Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(2): 2669–2690.
DOI: 10.3934/mbe.2023125
Received: 14 July 2022
Revised: 24 October 2022
Accepted: 31 October 2022
Published: 28 November 2022

Research article

Weapon operating pose detection and suspicious human activity classification using skeleton graphs

Anant Bhatt[*] **and Amit Ganatra**

Devang Patel Institute of Advance Technology and Research (DEPSTAR), Charotar University of Science and Technology (CHARUSAT), Nadiad Petlad Road, Changa, Gujarat-388421, India

[*] **Correspondence:** Email: capt.anant@gmail.com.

Abstract: Spurt upsurge in violent protest and armed conflict in populous, civil areas has upstretched momentous concern worldwide. The unrelenting strategy of the law enforcement agencies focuses on thwarting the conspicuous impact of violent events. Increased surveillance using a widespread visual network supports the state actors in maintaining vigilance. Minute, simultaneous monitoring of numerous surveillance feeds is a workforce-intensive, idiosyncratic, and otiose method. Significant advancements in Machine Learning (ML) show potential in realizing precise models to detect suspicious activities in the mob. Existing pose estimation techniques have privations in detecting weapon operation activity. The paper proposes a comprehensive, customized human activity recognition approach using human body skeleton graphs. The VGG-19 backbone extracted 6600 body coordinates from the customized dataset. The methodology categorizes human activities into eight classes experienced during violent clashes. It facilitates alarm triggers in a specific activity, i.e., stone pelting or weapon handling while walking, standing, and kneeling is considered a regular activity. The end-to-end pipeline presents a robust model for multiple human tracking, mapping a skeleton graph for each person in consecutive surveillance video frames with the improved categorization of suspicious human activities, realizing effective crowd management. LSTM-RNN Network, trained on a customized dataset superimposed with Kalman filter, attained 89.09% accuracy for real-time pose identification.

Keywords: human activity classification; weapon pose detection; weapon detection using skeleton graph; LSTM-based human activity recognition; crowd management using skeleton graphs; threat detection in crowd

1. Introduction

Rising armed violence, violent conflicts, armed protests, and criminal activities pose a significant concern for state actors. The contemporary global human rights issue involves the usage of explosive weapons and lethal personal arms, i.e., pistols, shotguns, automatic machine guns, revolvers, shotguns, and rocket launchers. Approximately 1.4 million deaths were recorded between 2012 to 2016, and two million people suffered physical injuries in armed protests and violence. Recent indicators show that 500 deaths and 2000 personnel injuries a day are a severe challenge to address [1]. The state actors aim to thwart threats by detecting miscreants in violent protests and curbing firearms trafficking. Technology growth helps monitor malicious activities using CCTV cameras and visual surveillance sensors. It encompasses minute monitoring of each video frame in numerous surveillance feeds with a high degree of attentiveness to identify suspicious activity simultaneously. The workforce-intensive, idiosyncratic manual analysis is a tedious and otiose method to observe each video frame in multiple inputs simultaneously.

The rapid evolution of Machine Learning (ML), Deep Learning (DL) algorithms, and Convolution Neural Networks (CNNs) help realize precise models that solve real-life challenges in medical, agricultural, traffic management, threat management, activity classification, object classification problems, and autonomous vehicles. Human activity classification and object detection approaches hold significant potential to transform video surveillance systems, human-computer interaction, and robotics for human behavior characterization. A distinct-object detection approach shows a substantial performance increase while using customized ConvNets (with superimposition of transfer learning) in the medical domain demonstrated suitability [2–4]. However, infrastructure intensive, the high training time with higher run time complexity of most CNN architecture demands a deliberate choice of CNNs.

Weapons detection and malicious activity detection are exploratory research areas where weapon and human pose classification approaches can be used for effective surveillance. In real-time scenarios, fixed CCTV and surveillance cameras capture specific visual coverage with defined and partial views in crowded, populated areas. The prominent viewing angle of the surveillance sensors in urban area capture weapons' chromatic exposure if the subject exposes the weapon. Captured visuals from crime scenes and event logs highlight that clear visuals are not available until the commitment of the crimes, thus unclear visuals of weapons. Weapon chromatic exposure and clarity are also subjected to the sensor's capability, i.e., angles, distance, and zoom. Figure 1 shows real-time surveillance visuals of various human activities in the mob. Various works have been focusing on the employment of object detection methodology. Various recent methodologies propose the employment of CNNs for weapon detection [5, 6]. Although these approaches demonstrate promising yields, most of this work proves a concept and lacks robustness for ground deployment. Bhatt et al. [7] demonstrated higher performance using singular classification to reduce the time complexity aiming ground deployable system . There have been challenges experienced on-the-ground employment by security agencies. In such scenarios, object detection underperforms while analyzing videos with hindered weapon visuals involving higher distances. Such methodology also suffers substandard identification accuracy while the subject is in possession of a small-sized personal weapon or operating the weapon at a distance. Human activity detection is one of the approaches which detects human activity based on various techniques. As weapon handling and operating is distinctly human activity, the human pose detection or activity classification approaches can address the real-time challenge described above. Hence, human activity recognition technology can help identify miscreants carrying or operating weapons. Although these approaches demonstrate promising yields, most of this work proves a concept and lacks robustness for ground deployment.

Malicious human activity recognition critically analyzes a mob or crowd's visual skimming for effective law order enforcement. Crowd management is a challenging situation handled by law enforcement agencies where mob management is the primary concern. In some cases, agitated groups of personnel or protests without arms demand identifying activities detrimental to a civilized society. Untoward activities, i.e., stampedes, also took a toll on human lives. Hence, detecting human activities like falling, running, stone pelting, and weapon handling helps improve mob management. While human activity recognition is addressed as a classification problem, it can be assimilated as activity recognition and the visuals' localization problem. The human figures' kinetic states can be classified into several activities like "walking", "running" and many more.

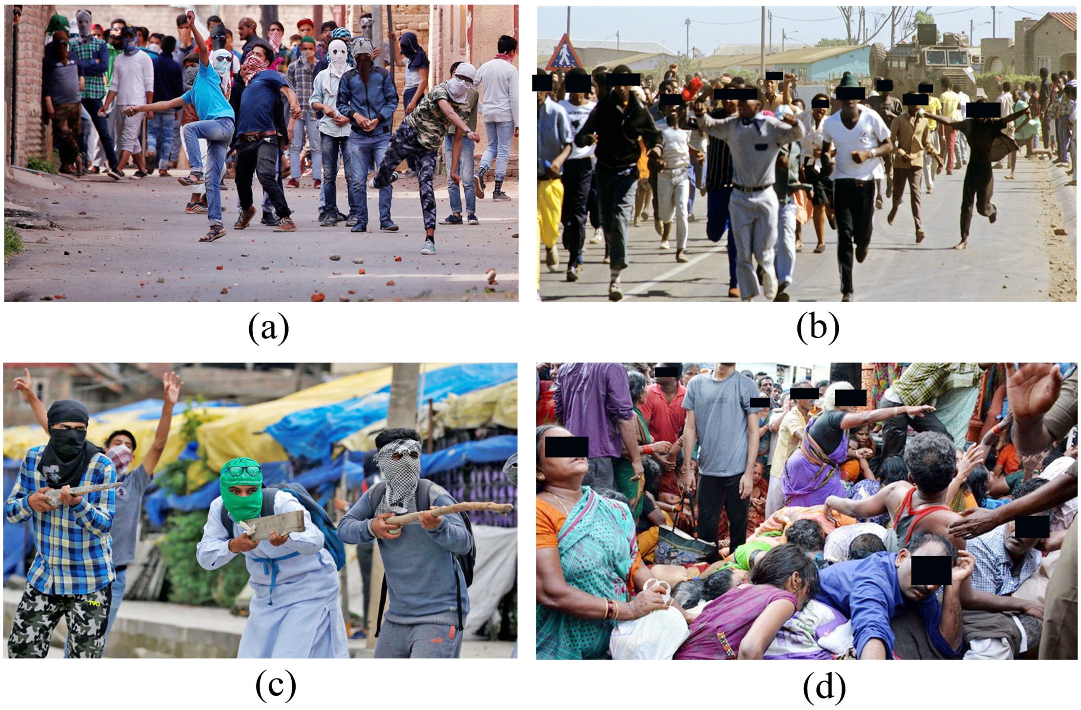

(a)
(b)
(c)
(d)

Figure 1. Surveillance visuals from the real-time hostile standoff showing (a) Stone pelting, (b) Hostile mob activities, (c) Agitation with weapon operating poses, (d) Crowd stampede.

Context-based classification and mapping enhance the scope to generate meaningful information by detecting various human objects in a scene. Identifying different body parts belonging to a single or multiple human bodies in a single surveillance frame and connecting body and foot key points is challenging. Effectiveness remains a core challenge while classifying human activities. A recent issue targets the detection of hand keypoint followed by combining these keypoint detection tasks into the same algorithm, which results in the estimation of the pose "whole-body" or "full-body" (body, face, hand and foot) [8, 9]. From the spatial perspective, specific human activity is associated with the main subset of joints that distinguishes the behavior. In the temporal perspective, final behavior can be concluded by building an action flow from sub-stages/frames that encompass different degrees of significance. In the real-time scenario, spatial and temporal perspective plays a vital role in mapping the multiple activities from multiple human movements. An attention mechanism is an efficient approach and can be explored for Skeleton-based behavior identification.

The experiments are motivated by the real-time challenge faced by the state actors. The paper proposes a methodology to identify the human activities for improved crowd management using various machine techniques. Existing human activity techniques superimposed with an effective customized overlay can aid in the identification of alarming and malicious in the crowd. Thus, the detected poses are classified in various human activities with a novel introduction of a class-'suspicious activity'. The real-time challenge demands highlighting suspicious activities by identifying weapon handling poses while standing or kneeling and also facilitates monitoring activities of stone-pelting and falling. Comprehensively, the problem statement incorporates effective threat identification and effective crowd management during hostile and violent standoffs. The real-time scenario inspires our work with challenges highlighted in earlier paragraphs. The paper proposes a novel methodology to identify malicious, alarming human activities using a skeleton graph for effective crowd management. The paper focuses on proposing a comprehensive pipeline for enhanced human activity classification. Our contributions are listed as follows:

- We present a novel human activity pose detection to classify suspicious human activity in the crowd using a comprehensive pipeline, achieving effective results.
- VGG architecture demonstrates promising results in skeleton graph generation on a customized dataset that comprises 6600 different body pose coordinates over eight activity classes.
- The pipeline incorporates the employment of LSTM for human activity recognition to achieve benchmark results by analyzing a series of visual surveillance frames, thus building an effective activity recognition for real-time detection.
- The Kalman filter overlay, in the end, ensures improved human activity classification, followed by a visual skeleton graph creation on the input surveillance feeds with visual classification output on the processed frames along with audio visual alarm generation for specific activity classification.

2. Literature review and methodology

Human pose estimation can be carried out with or without human body models, categorizing these methods into generative methods (model-based) or discriminative methods (model-free). Further, it can be classified into top-down and bottom-up methods, mainly considering the level from which the processing is carried out. We systematically studied the survey and review papers to identify the appropriate methodology for the defined problem definition. The surveys carried out on human motion analysis emphasize human pose estimation for real-life situations [10–14], and human motion analysis for video surveillance applications [15]. Some reviews focused on human motion capture systems [16, 17], model-based HPE [18,19], body parts-based HPE [20], and monocular-based HPE [21]. Skeleton-based, contour-based, and volume-based models are three prominent types of human body models that model the human body for human pose estimation [20,21]. Skeleton-based model is also known as the stick figure or kinematic model. It represents a set of joint (typically between 10 to 30) locations and the corresponding limb orientations following the human body's skeletal structure. Generated graph structure has vertices indicating joints and edges encoding constraints or prior connections of joints within the skeleton structure [22], and has been experimented with for 2D HPE [9]. This approach is very simple and flexible.

'Single Person Pose Estimation' derives inference from individual observations of body parts and spatial relations. The spatial model (based on graphical tree-structured models) encodes the spatial association between neighboring sections. It uses a kinematic line or non-tree model with expanding tree structure to identify and cater to occlusion, symmetry, and long-range relations. For position estimates, CNN achieves accurate spatial measurements of body parts. Multiple channels of a convolutional feature map embroil numerous semantic stages from the feature perspective, where each channel contributes to presenting relevant data samples for different actions. Most work proposes a top-down technique for multi-person pose estimation, which detects a person first, then measures each person's pose separately for each area detected. This approach not only suffers from early commitment to identifying persons but also scuffles to extract the spatial relations between multiple persons, allowing global inference. Existing methods consider the connections between the joints and bones and directly use physical information for modeling the topological structure of skeleton data. However, there exists a limitation in investigating every human action's key joints, bones, and body parts. Several experiments suggest using various methods for generating the body skeleton graphs.

2.1. Openpose: realtime multi-person 2D pose estimation using part affinity field

Openpose uses a nonparametric representation – Part Affinity Fields (PAFs) which descends knowledge to associate the concomitant body parts of a person within an image. The PAF refinement alone is more decisive for maximizing runtime accuracy than refining both PAF and body part location prediction [23, 24]. Openpose proposes a system for 2D pose detection of multiple persons using body, foot, hand, and vital facial points. They also include a runtime comparison to mask-R-CNN and alpha-pose, showing the computational advantage of our bottom-up approach [25]. This system uses non-maximum suppression on the detection confidence maps to obtain a discrete set of part candidate locations. The proposed work shows results for three models: MPII, COCO and COCO + foot, where (1) MPII recognizes 14 body parts. (2) COCO localizing 17 key points. (3) COCO + Foot includes a foot dataset for recognizing foot joints [26–28].

2.2. Centrality Graph Convolution Network (CGCN) for skeleton-based action recognition

The CGCN is the first work to highlight the centrality structures, i.e., key joints, bones, and body parts in human activity. The CGCN trails graph mechanisms in designing the centrality module to identify human action [29]. Temporal information models are derived using the motion information of a person entity between consecutive frames. The spatial and motion information is fed into a four-channel framework for the action recognition task. This model outperforms the state-of-the-art methods on to large-scale datasets for skeleton-based action recognition [30, 31].

2.3. Graph Convolution Network (GCN)

Graph Convolution Network methodology combines the topological structure with Graph Convolutional Networks, and it significantly increases performance by using the topological structure of skeleton data extracted from the human figure [32]. The approach proposes using topological information to distinguish key joints, bones and body parts, then highlighting the affirmative results using key joints and bones information in a four-channel framework. It shows an implementation of the reconstructed graph by the adaptive methods of the training process to bring out improvements. The models have been validated on NTU-RGB+D and kinetics datasets. The GSN modeling includes spatial and spectrum approaches, where the spatial approaches use graph theory to define nodes and edges for entities on data. Various experiments based on spectral approaches analyze the constructed graph in the frequency domain by leveraging the Laplacian eigenvector to transform a graph in the time domain [33–36]. The approach is known to have high computational costs and overheads. Hence, most methods use spatial approaches to construct CNN to classify human action. This approach provides scalability for large sizes of skeleton data [37–39].

2.4. Feedback Graph Convolution Network for skeleton-based action recognition

In FGCN, a multi-stage temporal sampling strategy is designed to extract spatial-temporal features for action recognition in a coarse-to-fine progressive process. A dense connection-based FGCN is proposed to introduce feedback connection into GCNs [40]. It transmits high-level semantics features to the low-level layers and flows temporal information stage by stage to progressively model. Early predictions are made on the FGCN model. The model gets partial knowledge of behavior in the early stages. Its forecasts are, of course, fairly coarse. The coarse predictions are essential to direct later-stage function learning to get precise prediction [41].

Body skeleton graphs can be constructed by applying CNN (Convolution Neural Network), GCN (Graph Convolution Network) and FGCN (Feedback Graph Convolution Network). Graph Convolutional Networks (GCNs) generalize the traditional technique to manage graph construction results using two main methods, i.e., the construction of GCNs with a temporal or spectral perspective. Spatial perspective approaches work out the convolution filters directly on the graph vertices and their neighbors [40]. On the other hand, spectral perspective techniques consider graph convolution as a type of spectral analysis using the Laplacian matrices graph's values and vectors. These approaches are based on the spatial perspective. The ST-GCN model claims it goes beyond the constraints of hand-crafted pieces and traversal rules used in previous methods [42]. These networks are based on CNN architecture and experimented on different datasets. On one side, where some of the proposed methods are fast but suffer from substandard accuracy, some show considerable accuracy with low speed. Some of the existing system models use RNN, which is less accurate than LSTM. Due to the advent of deep learning-based human pose estimation approaches, there has been a substantial novelty in approaching meaningful solutions with improved performance. Our work focuses on the amalgamation of human pose estimation and human activity classification, thus using multiple methods combined into one pipeline.

3. Material and methods

The paper proposes a comprehensive methodology by recognizing human activities without missing data on body joints and mapping accurate body pose coordinates to predict human activity while processing real-time surveillance video feed inputs. Human pose estimation has been one of the challenges explored by various experimental studies. Although datasets exist to explore the referenced problem definition, creating a universal dataset for this task is challenging as human poses are variant. There is a requirement for an appropriate dataset with balanced visuals in the human activity classes to detect and classify activities in the crowd-monitoring videos and visuals. Various datasets have been explored which may incorporate human pose with weapon handling, along with common activities noticed in the crowd, i.e., running, stampeding, and falling. We carried out the study of well-known datasets used in deep learning experiments or studies, i.e., MSCOCO [43], MPII [44], LSP [45], FLIC [46], PoseTrack [47] and AI Challenger [48]. Prominent limitations were envisaged while conceiving and conducting our experiments. Primarily, these datasets contained more images in complicated scenes, not befitting crowd management activities or problem definition. There exists a void of befitting datasets despite the availability of similar datasets that can be used for our experiments.

As the methodology is objected to highlighting suspicious activities by identifying weapon handling poses while standing or kneeling, stone-pelting and falling, we created a customized dataset. Eventually, the most perplexing concern is recognizing the activity from body coordinates in different activity/poses in the real-time surveillance visual feeds. The pose estimation model takes a processed camera image as the input and only estimates where key body joints are in the frame, giving output information about key points. In this model, we extract the joint coordinate of body parts using the TF-pose estimation library and manually enter data into the Excel/CSV files. The key points detected are indexed by a part ID, with a confidence score ranging from 0.0 to 1.0, indicating the probability that a key point exists in that position. Our experiments included 18 different body coordinates with respective part IDs, as shown in Table 1. These coordinates are expressed as (X, Y) where X contains (x,y) coordinates of body parts, and Y: class contains different human activities. A customized dataset was created using open-source images and videos, which include human activities classified as 'Shoot', 'Crawl' and 'Throw'. On the customized dataset, the experiments collected 18 different body joints for every activity and added variations in the data to make a more specific dataset. TF-pose estimation library [49] collects the various human pose coordinates. It incorporates approximately 6600 different body pose coordinates from eight human activities. The final dataset has eight human activity classes (Stand, Walk, Run, operate, fall, Shoot, Crawl and Throw). Figure 2 shows eight human pose classes with respective numbers of body pose coordinates.

Table 1. The table highlights 18 different body coordinates with respective part IDs while customizing the dataset.

ID	Keypoints	ID	Keypoints	ID	Keypoints
1	Nose	2	Right Knee	3	Neck
4	Right Foot	5	Right Shoulder	6	Left Hip
7	Right Elbow	8	Left Knee	9	Right Wrist
10	Left Foot	11	Left Shoulder	12	Right Eye
13	Left Elbow	14	Left Eye	15	Left Wrist
16	Right Ear	17	Right Hip	18	Left Ear

The experiments were conducted by training the VGG-19 backbone for the skeleton graph generation. The Deepsort person tracking algorithm was used for multiple people tracking over consecutive video frames, generating bounding boxes. LSTM was trained on a customized dataset (incorporating 6600 body poses with eight human activity classes). Finally, in the last stage, the pipeline recognizes human activity and categorizes it as normal or suspicious. Figure 3 shows the proposed architecture of the pipeline. We conducted our experiments on Nvidia GTX 2080Ti GPU (4352 CUDA cores).

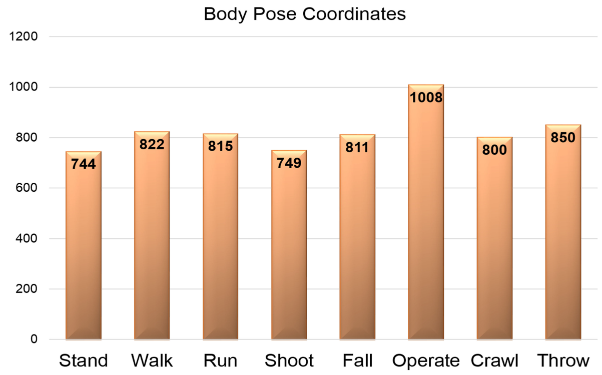


Figure 2. Statistical distribution of body-pose coordinates of eight human activity classes.

3.1. Model implementation

3.1.1. VGG backbone

As the initial phase of the pipeline proposed using CNN for skeleton graph generation, we explored VGG, Alexnet, and InceptionNet V3 architectures. Abouelnaga et al. proposed a genetically-weighted ensemble of Convolutional Neural Networks (CNNs) for driving posture estimation classification [50]. The methodology showed 95.98% accuracy with AlexNet architecture. AlexNet-eight layers architecture with learnable parameters has five convolution layers with a combination of max-pooling layers and two dropout layers, followed by three fully connected layers. Relu and softmax activation functions were used in all output layers. It achieved 93.65%, 93.60%, 84.23%, 89.52% and 86.68% accuracy on original, Skin segmented, face, hand and face + hand, respectively. A thinned version of the ensemble achieved 94.29% classification accuracy for a real-time environment. Various experiments using AlexNet and InceptionNet V3 were conducted in the same proposed methodology. Table 2 shows the performance attained in the paper.

In the proposed pipeline, Very Deep Convolutional Networks are designed to carry out a large-scale image recognition baseline model that extracts the body joints. It joins the graph and creates a full-body skeleton graph which is given as input to the human activity recognition algorithm. VGGNet uniform Architecture consists of 13 convolutional layers, five max-pooling layers and three fully-connected layers consisting of 138 million parameters [51,52]. Efficient feature extraction capability makes it a preferred CNN for computer vision challenges [53]. VGG16 and VGG19 CNN architectures perform well in several object detection scenarios. Bhatt et al. showed considerable efficiency while employing them for real-time surveillance video analysis [2,3]. The proposed pipeline uses a VGG (pre-trained on a COCO body dataset) model [54]. We maintain VGG-19 as the backbone. The probability of picking an image from each dataset is 76.5% for COCO, 5% each for foot and MPII datasets, 0.33% for each face dataset, 0.5% for Dome hand, 5% for MPII hand, 5% for whole-body data, and 2% for picking an image with no people in it [55]. Training the VGG net is the same as the CMU providing the Caffe model in the openpose library. Considering the real-time surveillance video analysis, low graph accuracy may result in an inefficient system. The system may fail to identify or classify correct human activity due to less accurate/incorrect graph generation in the early phase of the pipeline. Hence, the VGGNet increases the efficiency of skeleton graph generation. We used Local PAF Threshold = 0.2, PAF Count Threshold = 5, Part Count Threshold = 4, and Part Score Threshold = 4.5. Figure 4 shows VGG architecture, used for extracting body joints and joints, thereby creating a full-body skeleton graph.

Table 2. The table highlights a comparative analysis of the datasets used by the researchers during their experiments.

CNN/ Model	Source	Accuracy %
AlexNet [4]	Original	93.65
	Skin segmented	93.60
	Face	84.23
	Hand	89.52
	Face + Hand	86.68
InceptionNet V3	Original	95.17
	Skin segmented	94.57
	Face	88.82
	Hand	91.62
	Face + Hand	90.88

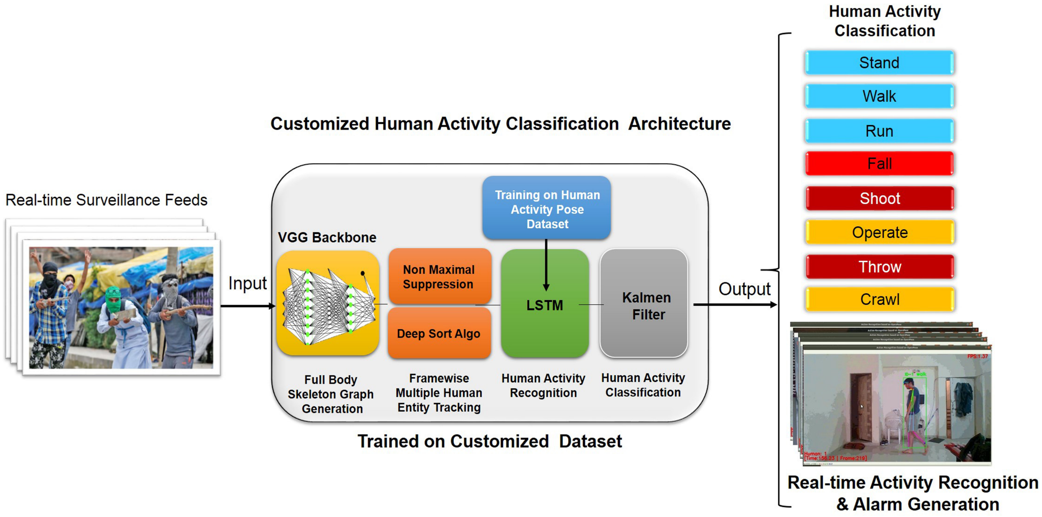

Human Activity Classification
Stand
Walk
Run
Fall
Shoot
Operate
Throw
Crawl
Customized Human Activity Classification Architecture
Real-time Surveillance Feeds
Input
VGG Backbone
Training on Human Activity Pose Dataset
Non Maximal Suppression
Deep Sort Algo
LSTM
Kalmen Filter
Output
Full Body Skeleton Graph Generation
Framewise Multiple Human Entity Tracking
Human Activity Recognition
Human Activity Classification
Trained on Customized Dataset
Real-time Activity Recognition & Alarm Generation

Figure 3. Proposed methodology for a comprehensive, end-to-end pipeline for human activity classification.

3.1.2. Non-Maximum Suppression

Non-Maximal Suppression (NMS) helps in whittling down many detected bounding boxes to only a few from several overlapping entities [56]. Numerous anchors of various sizes and shapes supposedly contain only one or few objects. Classification criteria can be used to conclude the object class. A classifier obtains a probability score or similarity measure, e.g., IOU. We use NMS to represent the detection of the human body and separate each person with an individual bounding box. Most target detection algorithms use NMS to cut a considerable number of rectangles observed down to a handful [57–59]. The window size value was taken as 3, and the threshold was set to 0.15. The real-time deployment demands effective bounding box generation over the fast-moving, dynamic objects in consecutive surveillance video frames. Figure 5 shows bounding box generation in consecutive video frames.

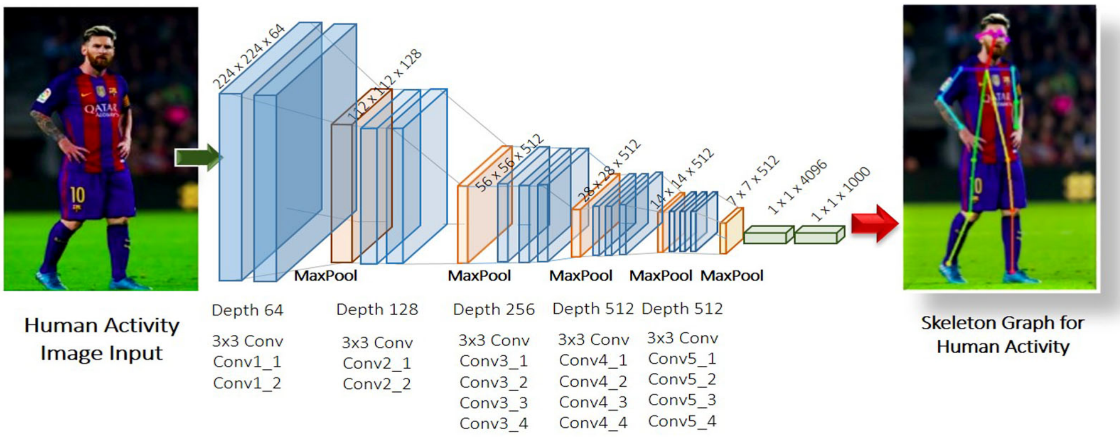


Figure 4. VGG architecture for the creation of full-body skeleton graph.

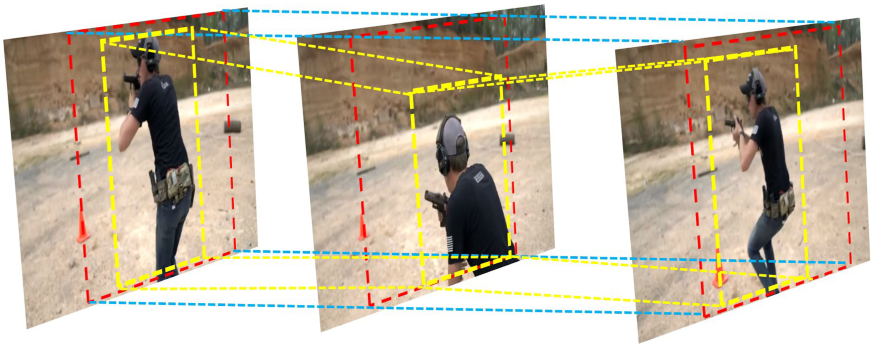


Figure 5. Non-Maximal Suppression for effective bounding box generation in consecutive surveillance video frames.

3.1.3. Deep Sort algorithm

Deep-Sort-an improvement of the SORT algorithm integrates the appearance information of objects to enhance associations [60]. Data association integrates an additional appearance metric based on pre-trained CNNs allowing the re-identification of tracks based on feature similarity besides overlap. The Deep Sort associates objects in multiple previous frames, alleviating the occlusion issue and rediscovering objects. Real-time scenario demands tracking specific personnel for mapping its activity in consecutive frames for further activity classification [61]. The module holds ID assignments to a person in subsequent frames unless the individual moves out of the visuals. The module considers ID assignments afresh on any person's reappearance to enable tracking of the activity [62,63]. Assignment of unique IDs to human entities helps track the same person in video frames, making the model robust for the actual scenario. A nearest-neighbor distance metric is a distance calculation class that returns the nearest sample by calculating two measurement methods, i.e., Euclidean Distance and Cosine Distance. We experimented with the maximum cosine distance value taken as 0.3. Some thresholds for detection are defined, and the feature extractor network is loaded. Figure 6 shows the allotment of IDs to multiple human objects for tracking.

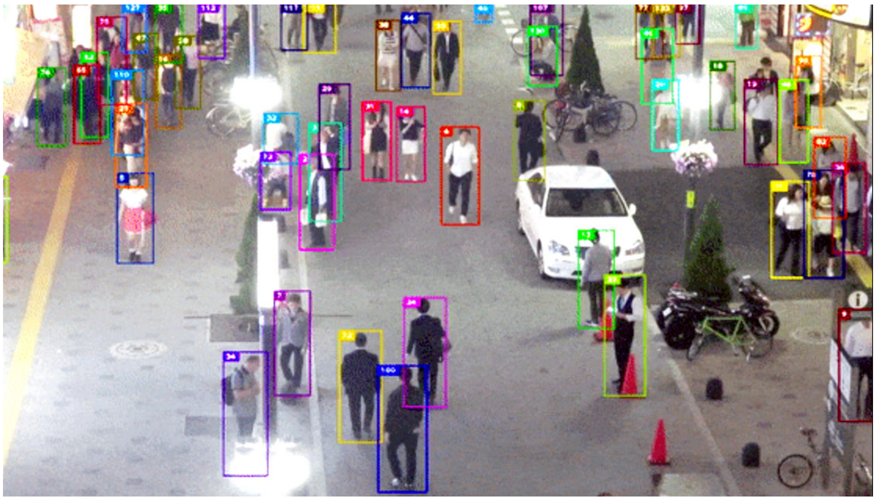


Figure 6. Multi person tracking with unique ID assignment using DeepSORT [65].

3.1.4. Long Short Term Memory-Recurrent Neural Network(LSTM-RNN)

LSTM is an RNN architecture that can remember values over arbitrary intervals, making them better suited to classify, process, and predict time series. LSTM has a distinct advantage due to its insensitivity to gap length over alternative RNNs, hidden Markov models and other sequence learning methods. Considering the inherent advantage of LSTM, the proposed methodology incorporates experiments using LSTM. The proposed pipeline involves LSTM experiments to analyze live surveillance videos for the classification of human activities correctly. The paper proposes the employment of LSTM to identify malicious human activities from multiple human entities in consecutive video frames. Long-term memory-referred to as cell state and its separate opening, is LSTM's central principle. The cell state moves quantitative knowledge down the chain of sequences. It can be assimilated as the network's 'memory', where the cell state will hold relevant information throughout the sequence production. As the cell state moves, information is added or removed through gates to the cell state. The gates are specific neural networks that determine which knowledge of the cell state is enabled. During the preparation, the gates will learn what information is relevant to hold or forget [19]. As details from the early steps will make it through later, it minimizes the short-term memory effects and can find similarities in more video frames. Figure 7 shows

LSTM-RNN Single memory cell [64].

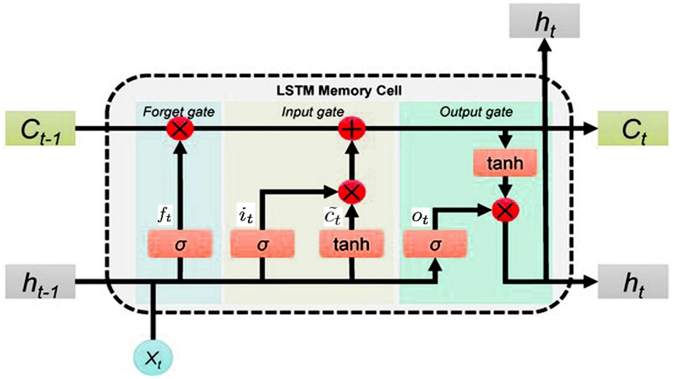
The key to LSTM is the cell state $\mathbb{C}_t$, which keeps the information along it unchanged. Three gates, named forget gate, input, and output gate, regulate the cell state to let information through optionally. The forget gate controls which elements of the cell state vector $\mathbb{C}_{t-1}$ will be forgotten, thereby deciding whether the previous activity is related to the present activity or not. Thus, the forget gate decides whether the output of the previous field is important for the next cell output or not.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{3.1}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{3.2}$$

$$\tilde{\mathbf{C}} = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{3.3}$$

where f_t is an output vector of the sigmoid layer whose value ranges between 0 and 1. W_f and b_f define the trainable parameters. The input cell (a value ranging between 0 and 1) decided which of the body coordinates is updated by the latest input of the input cell with W_i and b_i being the trainable parameters. $\tilde{C}$ (a value ranging between 0 and 1) is a potential vector of cell state that is computed by the current input x_t and the last hidden state h_{t-1}. In this model, tanh is the hyperbolic tangent, and W_c and b_c are the trainable parameters.



$i_t \rightarrow$ *represents input gate.*
$f_t \rightarrow$ *represents forget gate.*
$o_t \rightarrow$ *represents output gate.*
$\sigma \rightarrow$ *represents sigmoid function.*
$b_x \rightarrow$ *biases for the respective gates*(x).

$w_x \rightarrow$ *weight for the respective gate*(x) *neurons.*
$h_{t-1} \rightarrow$ *output of the previous lstm block(at timestamp* $t-1$).
$x_t \rightarrow$ *input at current timestamp.*
$c_t \rightarrow$ *cell state(memory) at timestamp*(t).
$\tilde{c}_t \rightarrow$ *represents candidate for cell state at timestamp*(t).

Figure 7. LSTM-RNN single memory cell.

After that, we can update the old cell state C_{t-1} into the new cell state C_t by element-wise multiplication:

$$C_t = f_t C_{t-1} + i_t \tilde{C}^t \tag{3.4}$$

Finally, the output gate decides which to be output by a sigmoid layer:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{3.5}$$

where o_t is a vector (value ranging between 0 and 1), and W_o, b_o are trainable parameters.

The new hidden state ht is then calculated by combining Eqs (5) and (6):

$$h_t = o_t \odot \tanh(C^t) \tag{3.6}$$

ADAM is an optimization algorithm that can be used instead of the classical stochastic gradient descent procedure to update network weights iteratively based on training data. Stochastic gradient descent maintains a single learning rate (termed alpha) for all weight updates, and the learning rate does not change during training. Adam optimizer combines Adaptive Gradient Algorithm, and Root Mean Square Propagation [66]. Adam optimizer is good in back-propagation and changes weight according to the variation between the actual output and the prediction output. Hence, LSTM models acquire good accuracy and benefit when applying real-time pose estimation.

3.1.5. Kalman filter

Kalman algorithm is a recursive algorithm in which time series is used noise data to remove inaccuracies in the measurement of multiple variables and projections of the variables more complex than single measurements, and thus an algorithm to time series. Kalman addresses the complexities of variables with higher weights to higher estimates unsure [67]. The Kalman filter decides the variance between this model's real-time body coordinates and actual training coordinates. The Kalman filter gives the most probable output of the human pose. Thus, the Kalman filter also constructs a state transition model to find the most localized and reliable value for the next state prediction. Using Kalman filter, significantly fluctuating serial data can be transformed into a successful application with real-time tracking functions.

4. Results and discussion

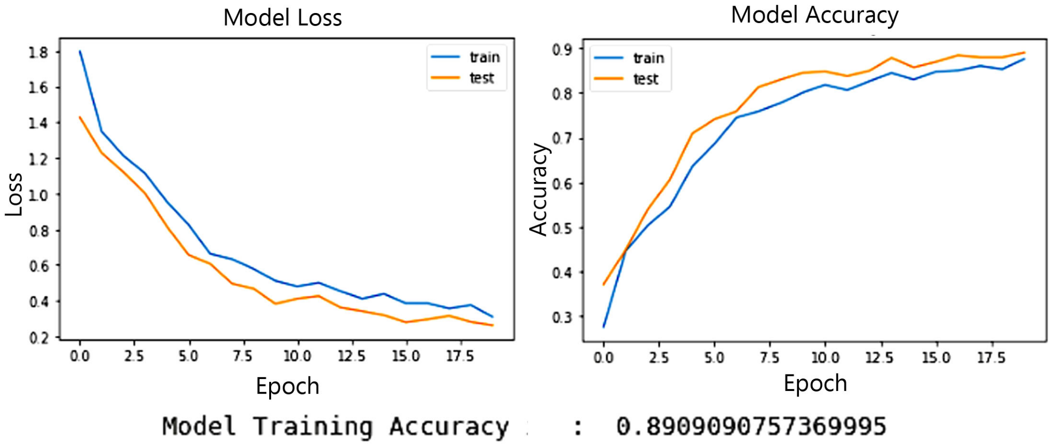
Pre-Trained VGG19-generated full-body skeleton graph. It attained 96.31% and 94.44% accuracy with regularization and without regularization, respectively. The VGG model gives good accuracy while generating a full-body skeleton graph. Good accuracy of VGG net ensures that the accuracy of the recognition model is also good VGG train on balance data which contain COCO body data set, Dome hand, and MPII hand dataset to achieve good accuracy. Table 3 shows results generated using VGG-19 architecture on our customized dataset. After generating a full-body skeleton graph, NMS separates each person's body by creating bounding boxes around it. It is clarified that the Skeleton graph of the person's body. Deep sort algorithms help track specific human bodies by giving them individual IDs.

Table 3. The table highlights the performance of the VGG model.

CNN/Model	Source	Accuracy %
Original VGG (140 M parameter)	Full body	94.44
Original VGG with Regularization (140 M parameter)	Full Body	96.31

Recurrent Neural Networks (RNN) are short-term memory sufferers, especially when a series is long enough. They may find it impossible to bring knowledge from early stages in time to later ones. RNNs leave crucial details at the start and suffer from the vanishing gradient problem, where gradients are values used to update a neural network's weights. As the gradient shrinks when it propagates back through time, the gradient value becomes extremely small and doesn't contribute much learning. So, layers that get a slight gradient upgrade avoid learning. Since these earlier layers don't remember, RNNs can forget what they saw in longer sequences. Hence, they have short-term memory. Approaches highlighted in the literature survey section impose a time penalty as they involve more stages to process a single frame. As Inference time is critical while envisaging real-time scenarios, our LSTM-based methodology only needs to go through a single stage for every video frame, thus performing more efficiently than multi-stage CNN-based methods. The LSTM implementation model showed 25.6 per frame, which is more effective than the flow-based methods, i.e., Thin-Slicing Net [68]. The paper addressed the transition of memory content resulting from the changing positions. The LSTM memory cell containing global and local information helps predict spatially correlated joints on a single frame. LSTM also maintains memory by using useful prior information and new knowledge, thus capturing temporal geometric consistency [69].

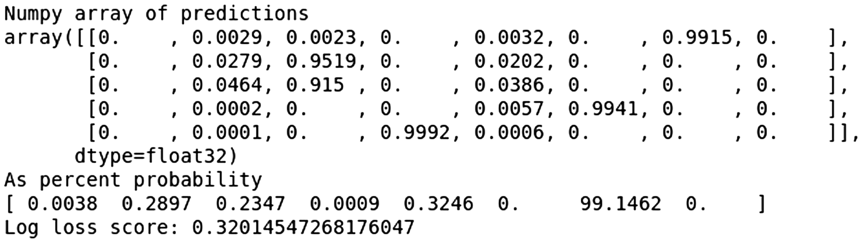
Implementation of a linear stack of layers-Sequential model was done. In the recognition, part LSTM trains on 6600 different human coordinates with one layer of Keras LSTM model, with 0.5 dropout and two fully connected dense layers, which gives a weight file with around 52 k parameters. With epochs = 20, and batch size = 32, we used categorical cross-entropy as the loss function. Finally, softmax activation was implemented. In this model output of the LSTM cell is continuously updated by the next activity and, using a Kalman filter, calculates MSE between actual data and predicts data, and decides the most accurate activity of the person. The LSTM model achieved 89.09% accuracy with a nominal loss, i.e., approx 0.25, while training the dataset. The log loss of the LSTM model is around 0.32, which is desirable for multi-class prediction. Figure 8 shows loss function and accuracy graphs while training the LSTM network, which attained 89.09% accuracy. According to the confusion matrix, the prediction of individual activity is accurately matched with the actual activity label. We analyze the run-time efficiency in two parts, i.e., CNN processing time and Multi-person parsing time. The earlier part has a time complexity of $O(1)$, whereas the second part has $O(n^2)$ time complexity, where n is the number of persons in the frame.



```
Model Training Accuracy   :  0.8909090757369995
```

Figure 8. Graphs show loss and accuracy scores during training and testing phases.

Figure 9 shows the log loss score for human activity classification while training the LSTM on a customized dataset with 6600 coordinates. Figure 10 shows the confusion matrix for multi-class human activity classification results, showing benchmark scores for the ‘Run’, ‘Weapon Handling-Shoot’, ‘Crawl’ and ‘Throw’ human activity classes. These human activities have been reported during violent standoffs, hence being classified as ‘Suspicious activities’. Figure 11 shows human activity classification output obtained using the proposed end-to-end pipeline for (a) Walk, (b) Run, (c) Stand and (d) Fall. Figure 12 shows weapon handling and operating activity (while the person is standing or kneeling) classification output obtained using the proposed end-to-end pipeline.



```
Numpy array of predictions
array([[0.    , 0.0029, 0.0023, 0.    , 0.0032, 0.    , 0.9915, 0.    ],
       [0.    , 0.0279, 0.9519, 0.    , 0.0202, 0.    , 0.    , 0.    ],
       [0.    , 0.0464, 0.915 , 0.    , 0.0386, 0.    , 0.    , 0.    ],
       [0.    , 0.0002, 0.    , 0.    , 0.0057, 0.9941, 0.    , 0.    ],
       [0.    , 0.0001, 0.    , 0.9992, 0.0006, 0.    , 0.    , 0.    ]],
      dtype=float32)
As percent probability
[ 0.0038  0.2897  0.2347  0.0009  0.3246  0.     99.1462  0.    ]
Log loss score: 0.32014547268176047
```

Figure 9. Log-loss score of the LSTM model: human activity classification.

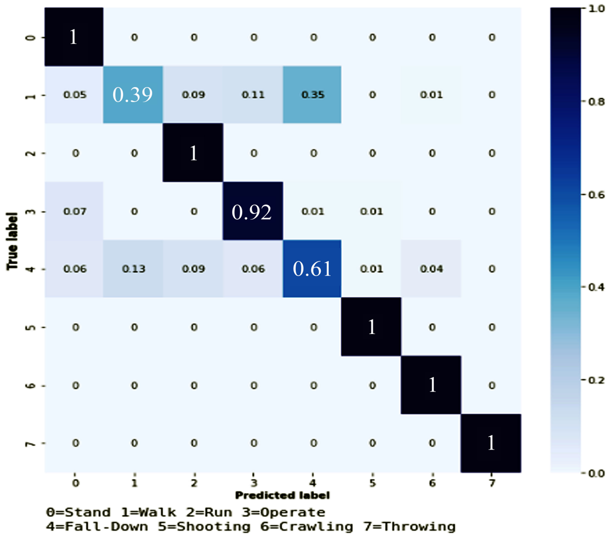


Figure 10. Confusion matrix for multi-class human activity classification results obtained using the proposed methodology, highlighting optimum results.

4.1. Potential application

Real-time challenges can be addressed by formulating an efficient human pose classification system. Lightweight, precise weapon pose estimation becomes a force multiplier in real-time situational scan and weapon detection. A prominent gap identified while employing an object detection system for weapon detection can be bridged with the proposed approach in the paper. During hostile stand-offs, operations being conducted by law enforcement agencies can be empowered where the hostile dynamic situations can be effectively controlled. Prompt detection of firearms and weapons and malicious activities helps identify the miscreants. Crime Identification while scanning the surveillance visuals from widespread CCTV networks in smart cities can enhance monitoring and surveillance in cities, responding effectively to crimes. An additional layer of geo-tagging can help limit the crimes, i.e., felony crimes, loot, robbery, and organized civil crimes. Border surveillance and areas with disturbed situations can be put under effective scans, where prompt identification of miscreants can be made. Finally, the methodology is best suited for crowd management, where the immediate alarm can be raised on detecting even the minutest malicious activity of an individual.

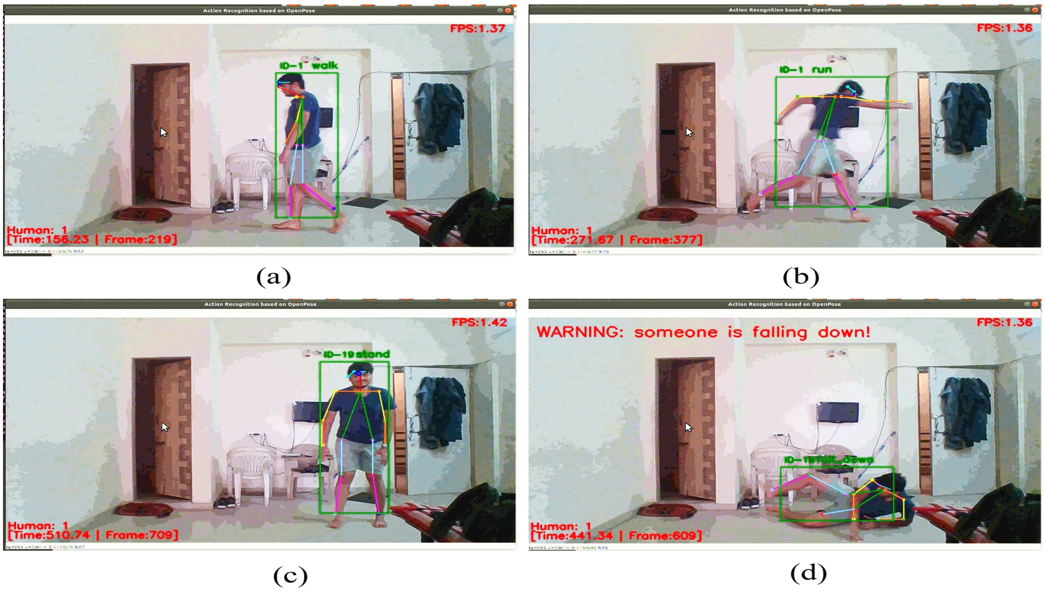

Action Recognition based on OpenPose
FPS:1.37
ID-1 walk
Human: 1
[Time:156.23 | Frame:219]
(a)
Action Recognition based on OpenPose
FPS:1.36
ID-1 run
Human: 1
[Time:271.67 | Frame:377]
(b)
Action Recognition based on OpenPose
FPS:1.42
ID-19 stand
Human: 1
[Time:510.74 | Frame:709]
(c)
Action Recognition based on OpenPose
WARNING: someone is falling down!
FPS:1.36
Human: 1
[Time:441.34 | Frame:609]
(d)

Figure 11. Human activity classification output obtained using proposed end-to-end pipeline for (a) Walk, (b) Run, (c) Stand and (d) Fall.

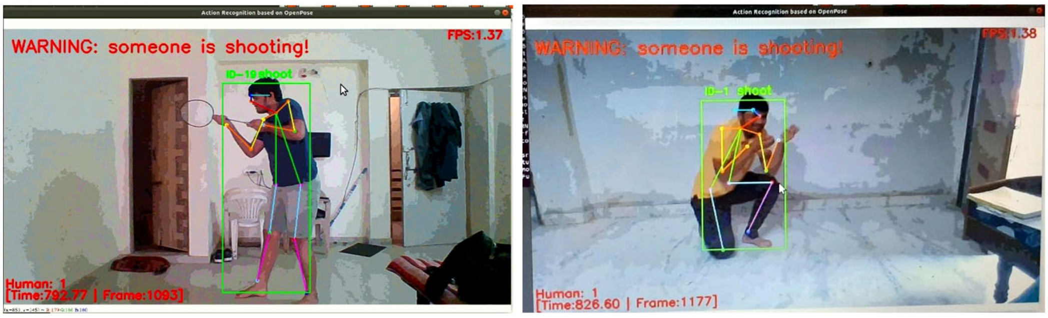

Action Recognition based on OpenPose
WARNING: someone is shooting!
FPS:1.37
ID-19 shoot
Human: 1
[Time:792.77 | Frame:1093]
Action Recognition based on OpenPose
WARNING: someone is shooting!
FPS:1.38
ID-1 shoot
Human: 1
[Time:826.60 | Frame:1177]

Figure 12. Weapon handling, operating activity classification output obtained using proposed end-to-end pipeline for (a) Weapon operating in standing pose, (b) Weapon operating in kneeling pose.

5. Conclusions

Human activity recognition is a well-established computer vision problem that plays an influential role in human-to-human interaction. Human activity classification aids in bio-metrics applications, video-surveillance, human-computer interaction, and crowd management. Real-time multi-person activity recognition is essential for identifying human activities in crowded places. Existing approaches can recognize a single human activity. There exists a significant scope in real-time, multi-person activity recognition, thereby segregating a set of human activities as-'Suspicious'. The proposed methodology presents a comprehensive methodology for effective law enforcement. Firstly, the pre-trained VGG model demonstrated 96.31% accuracy for body joint detection, finding the correct body joint, and creating a body skeleton graph. Subsequently, deepsort algorithm and NMS helped improve performance in separating multiple human body skeleton graphs in surveillance visual. The customized dataset included 6600 body pose coordinates defined in eight classes (stand, walk, run, operate, fall, shooting, crawling, and throwing). LSTM model-trained on the customized dataset, attained 89.09% accuracy, establishing benchmark performance for time sequence surveillance visuals. Lastly, the pipeline incorporated a Kalman filter for classification, making LSTM more effective in testing on real-time video surveillance. The human activity alarm trigger extends enhanced assimilation of the generated results, highlighting the 'Suspicious' human activities. A comprehensive pipeline is a prominent approach to circumvent real-time challenges in identifying suspicious activities in violent conflicts, armed violence, and standoff.

Acknowledgments

We want to convey gratitude to Provost, Registrar, CHARUSAT, Changa, India, for the support during this work, without which the outcome would not have been possible. We also thank Fenil Godhani for technical assistance during the dataset creation and graphical exploration.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. A. International, Gun violence–key facts, 2017. Available from: https://www.amnesty.org/en/what-we-do/arms-control/gun-violence/.
2. A. R. Bhatt, A. Ganatra, K. Kotecha, Cervical cancer detection in pap smear whole slide images using convnet with transfer learning and progressive resizing, *PeerJ Comput. Sci.* , **7** (2021). http://dx.doi.org/10.7717/peerj-cs.348
3. A. Bhatt, A. Ganatra, K. Kotecha, Covid-19 pulmonary consolidations detection in chest x-ray using progressive resizing and transfer learning techniques, *Heliyon*, **2021** (2021). http://dx.doi.org/10.1016/j.heliyon.2021.e07211
4. A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with Deep Convolutional Neural Networks, *Commun. ACM*, **60** (2017), 84–90. http://dx.doi.org/10.1145/3065386

5. M. T. Bhatti, M. G. Khan, M. Aslam, M. J. Fiaz, Weapon detection in real-time cctv videos using deep learning, *IEEE Access*, **9** (2021), 34366–34382. http://dx.doi.org/10.1109/ACCESS.2021.3059170

6. N. Dwivedi, D. K. Singh, D. S. Kushwaha, Weapon classification using Deep Convolutional Neural Network, in *2019 IEEE Conference on Information and Communication Technology*, IEEE, 2019, 1–5. http://dx.doi.org/10.1109/CICT48419.2019.9066227

7. A. Bhatt, A. Ganatra, Explosive weapons and arms detection with singular classification (WARDIC) on novel weapon dataset using deep learning: enhanced OODA loop, *Eng. Sci.*, **20** (2022). http://dx.doi.org/10.30919/es8e718

8. M. Dantone, J. Gall, C. Leistner, L. Van Gool, Human pose estimation using body parts dependent joint regressors, in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, 3041–3048. http://dx.doi.org/10.1109/CVPR.2013.391

9. Z. Cao, T. Simon, S. E. Wei, Y. Sheikh, Realtime multi-person 2d pose estimation using part affinity fields, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, 7291–7299. http://dx.doi.org/10.1109/CVPR.2017.143

10. X. Ji, H. Liu, Advances in view-invariant human motion analysis: a review, *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, **40** (2010), 13–24. http://dx.doi.org/10.1109/TSMCC.2009.2027608

11. D. M. Gavrila, The visual analysis of human movement: a survey, *Comput. Vision Image Understanding*, **73** (1999), 82–98. http://dx.doi.org/10.1006/cviu.1998.0716

12. T. B. Moeslund, A. Hilton, V. Krüger, L. Sigal, *Visual Analysis of Humans*, Springer, 2011. http://dx.doi.org/10.1007/978-0-85729-997-0

13. R. Poppe, Vision-based human motion analysis: an overview, *Front. Sports Active Living*, **108** (2007), 4–18. http://dx.doi.org/10.1016/j.cviu.2006.10.016

14. J. K. Aggarwal, Q. Cai, Human motion analysis: a review, *Comput. Vision Image Understanding*, **73** (1999), 428–440. http://dx.doi.org/10.1006/cviu.1998.0744

15. W. Hu, T. Tan, L. Wang, S. Maybank, A survey on visual surveillance of object motion and behaviors, *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, **34** (2004), 334–352. http://dx.doi.org/10.1109/TSMCC.2004.829274

16. T. B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, *Comput. Vision Image Understanding*, **81** (2001), 231–268. http://dx.doi.org/10.1006/cviu.2000.0897

17. T. B. Moeslund, A. Hilton, V. Krüger, A survey of advances in vision-based human motion capture and analysis, *Comput. Vision Image Understanding*, **104** (2006), 90–126. http://dx.doi.org/10.1016/j.cviu.2006.08.002

18. M. B. Holte, C. Tran, M. M. Trivedi, T. B. Moeslund, Human pose estimation and activity recognition from multi-view videos: comparative explorations of recent developments, *IEEE J. Sel. Top. Signal Process.*, **6** (2012), 538–552. http://dx.doi.org/10.1109/JSTSP.2012.2196975

19. X. Perez-Sala, S. Escalera, C. Angulo, J. Gonzalez, A survey on model based approaches for 2d and 3d visual human pose recovery, *Sensors*, **14** (2014), 4189–4210.

20. Z. Liu, J. Zhu, J. Bu, C. Chen, A survey of human pose estimation: the body parts parsing based methods, *J. Visual Commun. Image Represent.*, **32** (2015), 10–19. http://dx.doi.org/10.1016/j.jvcir.2015.06.013

21. W. Gong, X. Zhang, J. Gonzàlez, A. Sobral, T. Bouwmans, C. Tu, et al., Human pose estimation from monocular images: a comprehensive survey, *Sensors*, **16** (2016), 1966. http://dx.doi.org/10.3390/s16121966

22. P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, *Int. J. Comput. Vision*, **61** (2005), 55–79. http://dx.doi.org/10.1023/B:VISI.0000042934.15159.49

23. S. Qiao, Y. Wang, J. Li, Real-time human gesture grading based on openpose, in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017, 1–6. http://dx.doi.org/10.1109/CISP-BMEI.2017.8301910

24. D. Osokin, Real-time 2d multi-person pose estimation on cpu: lightweight openpose, preprint, arXiv:1811.12004.

25. N. Nakano, T. Sakura, K. Ueda, L. Omura, A. Kimura, Y. Iino, et al., Evaluation of 3d markerless motion capture accuracy using openpose with multiple video cameras, *Front. Sports Active Living*, **2** (2020), 50. http://dx.doi.org/10.3389/fspor.2020.00050

26. W. Chen, Z. Jiang, H. Guo, X. Ni, Fall detection based on key points of human-skeleton using openpose, *Symmetry*, **12** (2020), 744. http://dx.doi.org/10.3390/sym12050744

27. C. B. Lin, Z. Dong, W. K. Kuan, Y. F. Huang, A framework for fall detection based on openpose skeleton and lstm/gru models, *Appl. Sci.*, **11** (2020), 329. http://dx.doi.org/10.3390/app11010329

28. A. Viswakumar, V. Rajagopalan, T. Ray, C. Parimi, Human gait analysis using openpose, in *2019 Fifth International Conference on Image Information Processing (ICIIP)*, IEEE, 2019, 310–314. http://dx.doi.org/10.1109/ICIIP47207.2019.8985781

29. D. Yang, M. M. Li, H. Fu, J. Fan, H. Leung, Centrality Graph Convolutional Networks for skeleton-based action recognition, preprint, arXiv:2003.03007.

30. M. Fanuel, X. Yuan, H. N. Kim, L. Qingge, K. Roy, A survey on skeleton-based activity recognition using Graph Convolutional Networks (GCN), in *2021 12th International Symposium on Image and Signal Processing and Analysis (ISPA)*, 2021, 177–182. http://dx.doi.org/10.1109/ISPA52656.2021.9552064

31. Z. Hu, E. J. Lee, Dual attention-guided multiscale dynamic aggregate Graph Convolutional Networks for skeleton-based human action recognition, *Symmetry*, **12** (2020), 1589. http://dx.doi.org/10.3390/sym12101589

32. L. Zhao, X. Peng, Y. Tian, M. Kapadia, D. N. Metaxas, Semantic Graph Convolutional Networks for 3d human pose regression, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 3425–3435. http://dx.doi.org/10.1109/CVPR.2019.00354

33. M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, Q. Tian, Actional-structural Graph Convolutional Networks for skeleton-based action recognition, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 3595–3603. http://dx.doi.org/10.1109/CVPR.2019.00371

34. K. Thakkar, P. Narayanan, Part-based Graph Convolutional Network for action recognition, preprint, arXiv:1809.04983.

35. M. Li, S. Gao, F. Lu, K. Liu, H. Zhang, W. Tu, Prediction of human activity intensity using the interactions in physical and social spaces through Graph Convolutional Vetworks, *Int. J. Geog. Inf. Sci.*, **35** (2021), 2489–2516. http://dx.doi.org/10.1080/13658816.2021.1912347

36. W. Liu, S. Fu, Y. Zhou, Z. J. Zha, L. Nie, Human activity recognition by manifold regularization based dynamic Graph Convolutional Networks, *Neurocomputing*, **444** (2021), 217–225. http://dx.doi.org/10.1016/j.neucom.2019.12.150

37. M. Korban, X. Li, Ddgcn: a dynamic directed Graph Convolutional Network for action recognition, in *European Conference on Computer Vision*, 2020, 761–776. http://dx.doi.org/10.1007/978-3-030-58565-5_45

38. F. Manessi, A. Rozza, M. Manzo, Dynamic Graph Convolutional Networks, *Pattern Recognit.*, **97** (2020), 107000. http://dx.doi.org/10.1016/j.patcog.2019.107000

39. R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, et al., Graph Convolutional Networks for temporal action localization, in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, 7094–7103. http://dx.doi.org/10.1109/ICCV.2019.00719

40. H. Yang, D. Yan, L. Zhang, Y. Sun, D. Li, S. J. Maybank, Feedback Graph Convolutional Network for skeleton-based action recognition, *IEEE Trans. Image Process.*, **31** (2021), 164–175. http://dx.doi.org/10.1109/TIP.2021.3129117

41. J. Sanchez, C. Neff, H. Tabkhi, Real-world Graph Convolution Networks (rw-gcns) for action recognition in smart video surveillance, in *2021 IEEE/ACM Symposium on Edge Computing (SEC)*, 2021, 121–134. https://doi.org/10.1145/3453142.3491293

42. L. Feng, Q. Yuan, Y. Liu, Q. Huang, S. Liu, Y. Li, A discriminative stgcn for skeleton oriented action recognition, in *International Conference on Neural Information Processing*, 2020, 3–10. http://dx.doi.org/10.1007/978-3-030-63823-8_1

43. T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, et al., Microsoft coco: common objects in context, in *European Conference on Computer Vision*, 2014, 740–755. https://doi.org/10.1007/978-3-319-10602-1_48

44. M. Andriluka, L. Pishchulin, P. Gehler, B. Schiele, 2D human pose estimation: new benchmark and state of the art analysis, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, 3686–3693. http://dx.doi.org/10.1109/CVPR.2014.471

45. S. Johnson, M. Everingham, Clustered pose and nonlinear appearance models for human pose estimation, in *Proceedings of the British Machine Vision Conference*, 2010, 12.1–12.11. http://dx.doi.org/10.5244/C.24.12

46. B. Sapp, B. Taskar, Modec: multimodal decomposable models for human pose estimation, in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, 3674–3681. http://dx.doi.org/10.1109/CVPR.2013.471

47. M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, A. Milan, J. Gall, et al., Posetrack: a benchmark for human pose estimation and tracking, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, 5167–5176. http://dx.doi.org/10.1109/CVPR.2018.00542

48. J. Wu, H. Zheng, B. Zhao, Y. Li, B. Yan, R. Liang, et al., Large-scale datasets for going deeper in image understanding, in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 2019, 1480–1485. http://dx.doi.org/10.1109/ICME.2019.00256

49. W. Mao, Y. Ge, C. Shen, Z. Tian, X. Wang, Z. Wang, Tfpose: direct human pose estimation with transformers, preprint, arXiv:2103.15320.

50. Y. Abouelnaga, H. M. Eraqi, M. N. Moustafa, Real-time distracted driver posture classification, preprint, arXiv:1706.09498.

51. K. Simonyan, A. Vedaldi, A. Zisserman, Deep Inside Convolutional Networks: visualising image classification models and saliency maps, preprint, arXiv:1312.6034.

52. K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for large-scale image recognition, preprint, arXiv:1409.1556.

53. M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, et al., The history began from alexnet: a comprehensive survey on deep learning approaches, preprint, arXiv:1803.01164.

54. Q. Zhang, Y. N. Wu, S. C. Zhu, Interpretable Convolutional Neural Networks, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, 8827–8836. http://dx.doi.org/10.1109/CVPR.2018.00920

55. G. Hidalgo, Y. Raaj, H. Idrees, D. Xiang, H. Joo, T. Simon, et al., Single-network whole-body pose estimation, preprint, arXiv:1909.13423.

56. A. Neubeck, L. Van Gool, Efficient non-maximum suppression, in *18th International Conference on Pattern Recognition (ICPR'06)*, 2006, 850–855. http://dx.doi.org/10.1109/ICPR.2006.479

57. L. Cai, B. Zhao, Z. Wang, J. Lin, C. S. Foo, M. S. Aly, et al., Maxpoolnms: getting rid of NMS bottlenecks in two-stage object detectors, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 9356–9364. http://dx.doi.org/10.1109/CVPR.2019.00958

58. S. Goswami, Reflections on Non-Maximum Suppression (NMS), 2020.

59. D. Wang, C. Li, S. Wen, Q. L. Han, S. Nepal, X. Zhang, et al., Daedalus: breaking nonmaximum suppression in object detection via adversarial examples, *IEEE Trans. Cybern.*, http://dx.doi.org/10.1109/TCYB.2020.3041481

60. I. Ahmed, M. Ahmad, A. Ahmad, G. Jeon, Top view multiple people tracking by detection using deep sort and yolov3 with transfer learning: within 5g infrastructure, *Int. J. Mach. Learn. Cybern.*, **12** (2021), 3053–3067, http://dx.doi.org/10.1007/s13042-020-01220-5

61. N. Wojke, A. Bewley, D. Paulus, Simple online and realtime tracking with a deep association metric, in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, 3645–3649. http://dx.doi.org/10.1109/ICIP.2017.8296962

62. S. Challa, M. R. Morelande, D. Mušicki, R. J. Evans, *Fundamentals of Object Tracking*, Cambridge University Press, 2011. http://dx.doi.org/10.1017/CBO9780511975837

63. A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, *ACM Comput. Surv. (CSUR)*, **38** (2006). http://dx.doi.org/10.1145/1177352.1177355

64. H. Fan, M. Jiang, L. Xu, H. Zhu, J. Cheng, J. Jiang, Comparison of long short term memory networks and the hydrological model in runoff simulation, *Water*, **12** (2020), 175. http://dx.doi.org/10.3390/w12010175

65. A. Agarwal, S. Suryavanshi, Real-time* multiple object tracking (mot) for autonomous navigation, *Tech. Rep.* Available from: http://cs231n.stanford.edu/reports/2017/pdfs/630.pdf.

66. D. P. Kingma, J. Ba, Adam: a method for stochastic optimization, preprint, arXiv:1412.6980.

67. J. Teow, Understanding kalman filters with python, 2017.

68. J. Song, L. Wang, L. Van Gool, O. Hilliges, Thin-slicing network: a deep structured model for pose estimation in videos, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, 4220–4229. http://dx.doi.org/10.1109/CVPR.2017.590

69. Y. Luo, J. Ren, Z. Wang, W. Sun, J. Pan, J. Liu, et al., Lstm pose machines, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, 5207–5215. http://dx.doi.org/10.1109/CVPR.2018.00546