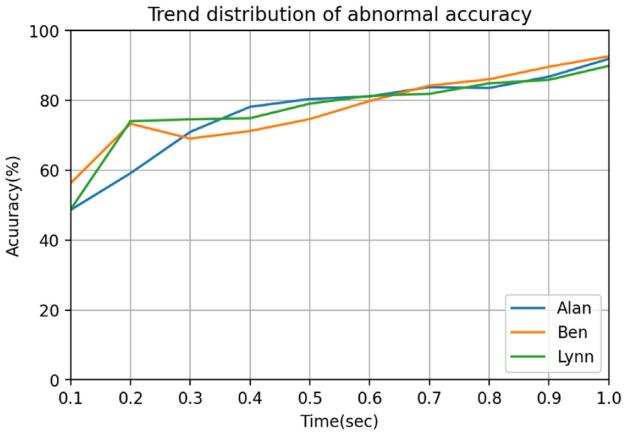

Figure 14: Trend distribution of abnormal accuracy.

recognition can be used to prevent early exceptions and be a good way to understand the cause and development of the exceptions in later the forensic check and explanation.

As a result, an example was used to grasp dynamic recognition better. Ten movies have been added with more than 70% of the repetitions of falling down with the body occluded to the 10 static pictures used in the previous static model evaluation. As a result, the experiment was constructed with 10 dynamic streaming pictures to explain the falling down and 10 static photos to display the falling down directly to compare dynamic recognition and static recognition. The results are therefore shown in Fig. 12. It can be seen that the dynamic repetitive videos were correctly recognized without loss, while those static images were recognized with three (15%) misclassifications. The main reason for the unusual results is that if detecting through a static image when the body is impeded 70% or more, the developed system would not directly ensure whether the target is a particular man or just an ordinary object. So, in this case, three images are not correctly identified. However, if the recognition is dynamic, the transitional images would give particularly reference information in the successive recognition, so, even 70% or more occlusion, the system can still detect successfully.

Recognition dynamics with anomaly development – an experiment: Three different volunteers are tested 15 times in the scenes—taking "fall from chair" as abnormal behavior and recording the confidence scores at the corresponding subdivision tags for statistical analysis (shown in Fig. 13). Table 5 calculates the average confidence scores at each subdivision tag of the 15 repetitions of the 3 volunteers. The values in Table 5 are averaged from the 15 repetitions conducted by 3 different volunteers in the same scene. It can be observed that the values are rather reliable.

Figure 14 shows the trend distribution based on the values in Table 5. It can be seen that the change of confidence score is fluctuating, or even fluctuating, in about 0.1 to 0.3 to 0.4 seconds

Figure 15: Fall process chart of body occlusions. (a) 50% and (b) 70% occlusions.

Table 6: Accuracy of test 50% and 70% body occlusions.

Body occlusions	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
50%	37.16	43.25	56.16	63.66	70.5	77.58	83.33	91.25	92.83	96.33
70%	44.66	44.91	53.75	53.66	66.66	77.08	78.58	79.41	79.75	87.91

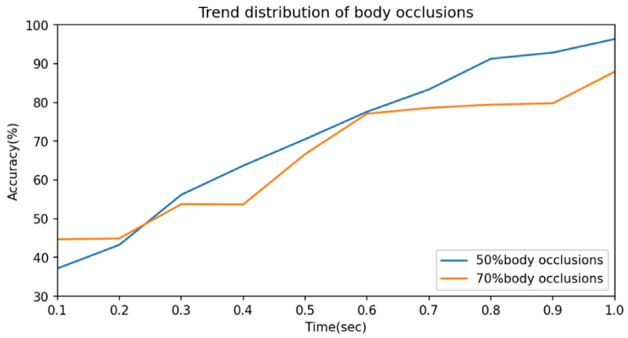

Figure 16: Trend distribution of body occlusions.

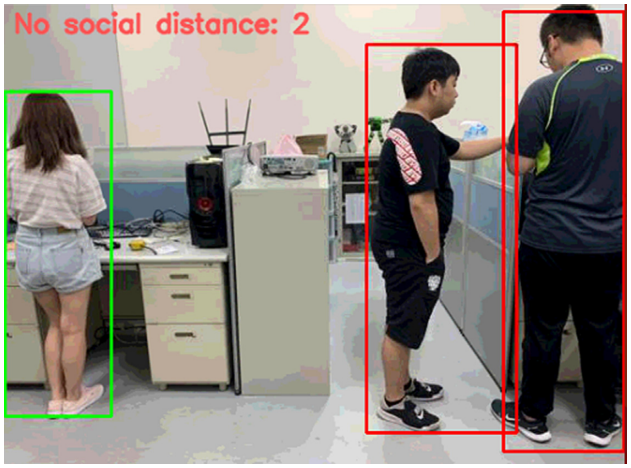

Figure 17: Social distancing experiment (two people are not following social distancing protocol).

of the fall. It did not grow steadily until about 0.5 seconds later and finally stopped at about 90%.

Recognition dynamics with occlusion: then, looking backward to check the trend with the cases of body occlusion. The test recruited more data of 70% body occlusion and 50% body occlusion in 10 repetitions. The same experimental procedure in the previous section recorded the confidence scores at each subdivision tag of fall framing. People want to understand further whether the degree of body occlusion will affect the system's confidence score in detection when the system can identify the anomaly conditions. Sample resultant successive identifications are tiled as the panels of Fig. 15.

The confidence scores were then averaged and listed in Table 6. The average values were averaged from the 10 repetitions of every volunteer in each subdivision tag. It can be seen that the values of 70% body occlusion before subdivision tag 0.3 are higher than those of 50% body occlusion, and then reversed afterward the subdivision tag 0.3. The data in Table 6 are then converted into a graph of Fig. 16. In Fig. 16, it is obvious that the curve is still unstable in the first 0.3 seconds of detection. After 0.3, 70% body occlusion detection effect is lower than that of 50% body occlusion. However, the final results of both are very high. However, the detection effect of 50% body masking is better than that of 70% body occlusion.

4.4. Social distance

In the section of the experimental results of social distance, the authors verify it in two different places: one is the experimental verification in the laboratory and the other is the actual experiment in public places. The former participants are all students in the laboratory, while the latter are all college students or teaching staff in the building. Finally, the detection data of the two will be discussed and the results will be analysed. In the laboratory validation, a total of three people participated in the experiment. An image of such an experiment is shown in Fig. 17, a 445 × 330 cm image. The distance between the center of the green box on the left and the red box in the middle is 250 pixels. It is equivalent to the actual measurement distance of 150 cm. So, the value of 250 has been set as the distance threshold parameter. When the distance between the center points of the two boxes is greater than 250, it means that the social distance specification is met.

On the contrary, if the distance is less than 250, the social distance is not observed. It can directly see when the two people on the right are too close. Hence, a red box is displayed. On the left, a green box is displayed while keeping it safe. Finally, a text indicating some people without social distance is presented on the upper left side of the image.

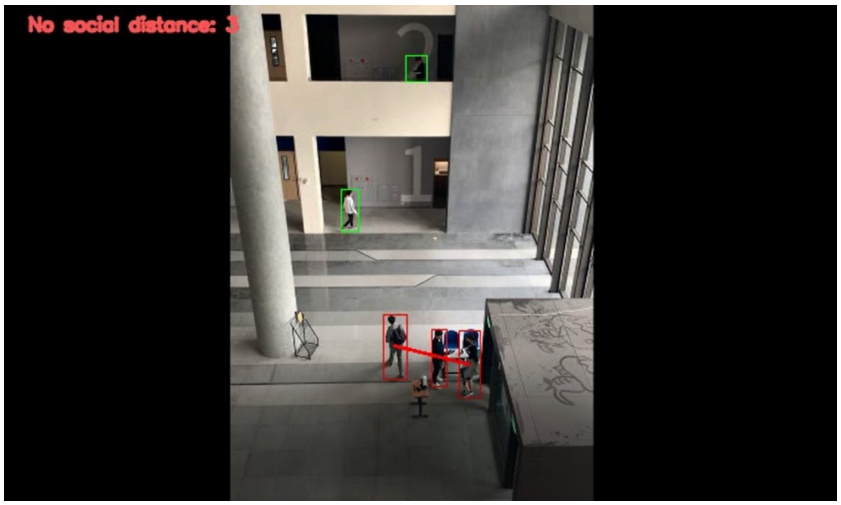

Figure 18: Detection at the entrance of the building.

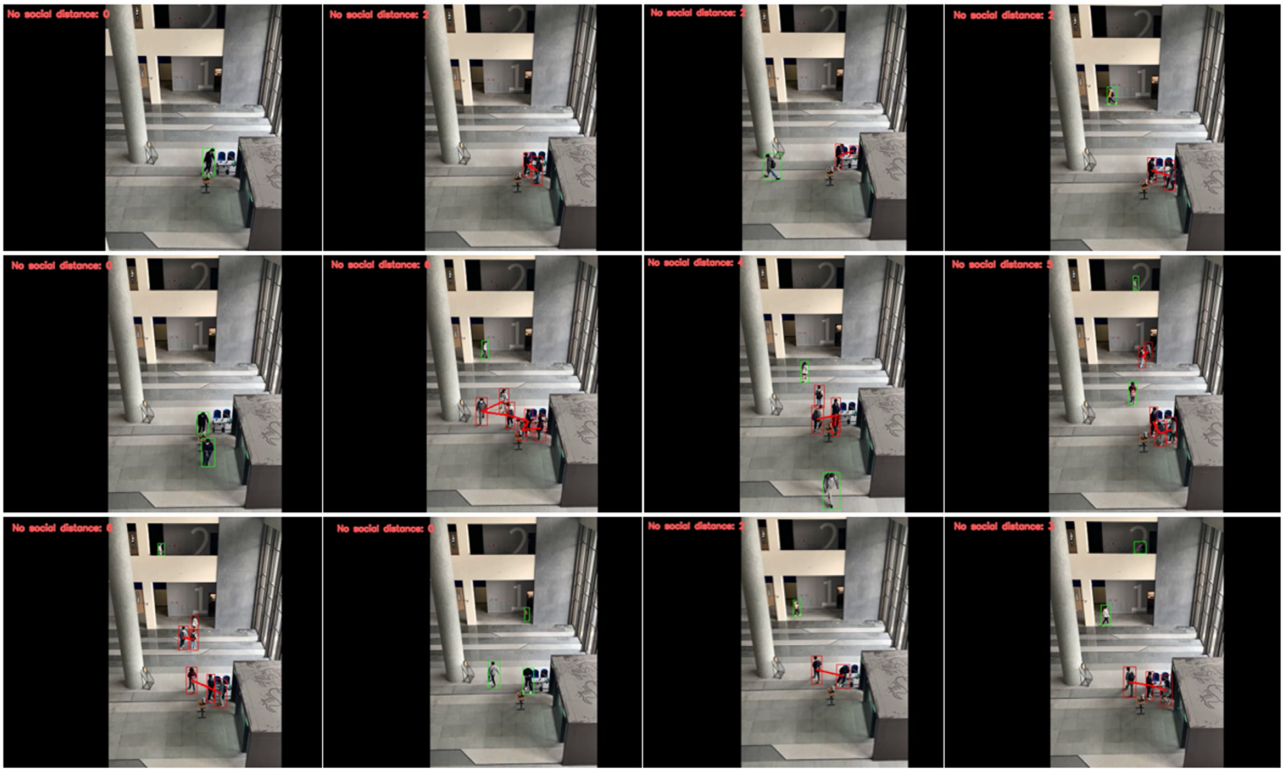

Figure 19: Social distance monitoring system.

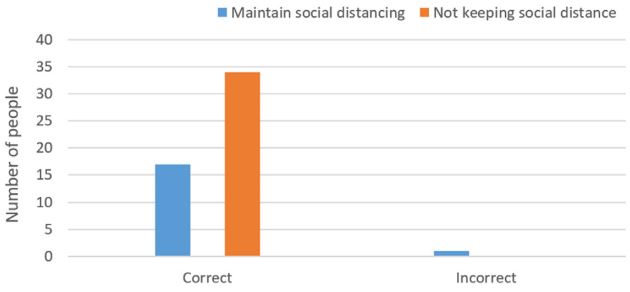

Figure 20: Test results of 52 students.

The experiment in a public place took place at the entrance of the school building. Because in the case of a severe epidemic situation, the main entrance of the building is the only place for students to go in and out. Through the detection at the door, the flow of people effectively can be controlled to check whether every student keeps a safe social distance. Figure 18 is the top view taken from the third floor downward. It can be seen that students are passing by on the second floor and keeping social distance. Three students are coming in at the gate of the first floor simultaneously; the system detects that they do not keep social distance, while the students above the first floor keep social distance. This experiment verifies that the developed system has been successfully functional. Due to the long distance between the camera and the objects, an error might happen. If people are too close to stationary objects (chairs and tables), these objects will be detected as people.

To verify the measured results, 52 students were counted in the 3-minute video (Fig. 19). Among them, 17 students who entered the door alone kept a social distance, and 34 students who walked together did not keep social distance. One of the students did not maintain social distance, but because there were chairs in the space, he was mistakenly judged not to maintain social distance.

The verification results are recorded in Fig. 20. On the left, both with and without maintaining social distance are correctly detected. The incorrect cases are presented on the right side. In this part, because the camera is not a depth camera, it will not distinguish the distance in space when the angle of view and direction is not fixed. If a person is too close to the camera, the judgment of distance will be wrong. In this paper, because the camera is not a depth camera, it will not distinguish the distance in space when the angle of view and direction is not fixed. If a person is too close to the camera, the judgment of distance will be wrong.

4.5. System integration results

Finally, the two functions were combined as shown in Fig. 21. People can see that the two people on the right do not keep a social distance to be detected and marked with red boxes. Although the person on the left of the picture keeps social distance, it is detected as abnormal because it falls to the ground. In the upper left corner of the figure, the number of people who have not kept social distance will be calculated and displayed.

5. Conclusion and Future Works

In this paper, human abnormalities and phenomena during a viral pandemic have been explored by regarding human falls as abnormal behaviors and monitoring the social distance between people. Deep learning visual system has been used to solve social and security problems. In the part of abnormal behavior,

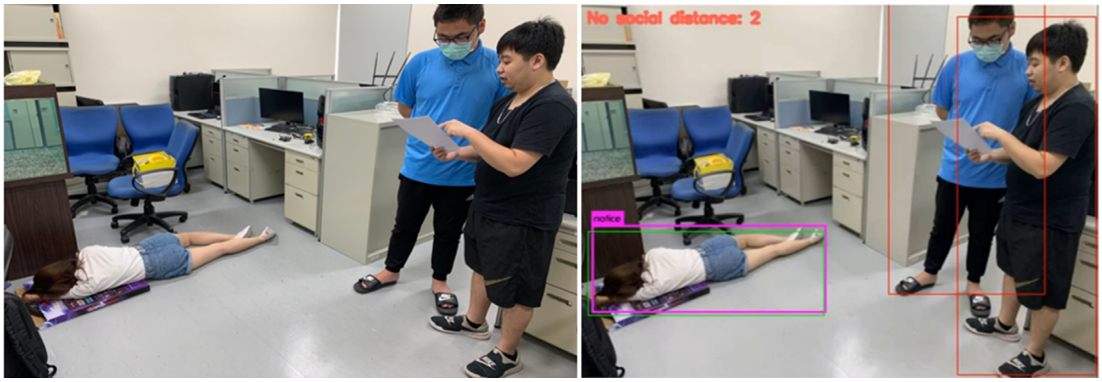

Figure 21: Result of the integrated final system in progress.

different situations have been investigated, such as the influence of the shelter in space on the recognition degree of detection and the influence of the bending degree of the human body on the accuracy of recognition. A series of experiments were conducted for the verification, and the results were recorded and analysed. The experimental result indicated that the accuracy of the trained model is 91%, and the multiple-image identification better identified the falling process. The 2D spatial distance calculation method was used to determine whether the actual three-dimensional distance maintains social norms in the social distance section. Then, the 2D linear distance in the screen is set as a threshold value to determine whether there is social distance. The system can detect the abnormal behavior of people in public places and monitor the social distance between people and reduce the risk of infection. The GitHub repository to all the files for this paper can be found in GitHub (2021a).

In the future, it is expected that the developed vision system be utilized in various systems such as robots or other monitoring systems. SLAM and machine vision-based sensing techniques will be used for path planning and obstacle avoidance. For different target areas, exploration paths will be formulated to guide the robot to perform global scanning. The robot will know its current position and determine the next destination based on position and exploration plan; for a discovered anomaly, the vision system is built into the microcontroller and powered by AI libraries, such as OpenCV, Keras, and TensorFlow, that will determine its coordinates corresponding to the robot's position. The robot will also identify obstacles during moving by collecting information from the sensors to execute avoidance routines. A depth or stereo camera model will be required to replace the 2D space distance calculation method to accurately capture each person's position in space and detect the distance between each other. The robot could also be used for various applications such as telehealth, screening, diagnosis, and disinfection.

Conflict of interest statement

None declared.

References

Andrews, J. T., Morton, E. J., & Griffin, L. D. (2016). Detecting anomalous data using auto-encoders. *International Journal of Machine Learning and Computing*, 6(1), 21.

Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. preprint arXiv:2004.10934.

Bragazzi, N. L., Dai, H., Damiani, G., Behzadifar, M., Martini, M., & Wu, J. (2020). How big data and artificial intelligence can help better manage the COVID-19 pandemic. *International Journal of Environmental Research and Public Health*, *17*(9), 3176.

Chen, J., Glover, M., Li, C., & Yang, C. (2016). Development of a user experience enhanced teleoperation approach. In *2016 International Conference on Advanced Robotics and Mechatronics (ICARM)*(pp. 171–177). IEEE.

Chung, C. L., Chen, D.-B., & Samani, H. (2020). Action detection and anomaly analysis visual system using deep learning for robots in pandemic situation. In *2020 International Automatic Control Conference (CACS)*(pp. 1–6). https://doi.org/10.1109/CACS50047.2020.9289819.

Cui, X., Liu, Q., Gao, M., & Metaxas, D. N. (2011). Abnormal detection using interaction energy potentials. In *CVPR 2011*(pp. 3161–3167). https://doi.org/10.1109/CVPR.2011.5995558.

Cui, J., Liu, W., & Xing, W. (2014). Crowd behaviors analysis and abnormal detection based on surveillance data. *Journal of Visual Languages and Computing*, *25*(6), 628–636.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*(pp. 248–255).

Dong, E., Du, H., & Gardner, L. (2020). An interactive web-based dashboard to track COVID-19 in real time. In *The Lancet Infectious Diseases*(pp. 533–534.S).

Ferretti, L., Wymant, C., Kendall, M., Zhao, L., Nurtay, A., Abeler-Dörner, L., & Fraser, C. (2020). Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing. *Science*, *368*(6491), eabb6936.

GitHub. (2021a). permanent2001/YOLOv4_project [online] Available at: https://github.com/permanent2001/YOLOv4_project. [Accessed 15 December 2021].

GitHub. (2021b). tzutalin/labelImg. [online] Available at: https://github.com/tzutalin/labelImg. [Accessed 7 July 2021].

Gu, S., Ding, Lu, Yang, Y., & Chen, X. (2017). A new deep learning method based on AlexNet model and SSD model for tennis ball recognition. In *2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA)*(pp. 159–164). IEEE.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(9), 1904–1916.

Hsu, H. J., & Chen, K. T. (2017). DroneFace: An open dataset for drone research. In *Proceedings of the 8th ACM on Multimedia Systems Conference*(pp. 187–192).

Juang, L. H., & Wu, M. N. (2015). Fall down detection under smart home system. *Journal of Medical Systems*, 39(10), 1–12.

Kiran, B. R., Thomas, D. M., & Parakkal, R. (2018). An overview of deep learning-based methods for unsupervised and semi-supervised anomaly detection in videos. preprint arXiv:1801.03149.

Krizhevsky, A., & Hinton, G. (2009). *Technical report TR-2009, Learning multiple layers of features from tiny images*. University of Toronto.

Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., & Kim, K. J. (2019). A survey of deep learning-based network anomaly detection. *Cluster Computing*, 22(1), 949–961.

Li, C., Yang, C., Wan, J., Annamalai, A., & Cangelosi, A. (2017). Neural learning and Kalman filtering enhanced teaching by demonstration for a Baxter robot. In *2017 23rd International Conference on Automation and Computing (ICAC)*(pp. 1–6). IEEE.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*(pp. 740–755). Springer.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In *European Conference on Computer Vision*(pp. 21–37).

Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. preprint arXiv:1803.01534.

Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys and Tutorials*, 20(4), 2923–2960.

Natanael, G., Zet, C., & Foşalău, C. (2018). Estimating the distance to an object based on image processing. In *2018 International Conference and Exposition on Electrical and Power Engineering (EPE)*(pp. 0211–0216). IEEE.

Pal, A., & Sankarasubbu, M. (2021). Pay attention to the cough: Early diagnosis of COVID-19 using interpretable symptoms embeddings with cough sound signal processing. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*(pp. 620–628).

Samani, H., & Zhu, R. (2016). Robotic automated external defibrillator ambulance for emergency medical service in smart cities. *IEEE Access Journal*, 4, 268–283.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., & Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*(pp. 146–157).

"Stock Images - Photos, vectors and illustrations for creative projects | Shutterstock," Shutterstock. (2021). [Online]. Available: https://www.shutterstock.com/. [Accessed: 7 July 2021].

Sultana, F., Sufian, A., & Dutta, P. (2020). A review of object detection models based on convolutional neural network. In *Intelligent Computing: Image Processing Based Applications* (pp. 1–16).

Xie, Z., Jiang, P., & Zhang, S. (2017). Fusion of LBP and HOG using multiple kernel learning for infrared face recognition. In *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*(pp. 81–84).

Zhang, T., Wang, J., Xu, L., & Liu, P. (2016). Fall detection by wearable sensor and one-class SVM algorithm. In *Intelligent Computing in Signal Processing and Pattern Recognition*(pp. 858–863).