IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS. 2017.Doi Number

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[2], Tanzila Saba[4], Imran Razzak[3], Amjad Rehman[4], Saeed Ali Bahaj[5]

[1]Intelligent Criminology Research Lab, NCAI, KICS, UET, Lahore, Pakistan, 54000 (email: zeeshan.khan@kics.edu.pk)
[2] Intelligent Criminology Research Lab, NCAI, KICS, UET, Lahore, Pakistan, 54000 (email: usman.ghani@kics.edu.pk)
[3] School of Info Technology, Deakin University, Geelong, Australia (email imran.razzak@deakin.edu.au)

[4] Artificial Intelligence & Data Analytics Lab (AIDA) CCIS Prince Sultan University Riyadh 11586 Saudi Arabia (drstanzila@gmail.com & rkamjad@gmail.com)

[5] MIS Department College of Business Administration Prince Sattam bin Abdulaziz University Alkharj Saudi Arabia (s.bahaj@psau.edu.sa)

Corresponding author: M. Zeeshan Khan (email: zeeshan.khan@kics.edu.pk)

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 following the rules and regulations that prevent further spread. There are different mechanisms like (Social Distancing, Mask Detection, Human occupancy etc.) through which we can able to stop the spread of corona virus. In this paper, we proposed hotspot zone detection using the computer vision techniques of deep learning. We have defined the hotspot area on which the person touches more than to some specific threshold. We further mark that area to some particular color, which will 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 the object detection algorithms (YOLO, 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 87.7% accuracy on person detection. Whereas, for the whole system of human-object interaction for detecting the hotspot area zone, we have achieved 81.7% accuracy using the confusion matrix.

INDEX TERMS CNN, Object Detection, Person Detection, Hotspot Zone, Fine-tuning

1. INTRODUCTION

The analysis suggests that corona virus 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 whole economy of the particular country goes down. Especially it creates a catastrophic situation for underdeveloped countries in terms of economy.

So, there are two possible solutions, either the vaccine came into the market for the cure of this catastrophic disease, or the people follow the SOPs to avoid the spread of the coronavirus. Presently, multiple companies in different countries are working on the development of the vaccine. This is a time taking process because it includes a lot of testing on different species including humans. So, roughly it takes approximately one year for the full-fledged development of the covid19 vaccine. 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 in most of cases, people are not following the SOPs until some penalty has not been added. So these authorities have taken the help from law enforcement agencies to implement the SOPs by the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

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 for the avoidance of the covid19.

So, to tackle this issue, government agencies and medical field are now looking at the artificial intelligence community to take the necessary actions. The artificial intelligence industry can check the behavior of the public, whether they follow the SOPs or not. Different applications are helpful to control the spread of covid19. These applications include predicting the statistical analysis of Covid19 patients into some specific region, finding 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.

This 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 specific threshold. In covid19 era, it is considered a dangerous act because there is a possibility that germs may be stayed out there and transmit through hands upon touch. This research work will be helpful to identify such spots and informed the concerned authorities to disinfect that place. This proposed work focuses on hotspot zone detection in public places like Railway Stations, Shopping Centers, hospitals, and any indoor public places.

Computer Vision is 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 done 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 the 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 union between the ROI of detected human and suspicious objects. If the intersection over union value more than the specific threshold, it will be marked as the touch. We have set the counter for touch value; if it increases to 20 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 a 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 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 the human 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 for the human detection.

2. 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 developing such techniques that can help implement the SOPs of the Covid19. These algorithms and applications are utilized to maintain the social distancing, control the human occupancy, mask detection, and detect those 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 separately 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. In computer vision, two approaches have been utilized now; the first is to use feature engineering and then trained these extracted features using some classifier like Support Vector Machine (SVM).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

Whereas another technique is deep learning. After deep learning came into being, the accuracies of the image-related task have been increased tremendously. Deep learning-based algorithms require computation power along with the large-scale dataset. In the next section, a detailed literature review of human detection and re-identification using both techniques has been described in detail.

2.1 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 accuracy and efficiency of the machine vision-related algorithms have also increased. In this section, we discussed the object detection part of computer vision. Like the computer vision stages, object detection has also been done until now using two different approaches. The first approach is using the traditional algorithms developed before 2014, and the second one is based on the deep learning-based algorithms and the era after 2014. First, we discussed the traditional approaches for object detection algorithms. P. Viola and M. 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, yet it demands high computation at that time. Later, they have introduced the three different techniques on their proposed algorithm to improve detection speed. 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.

N. Dalal et.al [9] proposed the Histogram of Oriented Gradients (HOG). At that time, it is considered as 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 using the normalization and the 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 same for all orientations. HOG has remained one of the best object detectors for many years along with the lot of applications.

Another technique named as the deformable part-based model (DPM) was proposed in 2008 by the P. 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 actually based on the divide and conquer rules, for instance the object is first breaking into the parts, and then decision have 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. First one is the roof filter and the second one is the part-based filters. In part-based filters, rather to give the size and location of each filter, a weekly supervised learning technique have been done to identify the filter size and location automatically. Although, they have achieved so much good accuracy, but there are some areas like negative mining of regions, bounding box regression etc. where this algorithm failed.

After the 2010, the performance of the object detection algorithms become decreases due to the saturated behavior of the traditional features. In 2012, the rebirth of the convolution neural network has been taken place. R. Girshick [11] proposed the model in 2014, which is also the start of the RCNN family. It is the 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. This image is passed to some convolutional neural network (CNN) to extract the convolution features. These features are then classify 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 performed the 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 object. So, thus it is computationally not effective.

In 2014 K. He et al. [12] proposed the SPPNet. The main contribution of this network is to introduce the spatial pyramid pooling. In CNN networks like AlexNet, a fixed size input has passed for feature extraction like 224x224. The spatial pyramid pooling generates the fixed length representation without affecting the region of interest and size of the image. Although SPPNet achieves the state-of-the-art accuracy but still it has some drawbacks like it is a multi-stage network and we did just fine tuning of the fully connected layers without affecting the previous layers. To overcome the drawbacks of the RCNN and SPNet, Fast RCNN have been proposed by R. Girshick [13]. In Fast RCNN, rather to passed the features to CNN for classification and object detection, CNN itself trained the detector and bounding box regressor. Although, it covers the drawbacks of the RCNN and SPNet, but its detection accuracy is not so accurate, because of no trainable region proposal network. This drawback has been overcome after the invention of the Faster RCNN [14]. 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,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

which proposed regions having the probability of containing the object. After getting the bounding box points, particular box is passed to the region of interest pooling and to bounding box regressor to reshaped 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. [15] proposed the network for object detection named as the feature pyramid network. Before the FPN, no one have extracted the low layer features of CNN, 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 is also useful for the 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 are that previous object detection models only detect objects from the top layer features. Whereas, FPN detect objects from the multiple layers.

The RCNN family algorithms are not good in terms of the speed, because they involved in multiple stages for object detection and recognition. R. Joseph et al. [16] 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 totally different from the previous deep learning-based object detection models. The YOLO model is based on the totally different strategies. This algorithm applies convolution neural network on the whole image and divides it in to the regions and then identify the bounding box and the object present in it. Later, the authors also proposed the different versions of the yolo.

Lin et.al [17] proposed the network architecture in which they find out the reason of the low accuracy of single stage architecture as compared to the two-stage detector. They found after the research, that the images who have the dense background and foreground create the unbalancing situation during the training. To tackle this problem, a new loss function has been utilized named as the focal loss.

These all-detection algorithms, which we discussed in the literature till now includes the traditional and deep learning approaches utilized for object detection. Since, human is also categorized as an object. So, for the detection of human, different features have been extracted to identify the object as a human. These features include, 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 generic environment and have a limitation to performs only in controlled environment. They have utilized the template matching techniques for person detection [18].

V. Gajjar et.al [19] proposed the human detection algorithm which extracts the histogram of oriented gradient features (HOG) and then trained it using the SVM classifier for human detection purpose. Dalal et al. [20] proposed the algorithm in which they utilized the texture based features by using the Histogram Oriented Gradient (HOG). They have extracted the high dimensional features such as edges and then trained these features using the support vector machine (SVM) for human detection. Some researchers have also done work on the person detection using Face Detection [21] and Gait analysis [22]. Andriluka et al. [23] proposed the methodology in which they detect the human from the partially occluded environment using the tracklet based detector. They are able to achieve the 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 the remarkable accuracy on object detection models. They have achieved almost more than 90 percent accuracy on person detection. Both Faster RCNN and YOLO-v3 [24] although outperforms 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 [25] [26].

Boudjit et.al. [27] used the convolutional neural networks (CNN) YOLO-v2 based on the camera of a drone, this paper presents research progress in the creation of applications for the identification and detection of people. Deep-learning-based computer vision is used to assess the person's location and state. The results of the individual detection indicate that YOLO-v2 can identify and classify objects with a high degree of accuracy from the aerial view.

This [28] 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. 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 till now, no one utilized the amalgamation of deep learning and traditional computer vision approaches for identifying the hotspot zone detection for the prevention of the spread of covid-19. The Yolo-v3 have been fine-tined to gain the results in real time for human

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

detection. The sequence of the paper is as follows; Proposed methodology is described in the section 3. Results and discussion have been discussed in section 4. Whereas, paper is concluded in section 5.

3. Proposed Methodology

Our 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 have directly or indirectly in a contact with the marked ROI. We increase the number in a counter, and if it exceeds to some specific threshold, then we highlight the particular region until the concerned authority is informed and disinfect 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 algorithm 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.

1.1 Human Detection

For human detection, we have utilized the YOLO-v3 object detection with the fine-tuned parameters. Till now, most of the object detection model based on the different steps which was 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 to 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, whole image is divided into the SxS grids. For instance, in fig. 1 the image is divided into the 5x5 grid. However, in all YOLO invariants the grid size is fixed of 7x7. If the centre of any particular object is found to be present in any of the grid, then it is responsibility of the grid to detect the object. Since YOLO applies the 7x7 grid on the image, so we have total 49 cell spaces. YOLO runs classification and localization at the same time on each cell.

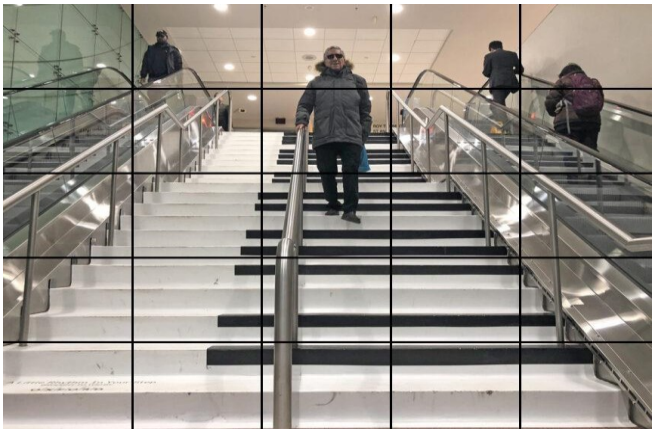

Figure 1: 5x5 grid on image

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. As it detects only one object per cell, so if the cells on the particular images contains more than the one object then the model cannot able to detect it. Another issue which occurs in all YOLO invariants are that the parts of the same objects may appear into the multiple cells. Just like in the figure 1, taxi have been detected in the multiple cells of the grid.

This problem has been sort out using the non max suppression. For each grid, YOLO produce 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 that the particular bounding box whether to contain the object or the background. By doing so, algorithm is able to prevent the detection of the background. Intersection over union is used to identify whether the predicted region of interest contains the object or not. It has been done by comparing the predicted bounding box with the ground truth bounding boxes which is annotated by the human.

3.2 Network Architecture

As, we utilized the YOLO-v3 architecture with the fine-tuned parameters for human detection as shown in the figure 2, so here we discuss the YOLO-v1 and its successor till 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 the objects. It contains the 24 convolution layers along with the 2 fully connected layers. But YOLO-v1 not utilized the inception modules instead of 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 which gives 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. As compared to the DarkNet19 architecture, DarkNet53 contains the residual blocks, which is 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 the problem of 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 have been predicted using the logistic regression. The objectness score is considered to be 1, if the predicted bounding box is overlapped with the ground

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

truth bounding box. Now, if we talk about the non-maximal suppression, which we are used to sort out the problem of if more than one bounding box contains the same object. This problem has been solved using some predefined threshold on the intersection over union (IOU) value. If the IOU value is less than some specific threshold for some bounding boxes, then they are all discarded. The algorithm chooses only those bounding boxes who have the highest confidence value and depicts its prediction.

3.3 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;

$$\lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{b} 1_{ij}^{OBJ} [(X_i, \hat{X_i})^2 + (Y_i, \hat{Y_i})^2] +$$
$$\lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{b} 1_{ij}^{OBJ} [(\sqrt{w_i} - \sqrt{\hat{w_i}})^2 + (\sqrt{H_i} - \sqrt{\hat{H_i}})^2]$$
$$\lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{b} 1_{ij}^{OBJ} (c_i, \hat{c_i})^2 + \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{b} 1_{ij}^{OBJ} (c_i, \hat{c_i})^2 +$$
$$\sum_{i=0}^{s^2} 1_i^{obj} \sum_{c \epsilon classes} (P_i(C) - \hat{P_i}(C)) \quad (1)$$

Here in equation 1, λ 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 is equal to 0.

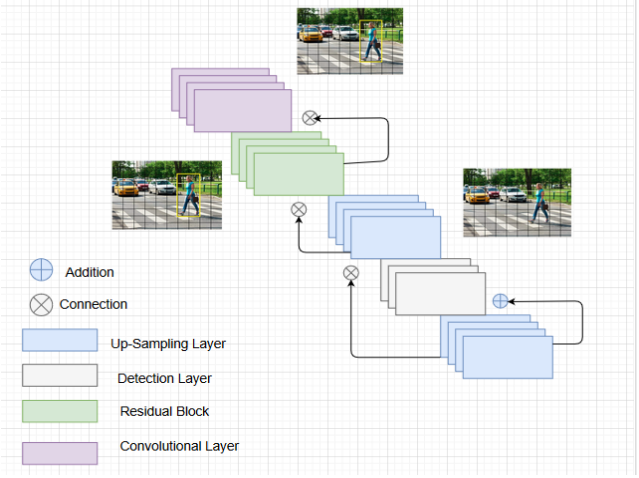

Figure 2: Architecture for human detection

3.4 Deep Sort

Deep sort is the deep learning based algorithm which is used to track the particular objects present in a video [29]. 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 the human detection from the images. This information has later combined with the temporal information to get the predictions of the associated trajectories of the detected objects. Deep sort maintains the track of each detected object which is under our consideration by using the unique identifier in order to performs the further statistical analysis on it.

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

$$y = [x, y, \lambda, h, u^{\wr}, v^{\wr}, \lambda^{\wr}, h^{\wr}]^T \quad (2)$$

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 are exceeding to some specific predefined limit, then tracking from the object have been removed, and thus the same process have been start 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 track the any particular detected object then we keep that object for tracking otherwise it is discarded.

After that Hungarian algorithm have 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;

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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

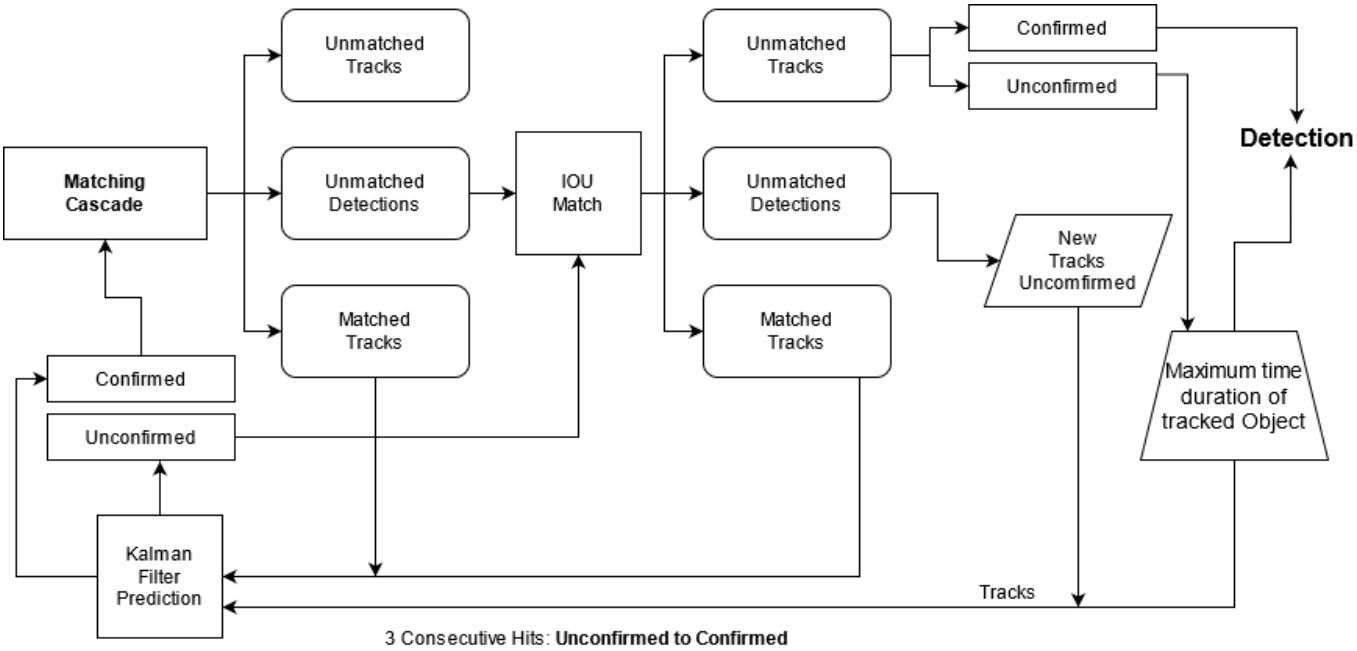

Figure 3: Pseoudo Code for tracking using Deep Sort

In equation 3, x^{th} track projection have 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 the figure 3.

4. Results and Discussions

4.1 Dataset acquisition and Pre-processing

Deep learning-based object detection model requires a huge amount of data, if the training is required from the scratch. So, the possible solution is to fine-tuned some pre-trained model on our dataset to generate the require results [30] in a small amount of data. In this proposed work, object detection model is fine-tuned for human detection purpose using the tensorflow which is 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 [31] dataset, Pascal [32] dataset and open images dataset [33]. 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 in such a way 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 un-necessary 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 rest of the images are in the testing fold. Sample frames have shown in the figure 4.

Figure 4: Sample frames from the dataset

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

4.2 Implementation Details and Evaluations

The training is carried out on the Nvidia-1080 Ti Gpu who have the memory capability of 11 Gb. The training takes almost 6-7 hours for fine-tuning the model. At 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 is calculated by calculating the difference between the actual label value with 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 data, then this means loss value will be high for particular algorithm. The loss value then optimized using the optimizer function and setting out some parameters of convolution neural network architecture. 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 equation have been used to calculate the mean square error.

$$\frac{1}{n}\sum_{N-1}^{n}(\hat{p_n}-p_n)^2 \qquad (4)$$

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

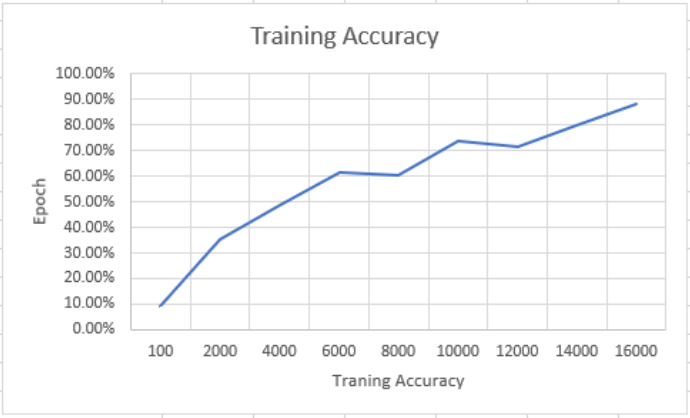

Figure 4: Training accuracy of person detection

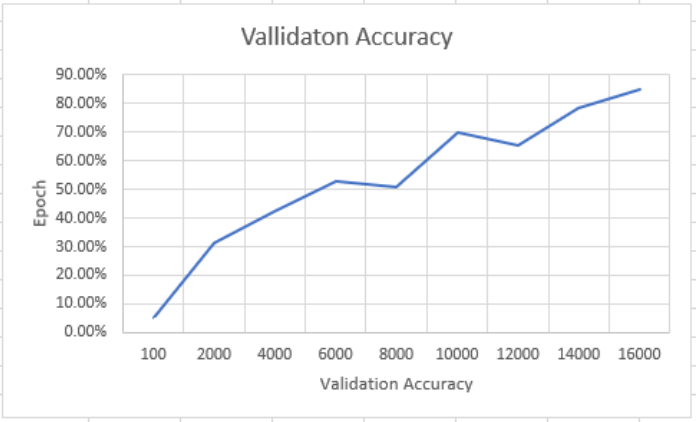

Figure 5: Validation accuracy of person detection

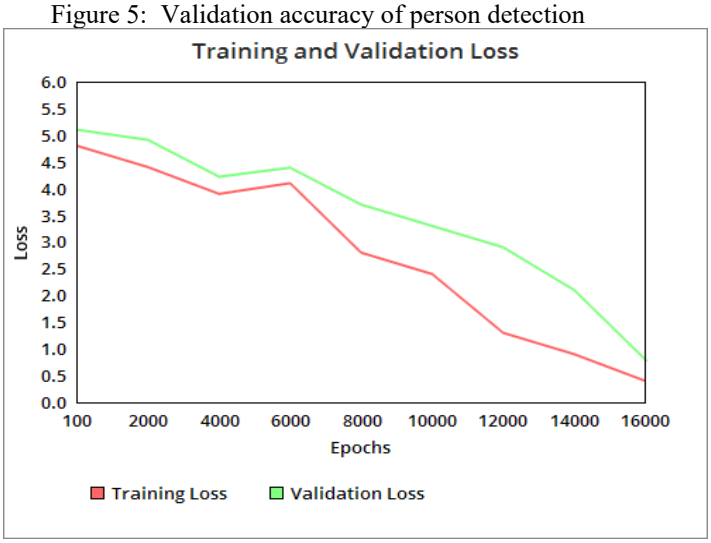

Figure 6: Training and Validation Loss on Person Detection

The model has also been evaluated, by training the same dataset on the other state of art detection architectures. The numeric results have been showed in the table 1.

Figure 4: Hotspot detection using proposed architecture

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3094720, IEEE Access

Table 1: Comparison on different object detection models

Model	mAP	FPS
Faster RCNN [14]	96.12	3
SSD [34]	68.23	10
Fine-tuned YOLO v3	84.81	22

From the table 1, it is clearly observed that although the Faster RCNN achieve 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 real time related to the hotspot detection, so that's why Yolo-v3 is most preferable. Sample images from our proposed problem have shown in the figure 4. The comparison is only made with state of the art object detection algorithms with respect to the human. Because, in our proposed methodology correctly identification of the human detection is the backbone. Whereas, Region of interest at suspicious place is drawn by human or the user. Furthermore, IOU (intersection over union) is calculated by measuring the intersection between the bounding boxes of detected human and human drawn ROI. These both (IOU and User Drawn ROI) is evaluated using the quantitative evaluation. The quantitate evaluation performed to make the experiments robust.

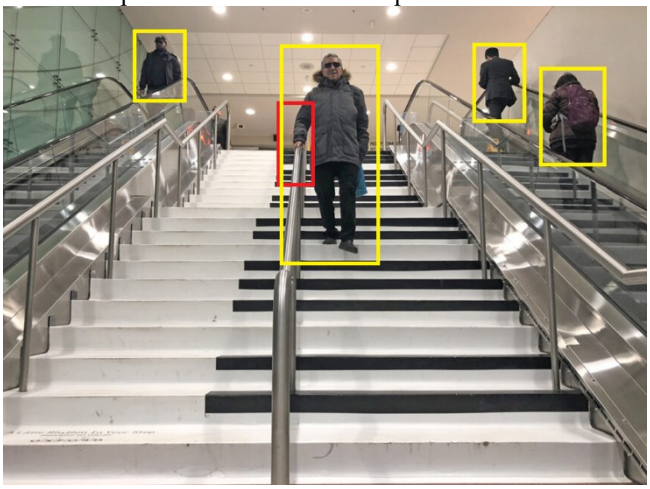

Figure 7. Hotspot detection using proposed architecture

The explanation and working of our proposed work as depicted in the figure 7 is as follows;

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

1. 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.
2. The intersection over union (IOU) is calculated between the user drawn rectangle (Red Rectangle) and the human detected rectangles Yellow Rectangles.
3. If IOU is more than to the specific threshold between red and yellow box, then we considered that human is present in that hotspot zone and considered as violation.
4. We set the counter to count the number of persons over there.
5. Tracking of that particular person using the deep sort to reduce the false positive in the counter.
6. Set the counter threshold of 10, if the number of violation exceeds to that threshold, concerned authority have been notified to take the necessary actions.

For quantitative evaluation, we passed 60 different videos to our proposed system to detect hotspot zone detection and then manually check its perfection. The results of these surveys elaborated in the below table 2. Total 30 videos for both (Hotspot Zone, No Hotspot Zone) were utilized for the quantitative evaluation purpose.

Table 2: Quantitative Evaluation of Hotspot Zone Detection

	Hotspot Zone	No Hotspot Zone
Hotspot Zone	26	4
No Hotspot Zone	7	23

Table 2 depicts the confusion matrix of hotspot zone detection. Each video has the length of 30 seconds. From, 30 videos of hotspot zone, our system correctly identify hotspot zone from 26 videos. This means that human is detected perfectly near to our drawn suspicious region and based on the intersection over union (IOU) value, we are able to identify that danger zone. Whereas, in four videos system are unable to figure out the zone, because of not able to detect of the human near to our specified region. Similarly, for no hotspot zone we have drawn the (Region of Interest) ROI far from the human pathway. There is a 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 81.7% accuracy on this quantitative evaluation.

5. Conclusion

This paper proposed the novel framework for identifying the hotspot zone detection using the state-of-the-art deep learning approaches. This task has been done by identifying the areas under consideration, and then find out the area of intersection 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 to 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 the 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

We would also like to express our earnest gratitude to National Centre of Artificial Intelligence Pakistan Fund and organization (KICS) for supporting this research work. The authors would also like to acknowledge the support of Artificial Intelligence & Data Analytics Lab (AIDA) CCIS Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

REFERENCES

1. 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 Access at: 3rd September 2020
2. Krizhevsky A, Sutskever I, Hinton GE. Image-net classification with deep convolutional neural networks. In Advances in neural information processing systems 2012 (pp. 1097-1105).
3. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L. Large-scale video classification with convolutional neural networks. InProceedings of the IEEE conference on Computer Vision and Pattern Recognition 2014 (pp. 1725-1732).
4. Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. InAdvances in neural information processing systems 2015 (pp. 91-99).
5. He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. InProceedings of the IEEE international conference on computer vision 2017 (pp. 2961-2969).
6. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. InAdvances in neural information processing systems 2014 (pp. 2672-2680).
7. Huang, Thomas. "Computer vision: Evolution and promise." (1996).
8. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001 2001 Dec 8 (Vol. 1, pp. I-I). IEEE.
9. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1. IEEE, 2005, pp. 886–893.
10. P. Felzenszwalb, D. McAllester, and D. Ramanan, “A discriminatively trained, multiscale, deformable part model,” in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008, pp. 1–8.
11. R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 580-587, Jun. 2014.
12. K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in European conference on computer vision. Springer, 2014, pp. 346–361.
13. R. Girshick, "Fast R-CNN", *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1440-1448, Dec. 2015.
14. S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks", *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, pp. 91-99, 2015.
15. T.-Y. Lin, P. Doll´ar, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection.” in CVPR, vol. 1, no. 2, 2017, p. 4.
16. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
17. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Doll´ar, “Focal loss for dense object detection,” IEEE transactions on pattern analysis and machine intelligence, 2018.
18. M. Singh, A. Basu, and M. K. Mandal, “Human activity recognition based on silhouette directionality,” IEEE transactions on circuits and systems for video technology, vol. 18, no. 9, pp. 1280–1292, 2008.
19. V. Gajjar, Ayesha. G, & Yash. K, Human detection and tracking for video surveillance: A cognitive science approach, In *Proceedings of the IEEE International Conference on Computer Vision Workshops,* pp. 2805-2809, 2017
20. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05), vol. 1. IEEE, 2005, pp. 886–893

21. P. Huang, A. Hilton, and J. Starck, "Shape similarity for 3d video sequences of people," International Journal of Computer Vision, vol. 89, no. 2-3, pp. 362–381, 2010.

22. D. Cunado, M. S. Nixon, and J. N. Carter, "Using gait as a biometric, via phase-weighted magnitude spectra," in International Conference on Audio-and Video-Based Biometric Person Authentication. Springer, 1997, pp. 93–102.

23. M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in 2008 IEEE Conference on computer vision and pattern recognition. IEEE, 2008, pp. 1–8.

24. Redmon J, Farhadi A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767. 2018 Apr 8.

25. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft coco: Common objects in context. InEuropean conference on computer vision 2014 Sep 6 (pp. 740-755). Springer, Cham.

26. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (voc) challenge. International journal of computer vision. 2010 Jun;88(2):303-38.

27. Boudjit, K., & Ramzan, N. (2021). Human detection based on deep learning YOLO-v2 for real-time UAV applications. Journal of Experimental & Theoretical Artificial Intelligence, 1-18.

28. Wei, H., & Kehtarnavaz, N. (2019). Semi-supervised faster RCNN-based person detection and load classification for far field video surveillance. Machine Learning and Knowledge Extraction, 1(3), 756-767.

29. N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in 2017 IEEE international conference on image processing (ICIP). IEEE, 2017, pp. 3645–3649.

30. Khan MZ, Harous S, Hassan SU, Khan MU, Iqbal R, Mumtaz S. Deep unified model for face recognition based on convolution neural network and edge computing. IEEE Access. 2019 May 23;7:72622-33.

31. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft coco: Common objects in context. In European conference on computer vision 2014 Sep 6 (pp. 740-755). Springer, Cham.

32. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (voc) challenge. International journal of computer vision. 2010 Jun;88(2):303-38.

33. Kuznetsova A, Rom H, Alldrin N, Uijlings J, Krasin I, Pont-Tuset J, Kamali S, Popov S, Malloci M, Kolesnikov A, Duerig T. The open images dataset v4. International Journal of Computer Vision. 2020 Mar 13:1-26.

34. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC. Ssd: Single shot multibox detector. InEuropean conference on computer vision 2016 Oct 8 (pp. 21-37). Springer, Cham.