IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received May 17, 2021, accepted June 9, 2021, date of publication July 5, 2021, date of current version July 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3094720

Hot-Spot Zone Detection to Tackle Covid19 Spread by Fusing the Traditional Machine Learning and Deep Learning Approaches of Computer Vision

MUHAMMAD ZEESHAN KHAN[1], MUHAMMAD USMAN GHANI KHAN[1], TANZILA SABA[2], (Senior Member, IEEE), IMRAN RAZZAK[3], (Member, IEEE), AMJAD REHMAN[2], (Senior Member, IEEE), AND SAEED ALI BAHAJ[4]

[1]Intelligent Criminology Research Lab, National Center of Artificial Intelligence (NCAI), Al Khawarizmi Institute of Computer Science (KICS), University of Engineering and Technology Lahore, Lahore 54000, Pakistan
[2]Artificial Intelligence & Data Analytics Lab (AIDA), CCIS, Prince Sultan University, Riyadh 11586, Saudi Arabia
[3]School of Information Technology, Deakin University, Geelong, VIC 3217, Australia
[4]Department of MIS, College of Business Administration, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

Corresponding authors: Muhammad Zeeshan Khan (zeeshan.khan@kics.edu.pk), Amjad Rehman (rkamjad@gmail.com), and Saeed Ali Bahaj (s.bahaj@psau.edu.sa)

This work was supported in part by the National Center of Artificial Intelligence, Pakistan, and in part by the Artificial Intelligence & Data Analytics Lab (AIDA), CCIS, Prince Sultan University, Saudi Arabia.

ABSTRACT Corona Virus is a pandemic, and the whole world is affected due to it. Apart from the vaccine, the only cure for this drastic disease is to follow the rules and regulations that avoid further spread. There are different mechanisms like (Social Distancing, Mask Detection, Human occupancy etc.) through which we can able to stop the spread of the coronavirus. In this paper, we proposed hotspot zone detection using the computer vision techniques of deep learning. We have defined the hotspot area as the particular region on which the person touches more than some specified threshold. We further mark that area to some specific color to help the authority take necessary action and disinfect that particular place. To implement this algorithm, we have utilized the human-object interaction concept. We have extracted the dataset of person classes from the publicly available dataset for the person detection and the self-generated dataset to train the algorithm. Different experiments on object detection algorithms (YOLO-v3, Faster RCNN, SSD) for person detection have been performed in this work. We achieved the maximum accuracy in real-time on the YOLO-v3 for person detection. Whereas we have marked the specific area using the template matching algorithm of computer vision techniques. Our proposed algorithm detects the persons and extracts the region of interest points on which the user draws the rectangle. Then we find the intersection over union ratio between the detected person and the region of interest of the marked area to make the decision. We have achieved 88.72% accuracy on person detection in the local environment. Whereas, for the whole system of human-object interaction for detecting the hotspot area zone, we have achieved 86.7% accuracy using the confusion matrix.

INDEX TERMS Convolution neural network, object detection, person detection, hotspot zone, fine-tuning.

I. INTRODUCTION

The analysis suggests that the coronavirus is probably originated from the bats and transmitted to the other animals before going into the humans from the Wuhan (China) wet market in December 2019. Soon after that, it is spread like a fire in a forest and wrapped the whole world. In the initial phases, most countries imposed a lockdown to stop the spread of this deadly disease. But this is not a practical solution as the whole economy of the particular country goes down. Especially it creates a catastrophic situation for underdeveloped countries in terms of economy.

Multiple companies have launched the vaccines in different countries. But, to be fully vaccinated in the world is a time

The associate editor coordinating the review of this manuscript and approving it for publication was Francisco J. Garcia-Penalvo.

taking process and, roughly it takes approximately five years for the full-fledged immunization to the world's population. So, there are two possible solutions, either the vaccine is available for everyone or the people follow the SOPs to avoid the spread of the coronavirus. Furthermore, on the other side, the possible solution is to follow the SOPs set by the World Health Organization (WHO) for the prevention of this disease.

To implement the SOPs in public places like Railway Stations, Airports, Metropolitan Bus Stations, states have issued the instructions to the heads of the concerned authorities. But it has been found that people are not following the SOPs until some penalty has not been imposed in most cases. So the authorities have taken help from law enforcement agencies to implement the SOPs on the general public strictly. This approach is not good for any part of the world. Because we cannot place so many security persons in that public places to avoid the covid19.

So, to tackle this issue, government agencies and the medical field are now looking at the artificial intelligence community to take the necessary actions. The artificial intelligence industry can check the public's behavior, whether they follow the SOPs or not. Different applications are helpful to control the spread of covid19. These applications include the prediction of statistical analysis of Covid19 patients into some specific region, find out the origin of this disease etc. Moreover, through computer vision using machine learning, we can implement the SOPs related to the prevention of Covid19 patients. These applications are non-interactive like Face Attendance System, Automatic Mask Detection from Face, Footfall Measuring, Maintaining the Social Distance of 6ft among people, hotspot zone detection etc.

In this paper, the research work is carried out on detecting the hotspot zone detection. Hotspot zone spot detection is defined as the particular area where the number of touches exceeds some particular threshold. In covid19, it is considered a dangerous act because there is a possibility that germs may be stayed out there and transmit through hands upon touch. Therefore, this research work will be helpful to identify such spots and informed the concerned authorities to disinfect that place. In this proposed work, research focuses on hotspot zone detection in public places like Railway Stations, Shopping Centers, hospitals, and any indoor public places.

Computer Vision is a major domain in artificial intelligence. Many tasks have been solved using computer vision like image classification, video classification, object detection, and image generation. Hotspot zone detection is also detected using computer vision approaches. In computer vision, tasks have been solved using two different approaches. The first one is based on the traditional approach of feature engineering like Histogram Oriented Features (HOG), Scale Invariant Features Transform (SIFT), Speeded Up Robust Features (SURF) and then trained these features are trained on classifier like Support Vector Machine.

On the other hand, the second approach is deep learning. In deep learning, automatic feature selection has been made using the convolution neural network. Deep learning-based algorithms require a huge amount of dataset along with good computation power. In this proposed work, the research has been carried out on deep learning-based approaches. We have passed the video frames to our proposed fine-tuned network. The first time, we have to mark the location by drawing the ROI using image processing techniques on the suspicious spots present in the frame. Then our proposed work detects the human into the frame and finds out the intersection over the union between the ROI of detected human and suspicious objects. If the intersection over union value exceeds the specific threshold, it will be marked as the infected area. We have set the counter for touch value, if it increases to 10 times, it will be marked as the infected place. The following contributions have been made through this research work.

1. A hotspot zone area is defined as the particular place, where the number of persons touches more than a specific threshold and needs to be disinfected to stop the spread of the virus. In this research work, we used artificial intelligence based on computer vision techniques to identify the hotspot zone area to implement the Covid19 SOPs.
2. The proposed work is a hybrid of the traditional machine learning techniques with the latest deep learning-based approaches. We first marked the suspicious region by drawing the bounding box around it. Then we detect the human using the fine-tuned Yolo-v3 version. After that, we checked whether humans interact with that marked region using the intersection over union approach. The approach is quite simple but quite efficient and innovative.
3. Generation of the dataset for this novel task by the amalgamation of locally generated dataset with the publically available dataset of human detection.

II. LITERATURE SURVEY

Humans witnessed a disastrous natural calamity with the inception of a new decade in the new year 2020. The world reckoned this adversity as COVID-19. It was first discovered in a Chinese city, Wuhan [1]. Consequently, it spread across the globe. Many researchers are working to control this disease with the help of the latest technology. One of the main technologies to stop the spread of Covid19 is artificial intelligence. Computer vision is the major domain of artificial intelligence. It has been widely used in vision-related tasks like image classification [2], video classification [3], object detection and segmentation [4], [5] and image generation [6]. Many researchers have been working on developing such techniques that can help in implementing the SOPs of the Covid19. These algorithms and applications are utilized to maintain the social distancing, control the human occupancy, mask detection, detection of those activities associated with the Covid19 (Handshaking, Hugging), Hotspot Zone Detection etc.

In this paper, we focused on Hotspot zone detection. So, in the literature survey section, we discussed the different techniques involved in identifying the hotspot zone detection. In our proposed architecture, there are different algorithms involved in identifying the hotspot zone detection. These algorithms are based on human detection and human re-identification tasks. Two approaches have been utilized in computer vision; the first is to use feature engineering and then train these extracted features using some classifier like Support Vector Machine (SVM). In contrast, another technique is deep learning. After deep learning came into being, the accuracies of the image-related task have been increased tremendously. However, deep learning-based algorithms require computation power along with the large-scale dataset. A detailed literature review of human detection and re-identification using both techniques has been described in the next section.

A. HUMAN DETECTION AND RE-IDENTIFICATION

Computer vision started to emerge as a field in the 1960s [7]. Its goal was to try to imitate human vision systems and ask computers to tell us what they see and automate the image analysis process. As computer vision evolved, algorithms began to be programmed to solve individual challenges. Moreover, the accuracies and efficiency of machine vision-related algorithms have also increased. In this section, we discussed object detection part of computer vision. Same as the computer vision stages, object detection has also done using the two different approaches. The first approach uses the traditional algorithms that were developed before 2014, and the second one is based on the deep learning-based algorithms, and it was the era after 2014. First, we discussed the traditional approaches for object detection algorithms. Viola and Jones [8] developed the algorithm for face detection, which they named the Viola-Jones face detector. The authors have utilized the simplest method to detect the object. They slide the window of a fixed size to the whole image at each corner and side to detect the face from it. Although it seems the simplest technique, it requires a huge computation at that time. Later, they have introduced the three different techniques on their proposed algorithm to improve the speed of the detection. These techniques include detection cascades, feature selection, and integral image. The drawback of this detector is that it cannot tackle the scale, illuminations and translation problem.

Dalal and Triggs [9] proposed the Histogram of Oriented Gradients (HOG). At that time, it is considered one of the best algorithms for detection because it tackles the scale-invariant feature transform and shape context very well. This algorithm is computed on the uniformly spaced cells in the dense grid by normalization and overlapped normalized local contrast. Furthermore, HOG is used to detect the number of different classes, but it is particularly designed to detect pedestrians. HOG algorithm has changed the input image into multiple sizes, but the sliding window remains the same for all orientations. HOG has remained one of the best object detectors for many years, along with a lot of applications.

Another technique named as the deformable part-based model (DPM) was proposed in 2008 by Felzenszwalb *et al.* [10]. Their proposed technique was the winner of the VOC challenge from 2007 to 2009. This algorithm sees the peak of the object detection algorithm using the traditional approaches. This work was actually the extension of the HOG algorithm. DPM was based on the divide and conquer rules; for instance, the object is first breaking into the parts, and then the decision has been concluded based on the inference which are drawn from these broken parts. The detection of the car is done by identifying the wheels, windows, and body of the car based on the learning of these parts. DPM consist of two types of filters. The first one is the roof filter and the second one is the part-based filter. In part-based filters, rather than giving each filter's size and location, a weekly supervised learning technique has been done to identify the filter size and location automatically. Although they have achieved so much good accuracy, some areas like negative mining of regions, bounding box regression, etc., were where this algorithm failed.

After 2010, the performance of the object detection algorithms become decreases due to the saturated behavior of the traditional features. So, in 2012, the rebirth of the convolution neural network has been taken place. Girshick *et al.* [11] proposed the model in 2014, which is also the start of the RCNN family. It is also the initiative to the deep learning-based object detection algorithms. This algorithm first extracts the 2000 regions from each image using the selective search method. Then these features are reshaped into the fixed image form. Finally, this image is passed to some convolutional neural network (CNN) to extract the convolution features. These features are then classified using the Support Vector Machine classifier to identify whether the object is present in that region or not. The drawback of this methodology is that the algorithms have to perform a lot of computation. Because first, it extracts 2000 regions for each image and then these regions are passed to the CNN to check the presence of an object. So, thus it is computationally not effective.

He *et al.* [12] proposed the SPPNet. The main contribution of this network is to introduce spatial pyramid pooling. In CNN networks like AlexNet, a fixed size input has passed for feature extraction like 224 × 224. The spatial pyramid pooling generates the fixed-length representation without affecting the image's region of interest and size. Although SPPNet achieves state-of-the-art accuracy, it still has some drawbacks, like it is a multi-stage network, and we did just fine-tuning the fully connected layers without affecting the previous layers. To overcome the drawbacks of the RCNN and SPNet, Fast RCNN has been proposed by Girshick [13]. In Fast RCNN, CNN trained the detector and bounding box regressor rather than passing the features to CNN for classification and object detection. Although it covers the drawbacks of the RCNN and SPNet, its detection accuracy is inaccurate because of no trainable region proposal network. This drawback has been overcome after the invention of the Faster RCNN [4].

In Faster RCNN, there are two major networks. First one is the backbone architecture and the second one is the region proposal network. In backbone architecture, convolutional features have been extracted. Backbone architecture based on some convolutional neural network like AlexNet, VGG16 etc. After extracting the features from the backbone, these convolution features are further passed to the region proposal network. Region proposal network is the trainable network, which proposed regions having the probability of containing the object. After getting the bounding box points, the particular box is passed to the region of interest pooling and bounding box regressor to reshape the box by removing extra area. This will help to reduce the extra area around the object. But, Faster-RCNN failed to achieve the results in real-time as compared to the YOLO (You Only Look Once) algorithm.

Lin *et al.* [4] proposed the network for object detection named as the feature pyramid network. Before the FPN, no one has extracted CNN's low layer features, although it is important in category recognition. Most of the algorithms utilized the top layer features for object detection. Whereas, the features present at the bottom layers are also useful for object localization. Hence, it presents the top-down architecture to detect and localize the object at different scales. In CNN-based backbone architecture, the feature pyramids normally formed in the forward direction. The major difference between the feature pyramid network and the other object detection models is that previous object detection models only detect objects from the top layer features. At the same time, FPN detects objects from multiple layers.

The RCNN family algorithms are not good in terms of speed because they are involved in multiple stages for object detection and recognition. Redmon *et al.* [15] first introduced the one-stage deep learning-based object detection model. YOLO (You Only Look Once) is extremely fast and almost runs along with the processing speed of 155 fps. As name suggested that its working paradigm is different from the previous deep learning-based object detection models. The YOLO model is based on totally different strategies. This algorithm applies a convolution neural network on the whole image, divides it into regions, and identifies the bounding box and its object. Later, the authors also proposed the different versions of the Yolo.

Lin *et al.*, [16] proposed the network architecture to find out why the low accuracy of single-stage architecture compared to the two-stage detector. Furthermore, after the research, they found that the dense background and foreground images create an unbalancing situation during the training. To tackle this problem, a new loss function has been utilized named the focal loss.

These all-detection algorithms, which we discussed in the literature, include the traditional and deep learning approaches utilized for object detection. Since, human is also categorized as an object. So, different features have been extracted to identify the object as a human for the detection of human. These features include the shape of the object, texture and motion features of the detected object.

Now, we will have discussed some of the work used for particularly human detection. The algorithms which utilized the shape-based features extracts the moving points and blobs to identify the human. Unfortunately, this algorithm does not perform well in a generic environment and can only perform in a controlled environment. Therefore, they have utilized the template matching techniques for person detection [17].

Gajjar *et.al.* [18] proposed the human detection algorithm which extracts the histogram of oriented gradient features (HOG) and trained it using the SVM classifier for human detection purpose. Similiarly, Dalal and Triggs [9] proposed the algorithm in which they utilized the texture-based features by using the Histogram Oriented Gradient (HOG). They extracted the high dimensional features such as edges and then trained them using the support vector machine (SVM) for human detection. Some researchers also detected humans through Face Detection [19] and Gait analysis [9]. Andriluka *et al.* [21] proposed the methodology in which they detect the human from the partially occluded environment using the tracklet based detector. They are able to achieve good accuracy on the person detection in occlusion. Later deep learning techniques have also been utilized for person detection. Latest deep learning algorithms like Faster RCNN, YOLO-v3 have achieved remarkable accuracy on object detection models. They have achieved almost more than 90 percent accuracy on person detection. Both Faster RCNN and YOLO-v3 [22] although outperform in person detection, but both have some pros and limitations as well on each other. YOLO-v3 is not able to recognize the smaller objects, however, it is really fast as compared to the Faster RCNN. Whereas Faster RCNN achieved very good accuracy even on the smaller objects, but it has the constrain of speed. Both architectures have utilized the MS COCO and Pascal dataset to trained their architecture [23], [24].

This paper presents research progress in the creation of applications for the identification and detection of people. Boudjit and Ramzan [25] used the convolutional neural networks (CNN) YOLO-v2 based on a drone's camera. First, deep-learning-based computer vision is used to assess the person's location and state. Then, the individual detection results indicate that YOLO-v2 can identify and classify objects with a high degree of accuracy from the aerial view.

This [26] paper introduces a semi-supervised faster region-based convolutional neural network (SF-RCNN) method for detecting people and classifying the load they carry in video data collected by high-power lens video cameras from distances of several miles. First, to detect areas that may contain a person. These areas are then fed into a faster RCNN classifier with ResNet50 transfer learning convolutional layers.

As per our best knowledge, no one utilized the amalgamation of deep learning and traditional computer vision approaches for identifying the hotspot zone detection to prevent the spread of covid-19. The Yolo-v3 have been fine-tined to gain the results in real-time for human detection. The sequence of the paper is as follows; the Proposed methodology is described in section 3. Results and discussion have

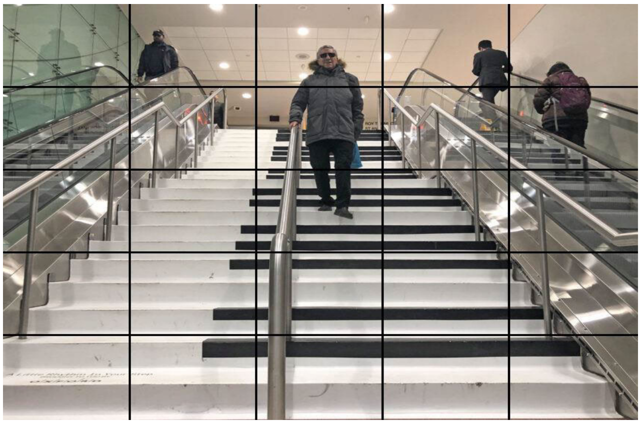


FIGURE 1. 5 × 5 grid on the image.

been discussed in section 4. Whereas, paper is concluded in section 5.

III. PROPOSED METHODOLOGY

The proposed methodology for the hotspot zone identification based on the two major steps. First, we detect the human and identify the region of interest (ROI). Then we calculate the intersection over union (IOU) between the detected human and the marked ROI. If the value of the IOU is greater than the specific threshold, then we considered that person has directly or indirectly in contact with the marked ROI. We increase the number in a counter. If it exceeds some specific threshold, we highlight the particular region until the concerned authority is informed and disinfects the particular region. For person detection, we have utilized the YOLO-v3 with the fine-tuned parameters. Whereas, for the person re-identification deep-sort algorithms have been utilized. In the consequent paragraphs, we explain the YOLO-v3 for the human detection and Deep Sort for the human re-identification with the tuned parameters.

A. HUMAN DETECTION

For human detection, we have utilized the YOLO-v3 object detection with the fine-tuned parameters. Until now, most of the object detection model is based on the different steps done by going through the visual features of the image more than once. But in YOLO case, it did not scan the image repeatedly, instead looks only once at the image to detect the objects present in the particular image. This is the main reason behind the real-time object detection for all YOLO versions. In YOLO, the whole image is divided into the SxS grids. For instance, in fig. 1 the image is divided into the 5 × 5 grid. However, in all YOLO invariants, the grid size is fixed at 7×7. If the centre of any particular object is found to be present in any of the grid, then it is the responsibility of the grid to detect the object. Since YOLO applies the 7 × 7 grid on the image, so we have total 49 cell spaces. YOLO runs classification and localization at the same time on each cell.

In YOLO, classification and localization network detect only one object at a time. So, it gives only maximum prediction of 49 objects which is obtained from each cell. Moreover, it detects only one object per cell, so if the cells on the particular images contain more than one object, then the model cannot detect it. Another issue in all YOLO invariants is that the parts of the same objects may appear in multiple cells. So, just like in figure 1, a person has been detected in the multiple cells of the grid.

For each grid, YOLO produces the bounding boxes which is set B=2 and for each bounding box, it gives the confidence score. Confidence score gives us the probability of the particular bounding box containing the object or the background. By doing so, the algorithm can prevent the detection of the background. Intersection over union is used to identify whether the predicted region of interest contains the object or not. This problem has been sort out using non-max suppression. It has been done by comparing the predicted bounding box with the ground truth bounding boxes which the human annotates.

B. NETWORK ARCHITECTURE

We utilized the YOLO-v3 architecture with the fine-tuned parameters for human detection as shown in Figure 2, so here we discuss the YOLO-v1 and its successor until the YOLO v3. YOLO v1 is built on the Google-Net architecture which is also known as the inception network. This network is trained for the classification of objects. It contains the 24 convolution layers along with the 2 fully connected layers. But YOLO-v1 not utilized the inception modules; instead, it only uses the reduction layer from the end of the convolution layer.

After that, YOLO-v2 came into being. This architecture is more accurate and efficient with the improved frame per second. YOLO- v2 architecture utilizes the DrakNet19 as a backbone architecture. It contains the 19 convolution layers and 5 max-pooling layers along with the softmax layer for the output. YOLO-v2 outperforms the previous version of YOLO-v1 in terms of frame per second, mean average precision and object classification. The same authors have proposed the YOLO-v3 which is the extended version of the YOLO-v2. It has utilized the Darknet-53 architecture as a backbone architecture to extract the features for object classification. Compared to the DarkNet19 architecture, DarkNet53 contains the residual blocks connected with the up-sampling layers to add concatenation and depth to the network. In contrast to the previous YOLO versions, YOLO-v3 generates the 3 predictions at each spatial location on different scales. This resolves one of its drawbacks that YOLO does not recognize the small objects from the given image.

Each of the prediction score is monitored by calculating the objectness, classification score and the bounding box regressor. The objectness score has been predicted using logistic regression. The objectness score is considered to be 1, if the predicted bounding box is overlapped with the ground truth bounding box. Now, if we talk about non-maximal suppression, we are used to sorting out the problem or if more than one bounding box contains the same object. This problem has been solved using some predefined threshold on

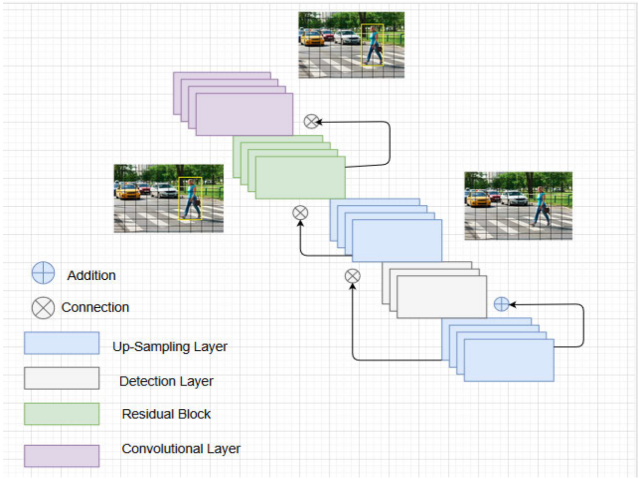


FIGURE 2. Architecture for human detection.

the intersection over union (IOU) value. If the IOU value is less than some specific threshold for some bounding boxes, they are discarded. Then, the algorithm chooses only those bounding boxes that have the highest confidence value and depicts its prediction.

C. LOSS FUNCTION

The overall loss function for our fine-tuned Yolo-v3 is calculated with the help of the bounding box regressor or (localization loss), confidence loss along with the cross-entropy. This loss function has been defined as follows;

$$\begin{aligned}&\lambda_{coord}\sum_{i=0}^{s^2}\sum_{j=0}^{b}1_{ij}^{OBJ}[(X_{i,}X_i^{\wedge})^2+(Y_i,Y_i^{\wedge})^2]\\&\quad+\lambda_{coord}\sum_{i=0}^{s^2}\sum_{j=0}^{b}1_{ij}^{OBJ}\\&\quad\times[(\sqrt{w}_i-\sqrt{w}_i^{\wedge})^2+(\sqrt{H}_i-\sqrt{H}_i^{\wedge})^2]\\&\lambda_{coord}\sum_{i=0}^{s^2}\sum_{j=0}^{b}1_{ij}^{OBJ}(c_{i,}c_i^{\wedge})^2\\&\quad+\lambda_{noobj}\sum_{i=0}^{s^2}\sum_{j=0}^{b}1_{ij}^{OBJ}(c_{i,}c_i^{\wedge})^2\\&\quad+\sum_{i=0}^{S^2}1_i^{obj}\sum_{c\epsilon classes}(P_i(C)-P_i^{\wedge}\end{aligned}\tag{1}$$

Here in equation 1, presents the three different aspects (class prediction, bounding box prediction and objectless) in the loss function. S represents the number of grids, whereas B is the bounding box predictor for each grid cell and C is the class prediction for each grid cell. X_{ij} and Y_{ij} are the bounding boxes having the center of i and j.

Here in equation 1, λ_{coord} represents the weight for the coordinate error, whereas s^2 depicts the grids present in the image. The number of bounding boxes generated per grid is represented by the B. $1_{l,m}^{noobj}=1$ depicts that bounding box m contains the object in the specific grid I; otherwise, its value equals 0.

D. DEEP SORT

Deep sort is the deep learning-based algorithm used to track the particular objects present in a video [27]. In our proposed work, we have utilized the deep sort to track the detected persons in the video stream for identifying the hotspot detection. The deep sort used the learned patterns of human detection from the images. This information has later combined with the temporal information to predict the associated trajectories of the detected objects. In addition, deep sort maintains the track of each detected object under our consideration by using the unique identifier to perform further statistical analysis on it.

Deep sort handles the different vision-related challenges like occlusion, change camera viewpoint, and data that is not stationary and dataset annotation very well. Moreover, the algorithm also utilized the Kalman Filter along with the Hungarian algorithm to make effective tracking. The better association has been achieved using the Kalman Filter because it can predict future positions by sustaining the current position. The Hungarian algorithm identifies the id attribution and association to ensure that the object present in the current frame is the same as the object present in the previous frame. We have utilized the YOLO v3 for object detection and tracking purpose. Eight-dimensional space has been utilized to describe each target using the linear constant velocity model.

$$y=[x,y,,h,u^{\wr},v^{\wr},^{\wr},h^{\wr}]^T\tag{2}$$

In equation 2, (x,y) depicts the centroid of the predicted bounding box, λ represents the aspect ratio, and h shows the height of the image. Remaining variables shows the respective velocities. After that Kalman filters along with the constant velocity motion and having the linear model are being utilized. Whereas the bounding box coordinates (x,y,λ,h) which depicts the current object state, have also been taken under consideration. The total frames for the particular chunk of track k along with the association with object a_k are calculated. There is a counter which is to be set and incremented till then the association remains to the particular object, after that the counter value is set to zero again. Furthermore, if the track objects exceed some specific predefined limit, then tracking from the object has been removed, and thus the same process has been starting again. If the detected objects are not fulfilling the tracking criteria, then the process is initiated again, and maps are generated for the newly detected objects. The tracking is considered as indefinite for the first three frames of any chunk and if it continues to track any particular detected object, then we keep that object for tracking; otherwise, it is discarded.

After that Hungarian algorithm has been utilized, which is used to map the measurements between the newly arrived objects and already detected objects using the Kalman tracking. The Hungarian algorithm utilized the motion and appearance-based information using the Mahalanobis distance using the following equation;

$$d(x,y)=(D_y-i_x)^T S^{X-1}(D_{y|}-i_x)\tag{3}$$

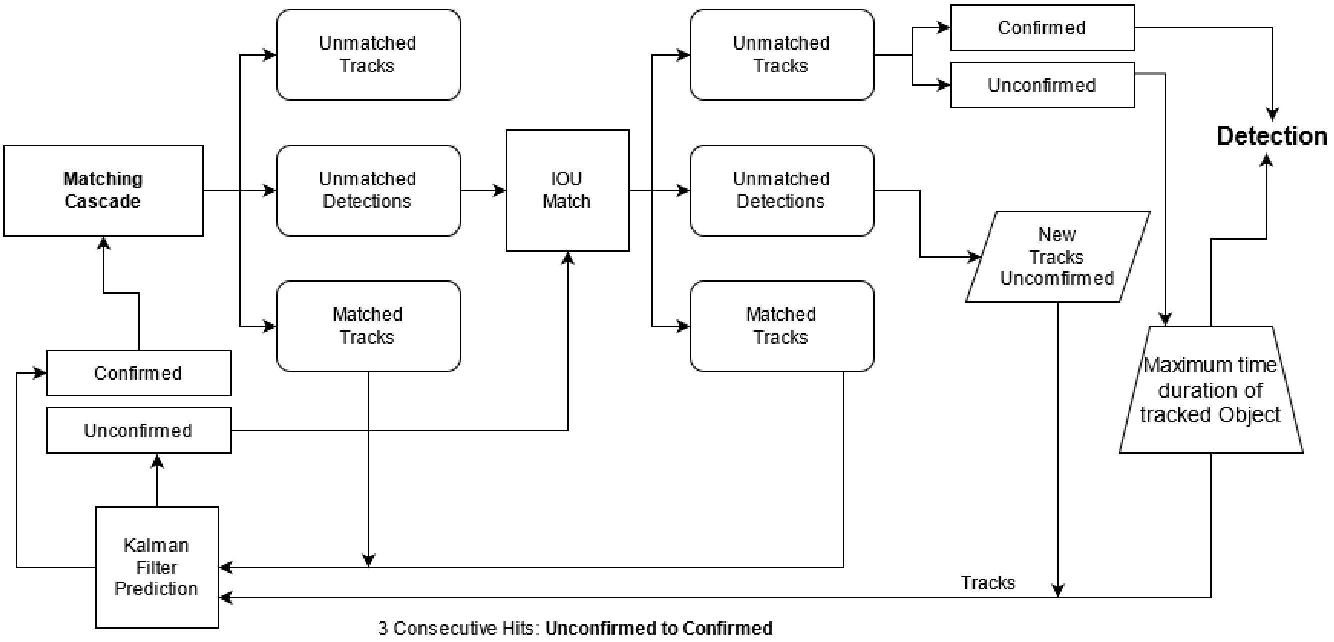


FIGURE 3. Pseudo code for tracking using deep sort.

In equation 3, x^{th} track projection has been represented by using the (i_x, S_x). Whereas y^{th} bounding box detection showed by the $D_{y'}$. So, this distance also helpful to find out the uncertainty in tracking. It calculates the count using the standard deviation from the mean track location. So, directly it helped to discard the unlike associations by setting the threshold on the Mahalanobis distance. The pseudo-code for tracking using the deep sort have shown in figure 3.

IV. RESULTS AND DISCUSSIONS

A. DATASET ACQUISITION AND PRE-PROCESSING

Deep learning-based object detection model requires a huge amount of data, if the training is required from scratch. So, the possible solution is to fine-tuned some pre-trained model on our dataset to generate the required results [28] in a small amount of data. In this proposed work, the object detection model is fine-tuned for human detection purposes using the TensorFlow, the deep learning framework. The model is trained on the dataset which was generated by ourselves with the amalgamation of self-generated dataset, MS COCO [23] dataset, Pascal [24] dataset and open images dataset [29]. MS COCO, Pascal and Open Images dataset are publically available on the worldwide web and contains the person class. We have extracted the Person class data from these three described datasets. Moreover, to train the model so that it also performs well in the local environment, we have also collected the person images from the local community in a domestic environment. We gathered total 3000 images from local environment and 1000 images from each of the above described publically available dataset (MS-COCO, Pascal, Open Images). We have total 6000 images. After gathered the dataset from different sources, we then perform the pre-processing steps. The pre-processing includes the removal of unnecessary and raw frames. Moreover, we have also done the annotations of these images, where the person class is present. The dataset is split into the 80 percent training data and 20 percent testing data. So, 4800 images are in training fold, and the rest of the images are in the testing fold. Sample frames have shown in the figure 4.

FIGURE 4. Sample frames from the dataset.

B. IMPLEMENTATION DETAILS AND EVALUATIONS

The training is carried out on the Nvidia-1080 Ti GPU which has the memory capability of 11 Gb. The training takes almost 6-7 hours for fine-tuning the model. At the start, we have set the learning rate 0.001 and increase it with the factor of 10^{-1}, if the training and testing accuracy stops to converge. To evaluate the accuracy against each, we have utilized the mean average precision loss function. The loss function calculates the difference between the actual label value and the predicted label value. The learning of the particular machine learning algorithm is measured with the help of the loss function value. If the loss function value deviates too much from the actual

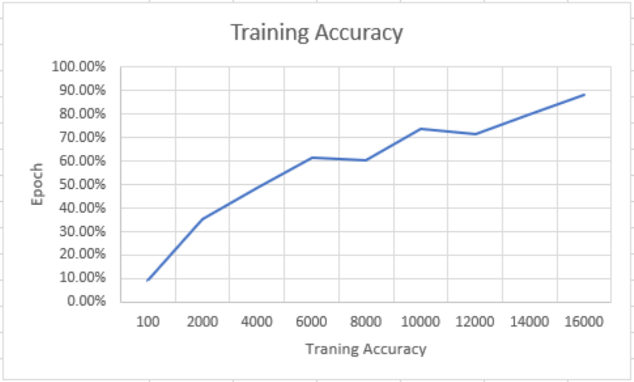


FIGURE 5. Training accuracy of person detection.

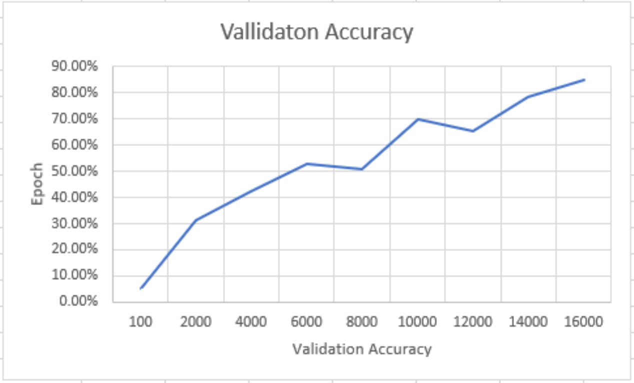


FIGURE 6. Validation accuracy of person detection.

data, the loss value will be high for a particular algorithm. The loss value then optimized using the optimizer function and set out some convolution neural network architecture parameters. Mean square error is one of the most utilized loss function in deep neural network architectures. This has been calculated by measuring the difference between the actual values and the predicted values. Following equations have been used to calculate the mean square error.

$$\frac{1}{n}\sum\nolimits_{N-1}^{n}(p_n^{\wedge}-p_n)^2 \qquad (4)$$

In equation 4, N depicts the specific images present in the dataset, which goes up to the n^{th} sample. Whereas, $p_n^{\wedge}$ presents the predicted label and p_n shows the actual label. In the training of the deep neural network, one of the main issues is over-fitting. Over-fitting happens when the trained model shows good accuracy on the training data, but failed to perform on the testing data. To overcome the overfitting process, we have utilized the dropout function. The value of the dropout is set to be 0.7 in our fine-tuned person detection architecture. The weights have been optimized using stochastic gradient descent. The model is trained till the 16000 epochs. Training and validation accuracy has been shown in figure 5 and 6, respectively. Both graphs show that how training and validation accuracy goes up. However, the training and validation loss for person detection have been depicted in figure 7.

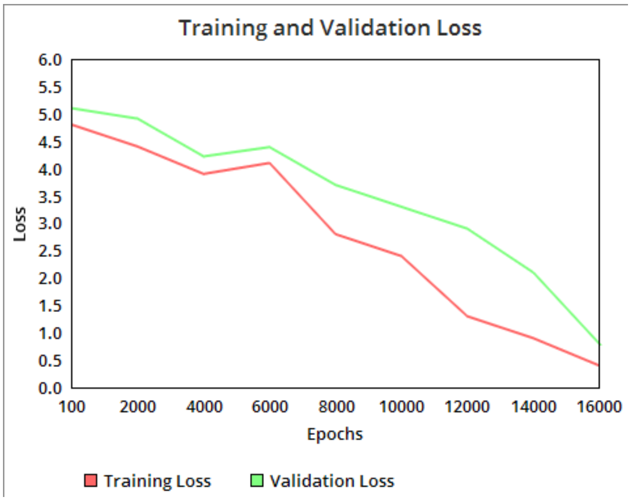


FIGURE 7. Training and validation loss on person detection.

TABLE 1. Comparison of different object detection models.

Model	mAP	FPS
Faster RCNN [4]	96.12	3
SSD [30]	68.23	10
Fine-tuned YOLO v3	84.81	22

The model has also been evaluated by training the same dataset on the other state-of-the-art detection architectures. The numeric results are presented in Table 1.

From table 1, it is clearly observed that although the Faster RCNN achieved good accuracy, but in terms of frame per second, fine-tuned Yolo-v3 surpass the other two algorithms. Since we have to make the decision in real-time related to the hotspot detection, so that's why Yolo-v3 is most preferable. Similarly, we have performed the experiments on Single Shot Detector [30] algorithm. This algorithm cannot achieve the good results and its (Frame per second) Fps are also very low compared to the other algorithms in the table 1. So, that's why we have adapted our fine-tuned Yolo-v3 in our proposed methodology because of its real time results along with good accuracy. In table 1, the comparison is only made with state of the art object detection algorithms with respect to the human. Because in the proposed methodology correctly identification of the human detection is the backbone. Whereas the user draws a region of interest at the suspicious place.

Furthermore, IOU (intersection over union) is calculated by measuring the intersection between the bounding boxes of detected human and human drawn ROI. Finally, both (IOU and User Drawn ROI) is evaluated using quantitative evaluation for robust experiments.

The explanation and working of the proposed work as depicted in figure 8 are as follows in table 2.

Our experiments have put the threshold value 10 according to the number of visitors of the area under examination. We set the appropriate value according to the circumstances of the crowd. However, it is totally controllable by the

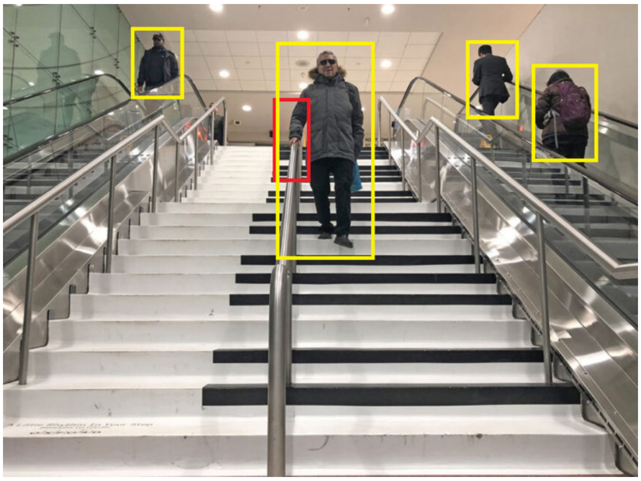


FIGURE 8. Hotspot detection using the proposed architecture.

TABLE 2. Pseudo code for Hotspot zone detection.

Pseudo Code	
Input: User Draw Rectangle on a suspicious point (Red Rectangle in the Image) on the coming video stream.	
Output: The output is generated using the following steps	
Step1	The persons are detected using our fine-tuned Yolo-v3 (Yellow rectangle in the image) version, as shown in the yellow rectangles in the image.
Step2	The intersection over union (IOU) is calculated between the user-drawn rectangle (Red Rectangle) and the human-detected rectangles Yellow Rectangles.
Step3	If IOU is more than the specific threshold between red and yellow box, we considered that humans are present in that hotspot zone and considered a violation.
Step4	We set the counter to count the number of persons over there.
Step5	Tracking of that particular person using the deep sort to reduce the false positive in the counter.
Step6	Set the counter threshold of 10, if the number of violation exceeds to that threshold, concerned authority has been notified to take the necessary actions.

administrator of the particular area. The administrator can increase or decrease its value based on the visitor's ratio. We passed 60 different videos to our proposed system for quantitative evaluation to detect hotspot zone detection and manually check its perfection. The results of these surveys are elaborated in table 3. Total 30 videos for both (Hotspot Zone, No Hotspot Zone) were utilized for the quantitative evaluation purpose.

Table 3 depicts the confusion matrix of hotspot zone detection. Each video has a length of 30 seconds. From 30 videos of hotspot zone, our system correctly identifies hotspot zone from 26 videos. This means that human is detected perfectly near our drawn suspicious region and based on the intersection over union (IOU) value, we can identify that danger zone. Whereas, in four videos, the system cannot figure out the zone because of not able to detect if the human is near our specified region.

TABLE 3. Quantitative evaluation of hotspot zone detection.

	Hotspot Zone	No Hotspot Zone
Hotspot Zone	27	3
No Hotspot Zone	5	25

Similarly, we have drawn the (Region of Interest) ROI far from the human pathway for no hotspot zone. There is no intersection between our drawn ROI and the human detected ROI. Our system correctly identifies that no hot spot zone in 23 videos, whereas in 7 videos, we get the false positive values. We achieved 86.7% accuracy on this quantitative evaluation.

V. CONCLUSION

This paper proposed a novel framework for identifying the hotspot zone detection using the state-of-the-art deep learning approach. This task has been done by identifying the areas under consideration and then finding out the intersection area between the detected person and marked ROI (region of interest). The counter has been initialized and increased as the number of violations proceed. We have set the threshold of 10. If the interaction between the human and the marked ROI exceeds the set threshold, then the area marked with red and corresponding authority get notified. For person detection, extensive experiments have been performed using the Faster-RCNN, Yolo-v3 and SSD. We have achieved good accuracy with real-time response on our fine-tuned Yolo-v3 architecture. The proposed work is fully automatic and will help to reduce the spread of Covid19.

ACKNOWLEDGMENT

The authors would like to express their earnest gratitude to the National Centre of Artificial Intelligence Pakistan Fund and Organization (KICS) to support their research work fully. In addition, they would also like to acknowledge the support of the Artificial Intelligence & Data Analytics Lab (AIDA) CCIS Prince Sultan University Riyadh KSA for the payment of Article Processing Charges (APC) of this publication.

REFERENCES

[1] *How Did Coronavirus Start and Where Did it Come From?* Accessed: Sep. 3, 2020. [Online]. Available: https://www.theguardian.com/world/2020/apr/28/how-did-the-coronavirus-start-where-did-it-come-from-how-did-it-spread-humans-was-it-really-bats-pangolins-wuhan-animal-market

[2] A. Krizhevsky, I. Sutskever, and H. Ge, "Image-net classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[3] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detectionwith region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2961–2969.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[7] T. Huang, "Computer vision: Evolution and promise," CERN Eur. Org. Nucl., Res. Rep., 1996, pp. 21–26.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2008, p. 1.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2005, pp. 886–893.

[10] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[11] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[12] M. Yang, B. Li, H. Fan, and Y. Jiang, "Randomized spatial pooling in deep convolutional networks for scene recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 346–361.

[13] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[14] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 2.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[17] M. Singh, A. Basu, and M. K. Mandal, "Human activity recognition based on silhouette directionality," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1280–1292, Sep. 2008.

[18] V. Gajjar, Y. Khandhediya, and A. Gurnani, "Human detection and tracking for video surveillance: A cognitive science approach," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2805–2809.

[19] P. Huang, A. Hilton, and J. Starck, "Shape similarity for 3D video sequences of people," *Int. J. Comput. Vis.*, vol. 89, nos. 2–3, pp. 362–381, Sep. 2010.

[20] D. Cunado, M. S. Nixon, and J. N. Carter, "Using gait as a biometric, via phase-weighted magnitude spectra," in *Proc. Int. Conf. Audio-and Video-Based Biometric Person Authentication*. Berlin, Germany: Springer, 1997, pp. 93–102.

[21] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[22] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[23] L. TY, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and Z. Cl, "Microsoft COCO: Common objects in context," *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Sep. 2014, pp. 740–755.

[24] 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.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[25] K. Boudjit and N. Ramzan, "Human detection based on deep learning YOLO-v2 for real-time UAV applications," *J. Experim. Theor. Artif. Intell.*, vol. 3, pp. 1–18, Apr. 2021.

[26] H. Wei and N. Kehtarnavaz, "Semi-supervised faster RCNN-based person detection and load classification for far field video surveillance," *Mach. Learn. Knowl. Extraction*, vol. 1, no. 3, pp. 756–767, Jun. 2019.

[27] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.

[28] M. Z. Khan, S. Harous, S. U. Hassan, M. U. Ghani Khan, R. Iqbal, and S. Mumtaz, "Deep unified model for face recognition based on convolution neural network and edge computing," *IEEE Access*, vol. 7, pp. 72622–72633, 2019.

[29] 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," in *Proc. Int. J. Comput. Vis.*, vol. 13, Mar. 2020, pp. 1–26.

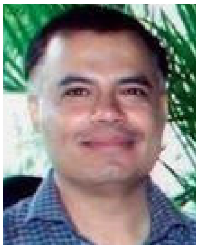
[30] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Oct. 2016, pp. 21–37.

MUHAMMAD ZEESHAN KHAN received the M.S. degree in computer science from the University of Engineering and Technology Lahore, Pakistan. He is currently a Team Lead with the Intelligent Criminology Lab, National Center of Artificial Intelligence, Al Khawarizmi Institute of Computer Science, UET Lahore. His research interests include computer vision, machine learning, and deep learning.

MUHAMMAD USMAN GHANI KHAN received the Ph.D. degree from The University of Sheffield, U.K. He is currently a Professor with the Department of Computer Science, University of Engineering and Technology Lahore, Pakistan, where he is also the Principle Investigator of Intelligent Criminology Lab, National Center of Artificial Intelligence, Al Khawarizmi Institute of Computer Science (KICS). His study was concerned with statistical modelling for machine vision signals, specifically language descriptions of video steams.

TANZILA SABA (Senior Member, IEEE) received the Ph.D. degree in document information security and management from the Faculty of Computing, Universiti Teknologi Malaysia (UTM), Malaysia, in 2012. She is currently serving as a Research Professor with the College of Computer and Information Sciences, Prince Sultan University (PSU), Riyadh, Saudi Arabia. She has more than 200 publications that have around 4000 citations with H-index of 40. Her primary research interests include bioinformatics, pattern recognition, machine learning, and applied soft computing. She received the best researcher awards from PSU, in 2014, 2015, 2016, and 2018. She has supervised the Ph.D. and M.S. students. Due to her excellent research achievement, she is included in Marquis Who's Who (S&T) 2012. She won the Best Student Award from the Faculty of Computing, UTM, in 2012. She is currently an editor of several reputed journals and on panel of TPC of international conferences.

IMRAN RAZZAK (Member, IEEE) is currently a Senior Lecturer with the School of Information Technology, Deakin University, Australia. He has published more than 70 refereed articles. His research interests include machine learning and data analytics in general, particularly in the healthcare industry. He is a passionate health informatician who wants to make the healthcare industry a better place through informatics.



AMJAD REHMAN (Senior Member, IEEE) received the Ph.D. degree (Hons.) in forensic documents analysis and security from the Faculty of Computing, Universiti Teknologi Malaysia, in 2010, where he received the Rector Award for best student in the university. He is currently a Senior Researcher with the Artificial Intelligence and Data Analytics Lab (AIDA), Prince Sultan University, Riyadh, Saudi Arabia. He is the author of more than 200 indexed journal articles. His keen interests include data mining, health informatics, and pattern recognition.

SAEED ALI BAHAJ received the Ph.D. degree from Pune University, India, in 2006. He is currently an Associate Professor with the Department of Computer Engineering, Hadramout University, and also an Assistant Professor with Prince Sattam Bin Abdulaziz University. His main research interests include artificial intelligence, information management, forecasting, information engineering, big data, and information security.