IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 March 2023, accepted 23 April 2023, date of publication 2 May 2023, date of current version 12 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3272481

RESEARCH ARTICLE

Realtime Crowd Monitoring–Estimating Count, Speed and Direction of People Using Hybridized YOLOv4

MUHAMMAD HARIS KAKA KHEL[1], KUSHSAIRY ABDUL KADIR[1], (Senior Member, IEEE), SHEROZ KHAN[2], (Senior Member, IEEE), MNMM NOOR[3], HAIDAWATI NASIR[3], (Senior Member, IEEE), NAWAF WAQAS[4], AND AKBAR KHAN[1]

[1]Electrical Section, Universiti Kuala Lumpur–British Malaysian Institute, Gombak, Selangor 53100, Malaysia
[2]Department of Electrical Engineering, College of Engineering and Information Technology, Onaizah Colleges, Onaizah 51911, Saudi Arabia
[3]Computer Engineering Section, Universiti Kuala Lumpur–Malaysian Institute of Information Technology, Kuala Lumpur 50250, Malaysia
[4]Department of Instrumentation and Control Engineering, Universiti Kuala Lumpur–Malaysian Institute of Industrial Technology, Kuala Lumpur 81750, Malaysia

Corresponding authors: Kushsairy Abdul Kadir (kushsairy@unikl.edu.my) and Muhammad Haris Kaka Khel (muhammad.haris@s.unikl.edu.my)

This work was supported by the Deputyship for Research and Innovation, Ministry of Education, Saudi Arabia, through the project titled "Intelligent Real-Time Crowd Monitoring System Using Unmanned Aerial Vehicle (UAV) Video and Global Positioning Systems (GPS) Data" by the Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia, under Grant QURDO001.

ABSTRACT Researchers are becoming more interested in crowd surveillance because of its several potential applications. These applications may include detecting unusual activity for security purposes, monitoring reasons for archiving records, and conducting inventory for facility planning and extension. Detecting people and tracking them from a security viewpoint and understanding their behavior in places large crowds is highly important because unruly crowds in public spaces can lead to serious health and security concerns. Crowd related accidents happen to cause injuries and deaths, which often occur during events not properly planned. The planning of the organizers relies heavily on exploring the behavior of the few in a crowd of individuals and groups in thousands that create the crowds. It is this focus that provides the main reason for this research. This work proposes a model that can count people in crowds, automatically detect and track people, and then estimate their direction and speed. Deep learning networks have proven costly to run, needing memory and power to perform computations beyond what is possible on edge devices with limited resources. As a result, we propose the use of hybrid YOLOv4 consisting of detection method combined with the training phase pruning and the use the convolution attention module strategy. Accuracy of the Hybrid YOLOv4 is increased by 33%, whereas mAP reached 92.1%. While training on the JHU dataset, the suggested hybrid YOLOv4 strategy decreases the computational memory requirements, all of which closely meet the real-time application conditions. This work will help avoid the threatening situation of crowding gathering around to cause stampedes and thus risking crowds with disastrous consequences.

INDEX TERMS Direction estimation, speed estimation, hybrid YOLOv4, crowd monitoring, crowd management, deep learning.

I. INTRODUCTION

Crowd is a specific group of individuals who have gathered together in an unbridled way in a society or a community. The components of a crowd are made up of distinct participants, each of whom performs moves characteristic of their own goals and behavioral inclinations [1]. Crowds gather together in public places such as airports, retail centers, colleges, clinics, railway and bus stations, sport stadiums and parks. Furthermore, people congregate at religious buildings such

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaojie Su.

as mosques and temples [2]. Crowd status has been used for applications like finding size, grading, and from block-chain to record transactions that are replicated for monitoring and to be shared and transmitted for the benefit of participants. Human detection can be crucial from a security point of view at religious places and public areas as they have frequently resulted in serious security and safety problems [3]. Poorly organised and planned events with large audience can lead to injuries with fatalities to becoming catastrophic, for example, the Indonesian football match of 01-10-2022. The core component of planning a large event for the organizers involves a study of the crowd size, the type of cultural diversity, and the behavior of individual participants. Hajj is considered to be one of the biggest gatherings of about 20 million visitors of diverse ethnicity and geological backgrounds from all parts of the world once every each year in the holy city of Markka. with 2-3 million pilgrims congregating at one spot at a time during some moment in time, while crowds in thousands during Umrah throughout the year [4]. As a result, public safety is at risks any time in such large, congested spaces. In the Kaaba, Mount Arafat, Muzdalifah, and Mina, the Hajj is performed by crowds moving in very close proximity. When dealing with such massive crowds of people, the authorities' primary concern is the safety of the pilgrims by ensuring that stampedes are avoided as the most common cause of accidents as a result of overcrowding.

Crowd monitoring helps in understanding to estimate crowd dynamics, which can help improve event management and public safety. Researchers are interested in the vast range of human detection implementations, such as the detection of anomalous events, crowd monitoring for counting of people, pedestrian recognition, movement evaluation, and detection of falls [5], [6]. Deep neural networks have several benefits, such as automatic feature collection and obtaining the most crucial information and advanced characteristics with a higher degree of classification probability. The two main categories of deep learning methods for problems involving object detection, are one-stage and two-stage detection algorithms. The one stage detection approaches transform the identification issue into a single regression level such as YOLO [7] and SSD [8]. One-stage techniques are faster than the two-stage strategies because of the inherit light structure. In a two-stage detector, such as Faster R-CNN, one model extracts object patches, and a second model classifies and fine-tunes the location of objects and adjust them [9]. The two-stage approach is considered somewhat slow, but very effective [10].

Furthermore, in challenging real-world situations, evaluating human detection systems are more difficult, as heavily crowded places call for a lot of computing power. Different people occlude with each other, resulting in significant overlaps in crowded circumstances, making crowd occlusion extremely difficult. For example, if a target pedestrian X is heavily overlapping with other pedestrians, the detection mechanism may struggle to distinguish between them due to their similar appearances. YOLOv4 [11] as an evolved edition in the YOLO family, employs a variety of distinctive tactics to dramatically enhance the speed and accuracy of the results. To accomplish accuracy and speed, a significant amount of computing power and memory space are required. However, due to its extensive calculations and poor performance, in such cases, YOLOv4 cannot meet the stringent real-time requirement. Operating YOLOv4 is tricky on devices with limited computing energy. YOLOv4 is regarded as being awful at detecting huge crowds since it struggles to estimate accurate crowds' sizes. Due to the aforementioned factors, this work modifies the YOLOv4 algorithm to make it capable of estimating individuals in a huge crowd while using a relatively low-processing power equipment.

Our goal is to develop model that can detect and track people in the crowd and then estimate their direction and speed for which we used a diverse dataset named JHU with crowd pictures to train YOLOv4. In order to operate it on low-cost computer devices, the pruning strategy is then employed, which makes our model fast and efficient without the need of additional retraining requirement. Then, we have employed the Convolutional block attention module (CBAM) to enhance the weights of beneficial features while decreasing the weights of invalid features to increase the accuracy of the model. Our algorithm tracks and detects individuals to count them afterwards for calculating their direction and speed - all done with accuracy and time intervals that seems suitable for real-time applications.

II. RELATED WORK

The authors in [12] have used global occlusion reasoning along with a prior local scale to identify people in crowds. Utilizing deep network characteristics, the authors in [13] have used recurrent architecture to sequentially identify and count people. Various studies, including [14], instead of a density map, suggest models that directly regresses to estimate the number of people. However, it has been shown that these approaches perform poorly due to the loss suffering from losing spatial information. Ali et al. [15] utilize binary spring search (BSS) to successively fix the flaws of earlier networks. In [16], a boosted decision forest is utilized to identify pedestrians, while deep neural networks are used to extract self-learned properties. Cai et al. in [17] provide a method that combines various degrees of characteristics to detect humans at varying scales.

Using images from depth sensing camera, the human tracking and detection systems have been presented by Wu et al. [18] and Wetzel et al. [19]. Hough circle was researched by Van Oosterhout, Bakkes, and Krose [20], while facts about the human head's and whorl shaped hairs have been studied by Nakatani et al. [21]. Color information has been employed by Wateosot et al. in [22], Nakatani et al. in [21], and Gao et al. in [23], while edge information has been used by others, such as Sobel filter or Canny edge detector [24]. Textual information, such as information bout hair texture, is also used by [21]. There are several regions of interest

(RoI) reported in the established approaches; for instance, some researchers thought the human head area was the target RoI [25], while others [26] believed that the RoI for human head detection has been based on the head-shoulder information. Ahmed and Adnan [27] demonstrated a reliable detection method for people in a closed area. This was considered a better technique as it has made use of bounding boxes of varying sizes and orientations. Watada et al. [28] employed a feature-based approach with HOG features for human count and identification. Punn et al. in [29] proposed a system that used the YOLOv3 and DeepSort algorithms to detect and track individuals. For locating and following people in crowds, a top-view model has been presented by Ahmad et al. in [30]. Li et al. in [31] introduced the use of AdaBoost to successfully integrate hundreds of weak, computationally less-costly classifiers for human detection. Many researchers tracked and detected people using the fisheye camera with a field of vision covering up to 180° and deep° learning techniques. The authors in [32] pyramidal Lucas- Kanade optical flow method to estimate the direction of person in which they calculate the displacement between the pixels. However, objects closer to the camera will have larger pixel displacement than the objects far from the camera although both objects are moving at the same speed in the real world. Thus, it is very difficult to find the accurate speed of pedestrians using the pixel displacement methods.

The authors in [33] provide a deep-fusion model that effectively and efficiently makes use of the nested features which are extracted from deep neural networks with varied convolution layers. They provide a network that integrates multi-scale data from shallow to deep levels of the network and maps the density map's input image. In order to recover missed detection, the authors in [34] suggested a motion-guided filter (MGF) that makes use of spatial and temporal information present in the video's subsequent frames. For crowd counting in low-density to medium-density movies, the method used is based on the deep convolution neural network (DCNN), hybridised with pruning and convolutional block attention module making it more suitable for head detection in crowded situations [35]. The authors in [36] used a scale-driven convolutional neural network (SD-CNN) model to detect the heads of the people in the crowd. The authors in [37] suggest two networks: the dense-scale convolutional neural network (DS-CNN) and the sparse-scale convolutional neural network (SS-CNN) (DS-CNN). With rough knowledge of the scales in the image, SS-CNN can distinguish human heads. The dense scale map produced by DS-CNN is created by globally rationalizing the coarse scales of detection acquired by SS-CNN using Markov Random Field (MRF). However, all these works only focus on the counting in crowd monitoring and ignore direction and speed of persons in the crowd which are important aspects of word monitoring.

It is well accepted that attention has a substantial influence on how people see things. The fact that the one of its most essential features of human visual system is that it does not attempt to comprehend a whole image at once. Instead, people actively focus on key areas of the image and take a succession of constituting peeks to better understand the visual composition. A network of residual attention suggested by Liang et al. [38] uses an attention module in its encoder-decoder method. The network functions effectively and is resilient to noisy inputs as a result of the refining of the map's features. Hu et al. [39] have considered developing a compact module that makes use of inter-channel interaction. In their Squeeze-and-Excitation module, they employ global average-pooled features to calculate channel-wise attention accordingly.

There are several pruning approaches that can also be classified into pruning with fine grain and pruning with coarse grain. Han et al. [40] have proposed trimming using parameters' method. It determines whether the parameters exceed a certain threshold value before evaluating the importance of the weights in focus. The compressed model has then been efficiently run inside the internal SRAM. On VGG-16 [41], the model obtains a 13-fold compression ratio while maintaining accuracy. The advantage of fine-grained pruning is that model accuracy is preserved. On the other side, the localized fine-grained pruning needs specialized hardware and run-times. Molchanov et al. in [42] have presented a pruning method at the filter level. The subset of loss is computed once a subset of filters is selected. Li et al. in [43] have developed an effective and efficient method for convolution neural network run-time acceleration. To minimize the calculation time without causing irregular sparsity, the filter is reduced. Li et al. in [44] have suggested a channel-level dynamic pruning algorithm. The normalization factor of the Batch-Normalization (BN) layer has been used to estimate the relevance of channels, while the regularization training has made the channel relatively sparse.

III. METHODOLOGY

We have focused our efforts in this paper to estimate count of individuals along with direction and speed, which plays a crucial role in crowd surveillance. The Hybrid YOLOv4 is first trained using the JHU dataset [45], which includes a variety of crowd images. During training, we have applied pruning technique in which we prune the model in every epoch until we reach to our desired pruned threshold value. We have employed L1 normalization-based pruning, a well-known pruning technique to eliminate unnecessary weights. Only the convolutional layer filters in the backbone of Hybrid YOLOv4 which are used for features extraction, have been pruned. The Feature Fusion Network of YOLOv4 has then been used to process these weights.

Convolutional Block Attention Module (CBAM) modules have been added to each of the three branches at the end of the feature fusion network to enable the neural network to pay more attention to the target area containing important details, suppressing thus the unimportant details, and to improve target detection accuracy as a whole. After that, we have used DeepSort tracker to process these detected information and

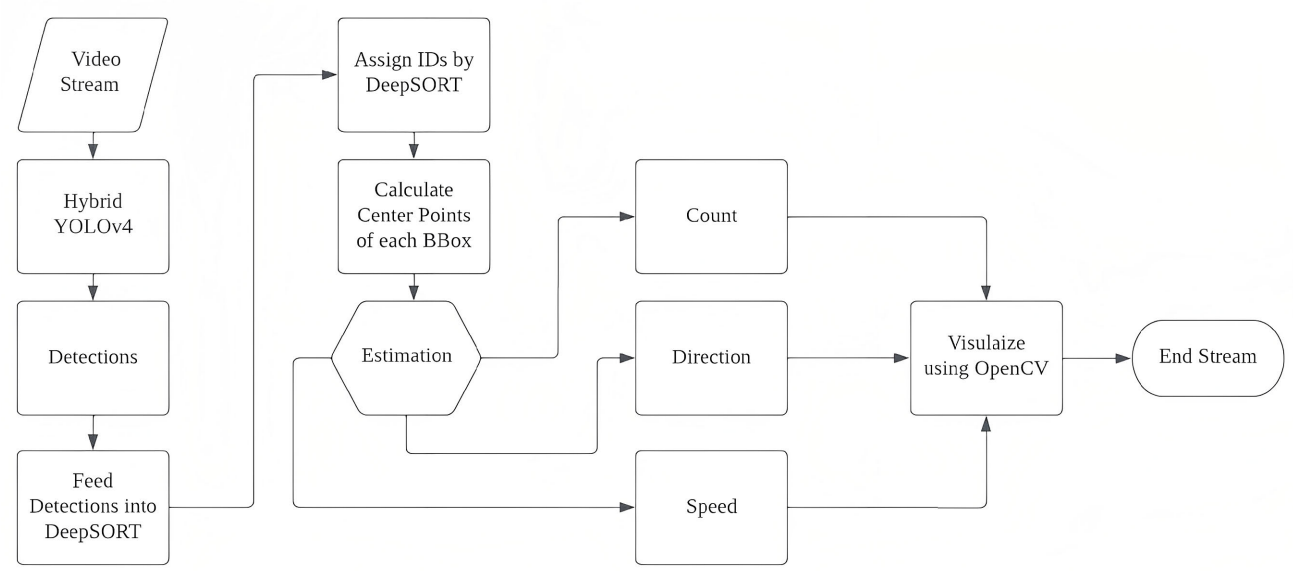

FIGURE 1. The proposed approach's model state diagram.

assigned every detected bounding box a unique ID. It also regularly checks for detection of new person and discards the ID of the bounding box which is no longer in the frame. Furthermore, the counter is used to count the bounding boxes in the frame and in total to produce the count and the center coordinates of the bounding box are then calculated for use in the calculation of speed and direction. Fig. 1 illustrates the overall structure of the proposed hybrid-YOLOv4.

IV. ORIGINAL YOLOv4 ARCHITECTURE

YOLOv4 introduces PANet (path aggregation network), Mish activation function, spatial pyramid pooling (SPP), mosaic data augmentation, self-adversarial training, and many more approaches to increase detection accuracy. The backbone of YOLOv4 as shown in Fig. 2, consists of CSPDarknet-53, which integrates the Cross Stage Partial Network (CSPNet), which shortens the calculation time while maintaining the accuracy due to which better speed and accuracy can be achieved. The CSPDarknet-53 is able to extract in-depth features from the input image using Resblock cells. The network has 53 convolution layers, each with a size ranging from 1×1 to 3×3, each of which is connected to a Batch_Normalization (BN) layer and a Mish activation layer. Additionally, all activation functions in YOLOv4 have been switched off for leaky-ReLU, which requires less processing.

Fig. 2 displays how the input image is forwarded to the backbone network to extract features followed by SPP and PANet to combine feature maps, and then output these feature maps onto three scales generating the bounding box, category, and counting. The head of YOLOv4 is consistent with YOLOv3. The residual module considerably decreases network parameters, and addresses the gradient disappearance issue brought on by the continuous network deepening, and

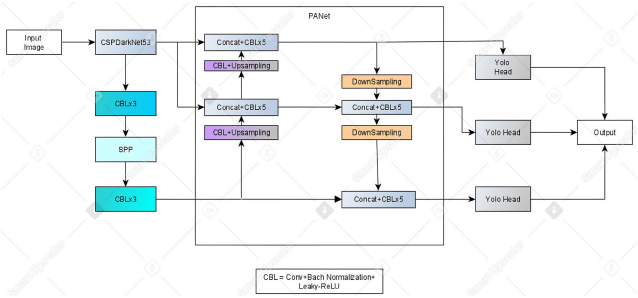

FIGURE 2. Architecture of YOLOv4.

facilitates training of deeper convolutional neural networks. The purpose of the SPP module is to allow the input of the convolutional neural network to get input of any size, allowing it to enlarge the responsive field and efficiently separate the important features of elements without reducing speed of the processing model. The SPP module is located after the CSPDarknet-53 feature extraction network. After the top-down feature pyramid that has two PAN structures, the PANet adds a bottom-up path augmentation structure. The PANet can make full use of shallow features, as well as feature fusion of multiple backbone layers for different detector levels, to improve feature extraction capabilities and detection performance. Then, the head of YOLOv4 forecasts the position of the bounding box and its corresponding category.

V. HYBRID YOLOv4 ALGORITHM

The large amount of gradient data produced by the network-deepening procedure makes the network very slow, complex and costly, therefore, we have tried to solve this

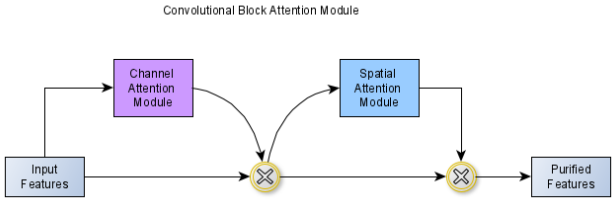

FIGURE 3. Layout of convolutional block module.

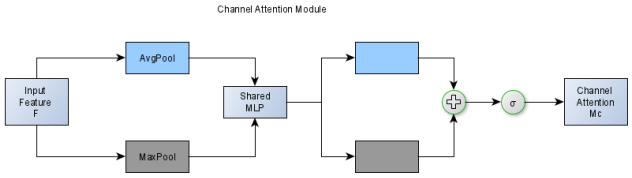

FIGURE 4. Channel attention section's structure.

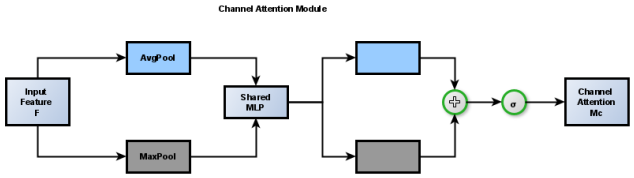

FIGURE 5. Composition of the spatial attention module.

problem by introducing the L1 Normalization based pruning during training to reduce the computation, redundant features and complexity of the network which improves the inference speed without adversely affecting the accuracy. Secondly, to solve the problem of detecting small objects and improve the overall accuracy of the model we have introduced the convolutional block attention module (CBAM) to the architecture of YOLOv4. which enhances the overall accuracy of target detection by making the neural network more attentive to the region holding the target person's valuable information. It also removes the unimportant data by directing the attention of neural network to a specific area holding crucial the data.

A. L1 NORMALIZATION BASED PRUNING

Pruning based on L1 Normalization [40] is a method of removing filters based on their L1 normalization, which is calculated using equation (1):

$$\text{L1\_norm} = \sum_{k=1}^{n} abs\,(wk) \tag{1}$$

The is the absolute value of the k^{th} filter weight, and there are $'n'$ total filter weights. The role of the filter is assessed using the filter magnitude. Low-magnitude filters are removed via pruning, as they do not contribute substantially to the network.

B. CONVOLUTIONAL BLOCK ATTENTION MODULE (CBAM)

CBAM is a small but highly effective attention module supported by almost all Convolutional Neural Network, which can be trained in parallel with CNN. Fig. 3 depicts the CBAM module's design.

The CBAM module is made up of the channel attention and spatial attention modules. First, the input feature map enters into the channel portion of the attention module so that it can generate the attention map. Next, it convolves the input feature map with the attention map and generate channel attention map. This channel attention map is then passed through the spatial attention module. The spatial attention module generates attention map and convolves it with the channel attention map to generate the final feature map. The final output feature map contains mathematical equation (2):

$$\begin{aligned} F' &= F \otimes M_c(F) \\ F'' &= F' \otimes M_S\left(F'\right) \end{aligned} \tag{2}$$

where $\otimes$ stands for factor-by-factor multiplication, and F is the input feature map, $F'$is the attention map and $M_c(F)$ channel attention map produced by the channel attention module, where, $M_S\left(F'\right)$ the spatial attention map produced by the spatial attention module, and F'' is the feature map produced by the CBAM.

1) CHANNEL ATTENTION MODULE

Channel attention is used to focus on the most important features as shown in Fig. 4.

The input feature map (F) is first exposed to global maximum pooling and global average pooling, after which a multi-layer perceptron (MLP) is used to combine the weights. There is a hidden layer in the MLP that is comparable to the two fully connected layers. The channel attention map is created by adding pixel-by-pixel the two MLP outputs together and applying the sigmoid activation function as described by equation (3).

$$\begin{aligned} M_c(F) &= \sigma(\text{MLP}(\text{AvgPool}(F)) + \text{MLP}(\text{MaxPool}(F))) \\ &= \sigma\left(w_1\left(w_0\left(F^c_{\text{avg}}\right)\right) + w_1\left(w_0\left(F^c_{\text{max}}\right)\right)\right) \end{aligned} \tag{3}$$

where σ represents sigmoid activation function. w_0 and w_1 are the weights of the MLP.

2) SPATIAL ATTENTION MODULE

The spatial attention module is utilized after the channel attention module to focus on where the crucially important features arise. Fig. 5 shows how the spatial attention module is structured.

The module of spatial attention take F' as an input feature map and, employing global average pooling and global maximum pooling on a channel-based basis, the two feature maps F^s_{avg} and F^s_{max} are combined to generate a channel number of two feature maps, then applies 7×7 convolutional layer to reduce the channel number to 1, and finally uses a Sigmoid activation function to produce a spatial attention map $M_s(F')$

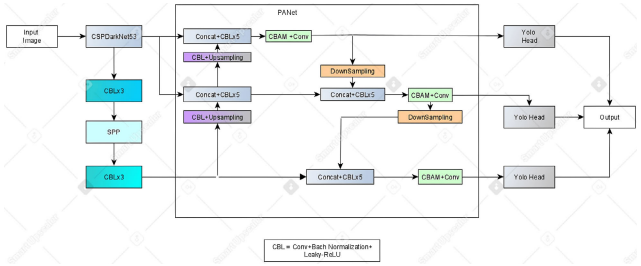

FIGURE 6. Hybrid YOLOv4 Architecture.

as shown in equation (4):

$$M_s(F') = \sigma\left(f^{7\times7}([\text{Avg Pool}(F'); \text{MaxPool}(F')])\right) = \sigma\left(f^{7\times7}\left(\left[F^s_{\text{avg}}; F^s_{\text{max}}\right]\right)\right) \quad (4)$$

After implementing these enhancements to the YOLOv4 architecture, the hybridised YOLOv4 architecture is as illustrated in Fig. 6.

VI. HYBRID YOLOv4 TRAINING

A. PREPARING DATASET

JHU-CROWD [45] is a crowd-sourced dataset that includes 1.11 million annotations on 4, 250 photos. This dataset has been collected in a variety of environments and circumstances. The collection is the ultimate dataset to work with because it includes a variety of photos with climate-related degradation and light changes, along with lots of distracting images. The collection also includes a wider variety of annotations, including dots, bounding boxes, blur levels, etc. From this dataset, we choose 1000 photos that are more related to our task. Among which, 70% are used for training, 20% for validating, and 10% for testing purposes.

B. PRUNING WHILE TRAINING

Before the model can be trained, its hyper-parameters must be setup. To hasten the meeting of the network and avoid over-fitting, the model continually updates the parameters throughout training. The training conducted in this paper are with a images having input size of 416 × 416x3, a batch size of 16, and trained until 85 epochs. Tab. 1. summarizes the hyper-parameters.

During the training phase, pruning procedure is employed by performing forward and backward network passes for every epoch to update the weights. Then, using the L1 normalization, all convolutional layers filters are pruned, if the current percentage of pruned filters is less than the necessary pruning rate we repeat the epoch again. We moved on to the next epoch when the necessary pruning rate is achieved during the epoch. We pruned 0.7 % of filters every epoch over the course of 85 epochs in order to get the desired pruning rate of 60 % through (60/0.7) = 85.71 epochs.

VII. OBJECT DETECTION AND LOCALIZATION

The Neck network of the Hybrid YOLOv4 is mainly composed of the SPP, PANet and CBAM, which is also called

TABLE 1. Hyperparameters used during training.

Parameters	Value
Batch_Size	16
Epochs	85
Learning Rate	0.001
Image size	416×416
Optimizer	Adam

feature fusion network as shown in Fig. 6. To enhance the model's capacity to identify objects, the retrieved features must be combined using the feature fusion network. The SPP module has successfully expanded the acceptance range of backbone features, separating the most relevant contextual features significantly. PANet has developed an architecture that allows layer information to be propagated more effectively from bottom to top or top to bottom. After that we have used CBAM module, which applies channel attention and spatial attention. During the channel attention, the dimensionality is reduced, and then the dimensionality is increased, and the channel attention map is obtained using the Sigmoid activation function.

Finally, the input feature map and the attention map are multiplied to produce an output feature map that is then used as an input to Spatial attention module. Channel-based global maximum pooling and global average pooling are then used to activate the spatial attention module. Two feature maps are created, and the number of channels of the two feature maps are combined to produce a feature map, which is then reduced to one channel using a 7×7 convolution. Finally, the Sigmoid activation function is used to generate a spatial attention map, which is then multiplied by the input of the spatial attention module to produce an output feature map. So the unnecessary weights are suppressed during this process, and the useful weights are passed to the head portion of the YOLOv4 architecture where prediction takes place.

The head of Hybrid YOLOv4 generates predictions based on the attributes that have been received from the model. The three efficient feature layers in the prediction network, are: 13-×-13-×-18, 26 × 26 × 18, and 52 × 52 × 18, which correspond to large, medium, and small objects, respectively. Here, the number 18 is obtained as the product of 3 and 6, and 6 is the sum of 4, 1, and 1, where 4 represent the 4 coordinates of the bounding box, 1 denotes if the box contains the target objects, and 1 denotes that there is one category of person detection. Fig. 7 displays the prediction of YOLOv4 bounding box.

Fig. 7 shows (c_x, c_y) the coordinates of the target center point in the upper left corner of the grid cell, (p_w, p_h) the width and height of the priori box, (b_w, b_h) the width and height of the actual prediction box, and $\sigma(t_x), \sigma(t_y)$ the offset value predicted by the convolutional neural network. (5)-(8) calculate the bounding box's position information, where (t_w, t_h) is also predicted by the neural network and (b_x, b_y) are the coordinates of the actual prediction box's center point. The length and width of each grid cell in the

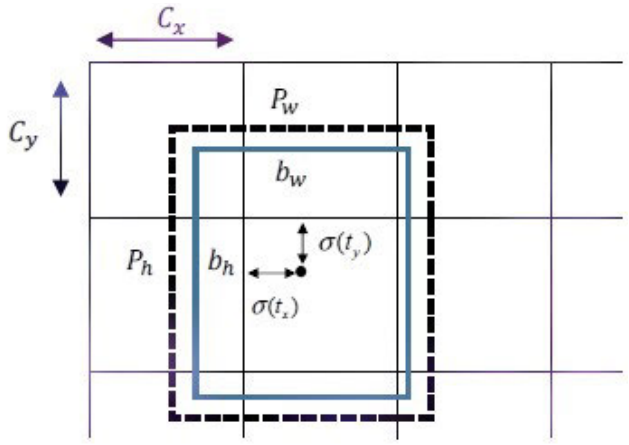

FIGURE 7. YOLOv4 bounding box prediction.

derived feature map are 1, thus $(c_x, c_y) = (1, 1)$ in Fig. 7, and the sigmoid function is employed to keep the predicted offset between 0 and 1.

$$b_x = \sigma\,(t_x) + c_x \tag{5}$$

$$b_y = \sigma\,(t_y) + c_y \tag{6}$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h} \tag{7}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{8}$$

VIII. TRACKING OF PERSON

After the person is detected, the tracking process is carried out. In the tracking process, each person detected is assigned a unique ID that remains the same until the end of the video-stream. Deep SORT is extremely helpful and quick in allowing for high number of frames per second. We have employed Deep SORT technique to track objects because of its greater accuracy and speed. For each bounding box, Deep SORT produces a 128-dimensional vector that should capture important features of the bounding box. Because of its resilience, the use of deep features permits for greater tracking and a real-world application of this technique and the ability to track IDs of intersecting objects or objects are in close very proximity.

While tracking provides an object with a unique ID, we have assigned an ID specific color to the person that has been tracked. We have added a trail of these people as they roam about in the video stream, and this trail has its own color. To improve the accuracy of tracking and their true location, this trail lane is appended around the lower side of these bounding boxes. This trail realises a movement pattern that will be visualized in the map later.

IX. COUNTING

The Hybrid Yolov4 detects every person in the image and videos and then creates bounding boxes around every detected people. These bounding boxes then forwarded to the tracker where tracker assigned them a unique IDs to keep it in the list. After that the counter is called once the individuals have been located, tracked, and bounding boxes around them have been raised. As every bounding box has unique ID so the list keeps updated with every new detection that has been found. The counter then performs two types of counting. It performs counting in the current frame and also overall counting of the number of people being detected so far. The structure of counting is shown in Fig. 8.

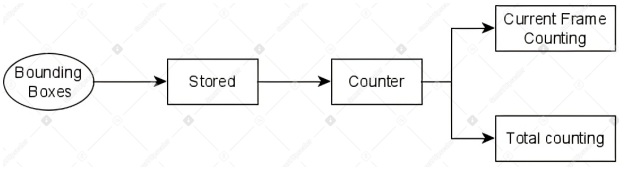

FIGURE 8. Structure of counting.

X. DIRECTION AND SPEED ESTIMATION

To estimate direction, first, we must convert the image into an x-y coordinate system. After that we determine the center points (x, y) of each bounding box using the bounding box co-ordinates. We use the upper left coordinate and the lower right coordinates of the bounding box to compute the center of the box using the equation (9).

$$(x, y) = \left(\frac{x1 + x2}{2}, \frac{y1 + y2}{2}\right) \tag{9}$$

The x coordinate of the center can be computed from the average value $(\frac{x1+x2}{2})$ and accordingly the y coordinate of the center by a similar average calculation $(\frac{y1+y2}{2})$. The tracker will save the x and y coordinates of the center. The same procedure is repeated for every subsequent frame in the stream. Now, we check the x and y coordinates by comparing the center point of the bounding box of the current frame with the center point of the bounding square of the previous frame.

If the x coordinate has increased from the previous value while the y-coordinate remains unchanged, we estimate the person is moving towards right direction. If the x coordinate is decreasing and y remains unchanged, we estimate the person is moving towards left direction. Now, if y axis is increasing and x remain unaltered, we estimate that person is moving in up direction, which means moving away from the source. Alternatively, if it is decreasing, we estimate the person is coming down it means person is coming towards the source. Fig. 9 shows the detection procedure as well as how right and down movements are estimated.

We have used the same center coordinates of the bounding box to estimate the person's speed, but this time we need to use the distance formula to estimate the separation between the two centers of consecutive frames. After calculating the distance between two points, we applied the equation (10) and

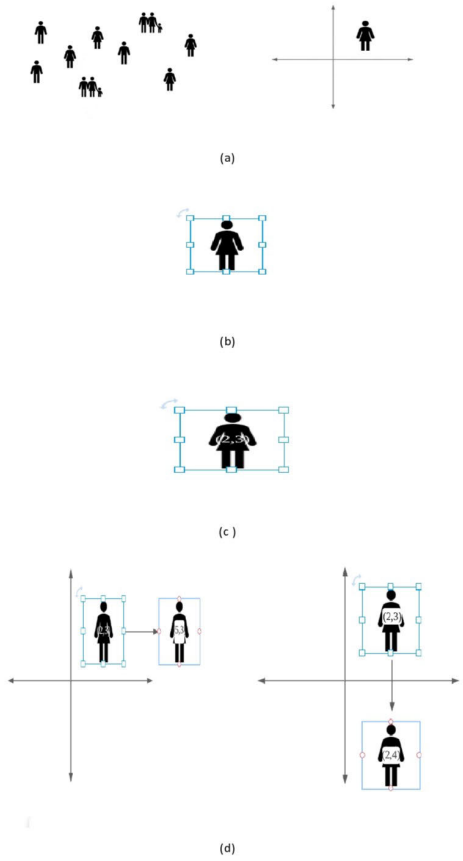

FIGURE 9. Movement detection procedure (a) Conversion of image into Cartesian plane, (b) Detection of people in the image, (c) Finding the center coordinates of the detected people (d) Estimation of direction in right and down direction.

equation (11) to estimate the person's speed.

$$s = \frac{d}{t} \tag{10}$$

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{11}$$

The time between two frames is calculated as t. We round the speed up because the time between two frames is very short.

XI. EVALUATION OF RESULTS AND PERFORMANCE

To determine if a target is successfully recognized, the intersection over union (IOU) ratio is used which is the ratio between the prediction box and ground truth box of the model and is widely used in the detection problems. The threshold value α can be set. When the IOU value is more than α the model's prediction is considered accurate and when the IOU value is less than α, the model prediction is considered inaccurate. In our case we set α value 0.5.

A. ACCURACY, mAP AND FPS

Accuracy is very important aspect to assess the performance of the model. The Hybrid YOLOv4 accuracy can be calculated using equation (12).

TABLE 2. Evaluation of the state-of-the-art models and the hybrid YOLOv4's performance.

Models	Accuracy	mAP(%)	FPS
SSD [7]	0.485	61.29	21.8
YOLOv3 [46]	0.576	80.52	49.1
YOLOv4 [9]	0.728	74.2	26
Hvbrid YOLOv4	0.951	92.1	48

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

The number of accurately anticipated positive samples is known as TP, FP for the quantity of incorrectly anticipated negative samples, FN for the quantity of samples that are wrongly forecast as positive, and TN for the proportion of accurately anticipated negative samples.

Target detection performance is frequently measured using the AP value. The mAP parameter shows the average accuracy of all categories and can be used to assess the model's performance across in all categories. AP and mAP can be calculated by using equation (14) and equation (15) respectively.

$$P = \int_0^1 P(R)dR \tag{13}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{14}$$

Precision, recall rate, and the total number of items across all categories are represented by letters P, R, and N, respectively. To determine how much a neural network has advanced and fast, the inference time must be calculated. Using a trained DNN model, deep learning inference is the act of making predictions based on previously unobserved data. To put it another way, the inference time is the length of time needed for a forward propagation to complete.

The detecting speed of the algorithm is measured in frames per second (FPS). The number of frames that can be handled per second is represented by this value. Various hardware configurations result in different processing speeds for different models. The FPs rate is significantly improved by the Hybrid Yolov4. Table 2 illustrates the comparison between state-of-the-art models and Hybrid Yolov4 accuracy, mAP and FPS. In terms of accuracy and mAP our model performance is better than the other state of the art models. However, the FPS of YOLOv3 is little higher than our model.

B. LOSS FUNCTIONS

Loss function in Hybrid YOLOv4 model consists of object localization offset loss, object classification loss, and object

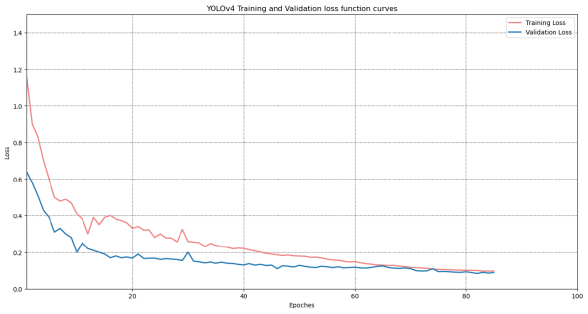

FIGURE 10. Hybrid YOLOv4 training and validation loss curves.

confidence loss are given in equations (16) to equation (18).

$$\text{Loss} = \lambda_1 L_{\text{conf}} + \lambda_2 L_{\text{cla}} + \lambda_3 L_{\text{loc}} \tag{15}$$

$$L_{\text{conf}} = -\sum \left(Obj_i \ln (p_i) + (1 - Obj_i) \ln (1 - p_i) \right) \tag{16}$$

$$L_{cla} = -\sum_{i \in Box} \sum_{j \in class} \left(ij \ln \left(p_{ij} \right) + \left(1 - O_{ij} \right) \ln \left(1 - p_{ij} \right) \right) \tag{17}$$

$$L_{loc} = 1 - IOU(A, B) + \frac{d^2_{AB}\left(A_{ctr}, B_{ctr}\right)}{l^2} + \alpha v \tag{18}$$

where α is the balance coefficient in equation (18). Obj_i in L_{conf} equation (18) shows if an object exists in the anticipated object bounding box and its value is 0 or 1. Where p_i represents the probability for real object in the predicted bounding box. Sigmoid function is used to calculate the probability value. In L_{cla}, O_{ij} indicate whether there is a j-class object and p_{ij} shows probability in the prediction bounding box i. The Hybrid YOLOv4 greatly reduces the loss function, that is, the Hybrid YOLOv4 training loss and validation loss reduce to 0.1% and 0.9% respectively as shown in Fig. 10.

XII. TESTING RESULTS OF HYBRIDISED YOLOv4

We have used videos of randomly selected crowded places, such as those taken during the Hajj and in front of some shopping malls to test our hybrid YOLOv4 model. As shown in Fig. 11, our algorithm first correctly detects the people in the videos before tracking them. When the person is detected, then the tracker assigns a unique ID to that person which also appears in the box above the bounding box. The model then successfully estimates every detected person's direction in which that person is going and also the estimates the speed of that person, which is shown in the box above every person's bounding box. In the left corner of the image appears the count. Therefore, two types of counting are shown in each image: the total counts, which shows how many individuals have been detected so far, and the count of people, which shows how many people have been detected and tracked in the current frame. In addition, it shows the general direction in majority of individuals are moving. Thus, along with this detailed knowledge information, managing large crowds is very simple.

FIGURE 11. Testing results of Hybrid YOLOv4.

XIII. CONCLUSION

In social sittings and religious sites where crowded areas often lead to serious security and safety problems, detection and tracking can be of extreme importance. Injuries and deaths similar to the Halloween incident in Seoul on October 30, 2022 are frequent during poorly organized events having

large crowds. The organizers mostly focus on monitoring individual behavior and the number of people in the crowd. However, as the scenes becomes more complex, the human detection systems that evaluate them become increasingly complex to operate in real-time situations, requiring a lot of processing power. In crowded environments, people huddle with each other, and detecting crowds embedded with obstruction becomes exceedingly challenging. As a result, we have focused on challenging crowd conditions. We build thus a compact and light model using the pruning approach that can now run on low-processing gadgets. By adding the CBAM module, we sought to address the issue of occlusion and opaque situations, which have enhanced the detection abilities of our model. We combined Deep SORT Tracker with YOLOv4 to improve tracking of people in extensively crowded situations, which is one of the most important aspects of our research. In addition, we estimate the orientation of the people and their speed in the crowd that is claimed to be the key contribution of this work. We have attained a higher level of accuracy and mAP values when compared to other state of the art models. The proposed hybrid YOLOv4 strategy reduces memory requirements while increasing computational operations. Our model has improved the inference and frames per second (FPS). In the future, this Hybrid model can be used for posture estimation, intent recognition and pedestrian future estimation as key components of crowd control.

REFERENCES

[1] Q. Wang, J. Gao, W. Lin, and X. Li, "NWPU-Crowd: A large-scale benchmark for crowd counting and localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 2141–2149, Jun. 2021, doi: 10.1109/TPAMI.2020.3013269.

[2] S. Lamba and N. Nain, "Crowd monitoring and classification: A survey," in *Advances in Computer and Computational Sciences* (Advances in Intelligent Systems and Computing). 2017, doi: 10.1007/978-981-10-3770-2_3.

[3] S. Elbishlawi, M. H. Abdelpakey, A. Eltantawy, M. S. Shehata, and M. M. Mohamed, "Deep learning-based crowd scene analysis survey," *J. Imag.*, vol. 6, no. 9, p. 95, Sep. 2020, doi: 10.3390/JIMAGING6090095.

[4] W. Albattah, M. H. K. Khel, S. Habib, M. Islam, S. Khan, and K. A. Kadir, "Hajj crowd management using CNN-based approach," *Comput., Mater. Continua*, vol. 66, no. 2, pp. 2183–2197, 2021, doi: 10.32604/cmc.2020.014227.

[5] S. Habib, A. Hussain, M. Islam, S. Khan, and W. Albattah, "Towards efficient detection and crowd management for law enforcing agencies," in *Proc. 1st Int. Conf. Artif. Intell. Data Anal. (CAIDA)*, Apr. 2021, pp. 62–68, doi: 10.1109/CAIDA51941.2021.9425076.

[6] M. H. K. Khel, K. Kadir, W. Albattah, S. Khan, M. N. M. Noor, H. Nasir, S. Habib, M. Islam, and A. Khan, "Real-time monitoring of COVID-19 SOP in public gathering using deep learning technique," *Emerg. Sci. J.*, vol. 5, pp. 182–196, Nov. 2021.

[7] B. S. V. Loghum, "YOLO!" *PodoPost*, vol. 31, no. 13, 2018, doi: 10.1007/s12480-018-0003-0.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Computer Vision—ECCV* (Lecture Notes in Computer Science). 2016, doi: 10.1007/978-3-319-46448-0_2.

[9] M. A. Almaiah, A. Ali, F. Hajjej, M. F. Pasha, and M. A. Alohali, "A lightweight hybrid deep learning privacy preserving model for FC-based industrial Internet of Medical Things," *Sensors*, vol. 22, no. 6, p. 2112, Mar. 2022.

[10] N. Waqas, S. I. Safie, K. A. Kadir, S. Khan, and M. H. K. Khel, "DEEP-FAKE image synthesis for data augmentation," *IEEE Access*, vol. 10, pp. 80847–80857, 2022.

[11] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[12] H. Idrees, K. Soomro, and M. Shah, "Detecting humans in dense crowds using locally-consistent scale prior and global occlusion reasoning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 1986–1998, Oct. 2015, doi: 10.1109/TPAMI.2015.2396051.

[13] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2325–2333, doi: 10.1109/CVPR.2016.255.

[14] C. Santhini and V. Gomathi, "Crowd scene analysis using deep learning network," in *Proc. Int. Conf. Current Trends Towards Converging Technol. (ICCTCT)*, Mar. 2018, pp. 1–5, doi: 10.1109/ICCTCT.2018.8550851.

[15] A. Ali, M. A. Almaiah, F. Hajjej, M. F. Pasha, O. H. Fang, R. Khan, J. Teo, and M. Zakarya, "An industrial IoT-based blockchain-enabled secure searchable encryption approach for healthcare systems using neural network," *Sensors*, vol. 22, no. 2, p. 572, Jan. 2022.

[16] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Computer Vision—ECCV* (Lecture Notes in Computer Science). 2016, doi: 10.1007/978-3-319-46475-6_28.

[17] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Computer Vision—ECCV* (Lecture Notes in Computer Science). 2016, doi: 10.1007/978-3-319-46493-0_22.

[18] C. J. Wu, S. Houben, and N. Marquardt, "EagleSense: Tracking people and devices in interactive spaces using real-time top-view depth-sensing," in *Proc. CHI Conf. Human Factors Comput. Syst.*, 2017, pp. 3929–3942, doi: 10.1145/3025453.3025562.

[19] J. Wetzel, A. Laubenheimer, and M. Heizmann, "Joint probabilistic people detection in overlapping depth images," *IEEE Access*, vol. 8, pp. 28349–28359, 2020, doi: 10.1109/ACCESS.2020.2972055.

[20] T. Van Oosterhout, S. Bakkes, and B. Kröse, "Head detection in stereo data for people counting and segmentation," in *Proc. VISAPP*, 2011, pp. 620–625, doi: 10.5220/0003362806200625.

[21] R. Nakatani, D. Kouno, K. Shimada, and T. Endo, "A person identification method using a top-view head image from an overhead camera," *J. Adv. Comput. Intell. Intell. Informat.*, vol. 16, no. 6, pp. 696–703, Sep. 2012, doi: 10.20965/jaciii.2012.p0696.

[22] C. Wateosot and N. Suvonvorn, "Top-view based people counting using mixture of depth and color information," in *Proc. ACIS*, 2013, pp. 1–4.

[23] C. Gao, J. Liu, Q. Feng, and J. Lv, "People-flow counting in complex environments by combining depth and color information," *Multimedia Tools Appl.*, vol. 75, no. 15, pp. 9315–9331, Aug. 2016, doi: 10.1007/s11042-016-3344-z.

[24] S. Mukherjee, B. N. Saha, I. Jamal, R. Leclerc, and N. Ray, "A novel framework for automatic passenger counting," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 2969–2972, doi: 10.1109/ICIP.2011.6116284.

[25] J. Garcia, A. Gardel, I. Bravo, J. L. Lazaro, M. Martinez, and D. Rodriguez, "Directional people counter based on head tracking," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 3991–4000, Sep. 2013, doi: 10.1109/TIE.2012.2206330.

[26] D. T. Nguyen, W. Li, and P. O. Ogunbona, "Human detection from images and videos: A survey," *Pattern Recognit.*, vol. 51, pp. 148–175, Mar. 2016, doi: 10.1016/j.patcog.2015.08.027.

[27] I. Ahmed and A. Adnan, "A robust algorithm for detecting people in overhead views," *Cluster Comput.*, vol. 21, no. 1, pp. 633–654, Mar. 2018, doi: 10.1007/s10586-017-0968-3.

[28] J. Watada, H. Zhang, H. Melo, D. Sun, and P. Vasant, "Boosted HOG features and its application on object movement detection," in *Advances in Intelligent Information Hiding and Multimedia Signal Processing* (Smart Innovation, Systems and Technologies). 2018, doi: 10.1007/978-3-319-63856-0_42.

[29] N. S. Punn, S. K. Sonbhadra, S. Agarwal, and G. Rai, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and deepsort techniques," 2020, *arXiv:2005.01385*.

[30] M. Ahmad, I. Ahmed, K. Ullah, and M. Ahmad, "A deep neural network approach for top view people detection and counting," in *Proc. IEEE 10th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf.*, Oct. 2019, pp. 1082–1088, doi: 10.1109/UEMCON47517.2019.8993109.

[31] G. Li, C. Zong, G. Liu, and T. Zhu, "Application of convolutional neural network (CNN)–AdaBoost algorithm in pedestrian detection," *Sensors Mater.*, vol. 32, no. 6, p. 1997, Jun. 2020, doi: 10.18494/SAM.2020.2787.
[32] S. D. Khan, F. Porta, G. Vizzari, and S. Bandini, "Estimating speeds of pedestrians in a real-world setting using computer vision," in *Proc. Int. Conf. Cellular Automata*. Cham, Switzerland: Springer, 2014, pp. 526–535.
[33] S. D. Khan, Y. Salih, B. Zafar, and A. Noorwali, "A deep-fusion network for crowd counting in high-density crowded scenes," *Int. J. Comput. Intell. Syst.*, vol. 14, no. 1, p. 168, Dec. 2021, doi: 10.1007/s44196-021-00016-x.
[34] M. Saqib, S. D. Khan, N. Sharma, and M. Blumenstein, "Crowd counting in low-resolution crowded scenes using region-based deep convolutional neural networks," *IEEE Access*, vol. 7, pp. 35317–35329, 2019, doi: 10.1109/ACCESS.2019.2904712.
[35] M. H. K. Khel, K. Kadir, S. Khan, W. Albattah, H. Nasir, M. Noor, A. Khan, and N. Waqas, "Hybridized YOLOv4 for detecting and counting people in congested crowds," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2022, pp. 1–6.
[36] S. Basalamah, S. D. Khan, and H. Ullah, "Scale driven convolutional neural network model for people counting and localization in crowd scenes," *IEEE Access*, vol. 7, pp. 71576–71584, 2019, doi: 10.1109/ACCESS.2019.2918650.
[37] S. D. Khan and S. Basalamah, "Sparse to dense scale prediction for crowd couting in high density crowds," *Arabian J. Sci. Eng.*, vol. 46, no. 4, pp. 3051–3065, Apr. 2021, doi: 10.1007/s13369-020-04990-w.
[38] L. Liang, J. Cao, X. Li, and J. You, "Improvement of residual attention network for image classification," in *Intelligence Science and Big Data Engineering. Visual Data Engineering* (Lecture Notes in Computer Science). 2019, doi: 10.1007/978-3-030-36189-1_44.
[39] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020, doi: 10.1109/TPAMI.2019.2913372.
[40] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1–9.
[41] K. Simonyan and A. Zisserman, "VGG-16," Tech. Rep., 2014.
[42] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," 2017, *arXiv:1611.06440*.
[43] H. Li, H. Samet, A. Kadav, I. Durdanovic, and H. P. Graf, "Pruning filters for efficient ConvNets," 2017, *arXiv:1608.08710*.
[44] G. Li, J. Wang, H.-W. Shen, K. Chen, G. Shan, and Z. Lu, "CNNPruner: Pruning convolutional neural networks with visual analytics," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 2, pp. 1364–1373, Feb. 2021, doi: 10.1109/TVCG.2020.3030461.
[45] V. Sindagi, R. Yasarla, and V. M. M. Patel, "JHU-CROWD++: Large-scale crowd counting dataset and a benchmark method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2594–2609, May 2022, doi: 10.1109/TPAMI.2020.3035969.
[46] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

MUHAMMAD HARIS KAKA KHEL was born in Charsadda, Khyber Pakhtunkhwa, Pakistan, in January 1997. He received the B.S. degree in electrical engineering (major in communication) from the University of Engineering and Technology (UET), Peshawar, Pakistan, in 2019. He is currently pursuing the master's degree with Universiti Kuala Lumpur–British Malaysian Institute, Malaysia, under the supervision of Dr. Kushsairy Abdul Kadir. His research interests include computer vision, deep learning, image processing, and wireless communication.

KUSHSAIRY ABDUL KADIR (Senior Member, IEEE) received the B.Sc. degree in engineering from the University of the West of England, Bristol, in 1998, the M.Sc. degree in mechatronic from International Islamic University, in 2007, and the Ph.D. degree in electronic and electrical engineering from Strathclyde University, in 2012. He is an Associate Professor with the Electrical Technology Section, Universiti Kuala Lumpur–British Malaysian Institute, where he is also the Dean. His current research interests include signal image processing, robotics for rehabilitation, and building energy efficiency. He is the Deputy Chair of the IEEE IMS Malaysia Chapter.

SHEROZ KHAN (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from the N-W. F. P. University of Engineering and Technology (UET), Peshawar, Pakistan, the M.Sc. degree in microelectronic computer engineering from Surrey University, U.K., in 1988, and the Ph.D. degree in electrical and electronic engineering from Strathclyde University, U.K., in August 1994. After serving with NWFP UET for five years, he joined Universiti Tenaga (UNITEN) as a Principal Lecturer, in January 2000. He joined International Islamic University Malaysia (IIUM), in January 2002. He has produced 22 M.Sc. under his direct supervision and ten M.Sc. under co-supervision. He has also produced 11 Ph.D.s and two postdoctorates under his direct supervision. He has been the Founder of the IIUM-Limoges, France, IIUM-Schmalkalden, UAS, Germany, and IIUM-IIU Islamabad, Pakistan, link programs. He has been with the Department of Electrical Engineering, Onaizah College of Engineering and Information Technology, since December 2019, where he established the IEEE Student Branch (STB64633) as a Founding Counselor. Being the best graduate of the Department of Electrical Engineering in 1982, he was awarded the university scholarship for doing the M.Sc. degree.

MNMM NOOR received the B.Sc. degree in engineering from the University of Kagoshima, Japan, in 1994, the master's degree in information technology from Open University Malaysia, in 2007, and the Ph.D. degree in information technology from Universiti Utara Malaysia, in 2012. He is currently a Senior Lecturer with Universiti Kuala Lumpur–Malaysian Institute of Information System. He is also the Principal Investigator of MIIT's Center of SDN/NFV IoT and a Fellowships Member of the Okinawa Open Laboratory, Japan. His current research interests include deep learning, cloud computing, and the IoT.

HAIDAWATI NASIR (Senior Member, IEEE) received the Ph.D. degree from the Department of Electrical and Electronics, University of Strathclyde, Glasgow, in 2012. She is an Associate Professor with the Computer Engineering Section, Universiti Kuala Lumpur–Malaysian Institute of Information Technology. Her research interests include signal and image processing, with application to super-resolution image reconstruction, image enhancement, and computer vision. She is the Student Activities Chair of the IEEE Malaysia Section and a Treasurer of the IEEE SPS Malaysia Chapter.

NAWAF WAQAS was born in Riyadh, Saudi Arabia, in March 1997. He received the B.Sc. degree in electrical engineering from the National University of Computer and Emerging Sciences, Pakistan, in 2018, and the M.Sc. degree in research from Universiti Kuala Lumpur–British Malaysian Institute (BMI), Gombak, Johar Baru, Malaysia, where he is currently pursuing the Ph.D. degree with the Malaysian Institute of Industrial Technology. His research interests include machine learning, deep learning, computer vision, medical image processing, segmentation, image and signal processing, and digital signal processing.

AKBAR KHAN received the B.E. degree (Hons.) in computer system engineering from BUET Khuzdar, in 2011, and the M.S. degree in computer engineering from BUITEMS, Quetta, Pakistan, in 2017. He is currently pursuing the Ph.D. degree with Universiti Kuala Lumpur–British Malaysian Institute Malaysia. In 2013, he joined BUITEMS, as a Lecturer with the Computer Engineering Department. He is an Assistant Professor with the Department of Computer Engineering, BUITEMS. His area of specialization is machine learning and computer vision.