Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 2449603, 12 pages
https://doi.org/10.1155/2021/2449603

Research Article

Robust Suspicious Action Recognition Approach Using Pose Descriptor

Waqas Ahmed,[1] Muhammad Haroon Yousaf,[2,3] and Amanullah Yasin[3]

[1] *Department of Telecommunication Engineering, University of Engineering and Technology, Taxila, Pakistan*
[2] *Department of Computer Engineering, University of Engineering and Technology, Taxila, Pakistan*
[3] *Swarm Robotic Lab-National Centre for Robotics and Automation (NCRA), University of Engineering and Technology, Taxila, Pakistan*

Correspondence should be addressed to Muhammad Haroon Yousaf; haroon.yousaf@uettaxila.edu.pk

Received 25 April 2021; Revised 11 July 2021; Accepted 3 August 2021; Published 13 August 2021

Academic Editor: Muazzam Maqsood

In the current era of technological development, human actions can be recorded in public places like airports, shopping malls, and educational institutes, etc., to monitor suspicious activities like terrorism, fighting, theft, and vandalism. Surveillance videos contain adequate visual and motion information for events that occur within a camera's view. Our study focuses on the concept that actions are a sequence of moving body parts. In this paper, a new descriptor is proposed that formulates human poses and tracks the relative motion of human body parts along with the video frames, and extracts the position and orientation of body parts. We used Part Affinity Fields (PAFs) to acquire the associated body parts of the people present in the frame. The architecture jointly learns the body parts and their associations with other body parts in a sequential process, such that a pose can be formulated step by step. We can obtain the complete pose with a limited number of points as it moves along the video and we can conclude with a defined action. Later, these feature points are classified with a Support Vector Machine (SVM). The proposed work was evaluated on the benchmark datasets, namely, UT-interaction, UCF11, CASIA, and HCA datasets. Our proposed scheme was evaluated on the aforementioned datasets, which contained criminal/suspicious actions, such as kick, punch, push, gun shooting, and sword-fighting, and achieved an accuracy of 96.4% on UT-interaction, 99% on UCF11, 98% on CASIA and 88.72% on HCA.

1. Introduction

Government and security institutions install surveillance cameras in homes, markets, hospitals, shopping malls, and public places to capture real-time events to ensure the safety of people. In the threat-laden context of vandalism, terrorism, or suspicious activities, the surveillance videos are of the utmost necessity for any incident investigation. These threatening situations highlight the critical need to develop a suspicious action recognition system to aid forensic experts in capturing criminals and resolving their criminal investigations. The concept of action recognition encompasses around detection, understanding, and classification of a simple action like clapping, walking, meetings, etc. In recent years, scholars started an investigation of actions in a complex environment like sports. Now, for criminal actions, crime can be defined as an action harmful to any individual, community, or society. They can be differentiated into many forms like homicide, robbery, burglary, and cybercrime, etc. Criminal actions are less studied and we can hardly find any dataset which provides substantial criminal actions. The interaction of potential victim and offender makes a criminal action. The motivation of the offender decreases when he is conscious of being watched [1]. Criminal actions are comparatively different from a person's regular activities. Criminal actions are generally those actions where an individual may harm other individuals, society, or the public. Criminal actions are unique, as threatening human gestures, poses, and activities are very different compared to other normal actions, which makes them difficult to recognize.

Human motion analysis is the most active research area in computer vision. Motion analysis can be divided into two different tasks. The first task is to describe the physical movements of the body parts, e.g., the raising of a hand or the turning of the head. Pose estimations and tracking of body parts are useful methods of identification. The second task is to describe the semantics of the movements, such as picking up an object or shaking hands.

Action recognition approaches require a large amount of data to process the actions, which requires computational power. For this reason, action recognition is receiving immense focus from the research society due to its considerable application. The action recognition process can generally be subdivided into preprocessing, feature extraction, feature encoding, and classification. For the feature extraction and encoding processes, there is a substantial area to explore, but the classification process is very mature. Currently, feature extraction is done either by a handcrafted process or by a learned features (deep) process. The most widespread feature extraction process in the category of handcrafted processes is the Histogram of Oriented Gradients (HOG) [2], the Histogram of Optical Flow (HOF) [3], the Motion Boundary Histograms (MBH) [4], and the Scale Invariant Feature Transform [5], etc. These descriptors use different methods for feature extraction from the various regions of video. Their methods include extracting features, such as interest points, dense samplings [6], and motion trajectories [7]. Recently, extracting features using deep neural networks has inspired new directions in the field and is achieving impressive results [8–10]. Feature encoding allows the translation of features into a feature space. Fisher Vectors [11] and Vector of Locally Aggregated Descriptors (VLAD) [12] are commonly used; such methodologies provide good performance for many solutions [8, 10, 13]. However, these encoding schemes lack spatiotemporal data, which is vital while dealing with videos. Another popular method [14], known as the “Bag of Expression (BOE)” model, provides an encoding solution by maintaining the spatiotemporal information. With the advancements in deep neural networks, the features of neural networks achieve better results compared to the result of handcrafted methods. The main advantage of deep features is that they provide higher discriminative powers on the top layers of the networks that are learned from low-level features. These features are transformed with deep neural layers, where handcrafted solutions mostly contain low-level information, such as edges and corners [9, 15–17]. Currently, three-dimensional poses can be extracted from monocular images or videos, where the human body is represented as a stick skeleton surrounded by surface-based (using polygons) or volumetric (using spheres or cylinders) flesh [18].

In the last few years, the researchers also explored the variants of action recognition with the help of suitable sensors [19]. Sensor based recognition is based on time series data collected from accelerometers either in mobile phones [20, 21] or wrist-worn [22, 23], magnetometers, and gyroscopes [24]. In these approaches, the raw data is acquired from the sensors, which are preprocessed and normalized. The features are extracted either using manual [25] or using CNN [26]. The time series data is segmented sequentially into smaller segments. Each segment is labeled based on the feature response in that segment. To analyze the time series, there are many parameters like Moving average (MA)/sliding window, autoregression (AR), Autoregressive Moving Average (ARMA) [27], etc. In our work, we have used Moving Averaging, as it models the next step in the sequence as a linear function. In our case, we only link the interest points present in the first frame with the next frames.

In recent years, pose estimation methods have become more complex and accurate. Many studies on pose estimation problems were concentrated on finding the body parts of a single actor in the image [28]. One approach to solving this problem, which is named a top-down approach [29], is to reduce the multiperson pose estimation to a single-person case. Therefore, a person detector is applied to the image, and then a single-person pose estimation is performed for images inside the bounding boxes of each detection. Examples of the systems that use a top-down approach include the work in [29–31]. However, this approach introduces several additional problems; the main problem is when a nonactor is detected as an actor. This increases errors in pose estimations of nonactors. Second, regions with detected persons may overlap and the body parts of different individuals, making it difficult for pose estimation algorithms to associate detected body parts with the corresponding actor.

Several multipeople bottom-up pose estimation systems were developed [28, 32] which used deep neural networks to achieve better performance of the pose estimation. Pose Estimation and action recognition can be done together as they are related to each other. Reference [33] used action recognition methods to build a pose estimation algorithm. Our approach utilizes the preprocessing steps employed in [28], as the idea is to extract the motion information of the human body parts. In our work, we have used pose estimation as a baseline for feature extraction. Our approach, convert the extracted frames from the video into a feature map. These feature maps are fed to the CNN network to provides the location of actor’s body part in the frame. The location of each part is stored as a feature representation and used to track the respective motion in the video frames. The performance of our approach depends on how accurately the human poses are estimated and linked with the associated body parts. In this study, the main research contributions are as follows:

(i) We have used the CNN network [28] for limbs extraction in a frame and used that information to further extract the limbs localization.

(ii) Additionally, the pose or skeleton is modified and restructured to get extra importance from the head and neck area, as, in suspicious actions, the head plays a vital role.

(iii) The extracted features from the previous and current frames are stored and served as guidance to temporally relate the motion. This will also help the descriptor to cope with situations where body parts are missing or occluded.

The rest of the paper is arranged as follows: in Section 2, we describe the proposed algorithm. In Section 3, we discuss the experiments and their results. The conclusions are summarized in Section 4.

2. Proposed Approach

Our approach is based on the pose estimations of actors in video clips. The proposed approach is given in Figure 1. Each step is elaborated in the sections below.

2.1. Feature Extraction. For feature extraction, the network [28] is used in such a way to extract the body parts and their association with each other to constitute a full skeleton (pose). The videos are decomposed into image frames and reshaped to $368 \times 654 \times 3$ to fit the GPU memory. Each frame is first analyzed through CNN (VGG-19, first 10 layers and fine-tuned), it generates feature maps $(\mathbf{F})$ which are input to the network. The network is divided into two parts with multiple stages. At stage 1, 6 convolutions layers of 3×3 and 2 Conv layers of 1×1 and max-pooling layer at the end of each stage. The feature maps from preprocessing stage (VGG-19 layers) are used as an input to both parts. The first part works as a feedforward network calculates a 2D confidence map $(\mathbf{S})$ and the second stage calculates the set of part affinities vectors $(\mathbf{L})$. At the end of each stage, the output of both the branches is concatenated along with the image feature maps. This process is iterative and successfully refines the predictions of the previous stage. This will provide the degree of relativity between the parts of one actor to another. $S = \{S_1, S_2, \ldots S_j\}$ has J confidence maps, one for each limb, and $L = \{L_1, L_2, \ldots L_c\}$ will have a C vector for each limb. Here, we have lowered down the matrices and extracted only 18 key points for one actor comprising 17 joints, which help us in achieving the lower dimension of feature vectors.

$$\mathbf{S}^{\mathbf{t}} = \boldsymbol{\rho}^{\mathbf{t}}\left(\mathbf{F}, \mathbf{S}^{\mathbf{t}-1}, \mathbf{L}^{\mathbf{t}-1}\right), \quad \forall \mathbf{t} \geq 2, \tag{1}$$

$$\mathbf{L}^{\mathbf{t}} = \boldsymbol{\varphi}^{\mathbf{t}}\left(\mathbf{F}, \mathbf{S}^{\mathbf{t}-1}, \mathbf{L}^{\mathbf{t}-1}\right), \quad \forall \mathbf{t} \geq 2. \tag{2}$$

Equations (1) and (2) represent the resultant confidence maps and affinity vector fields, respectively.

To fine-tune the network for precise detection of the body parts, loss functions are incorporated between the estimated predictions and ground truth. The loss functions [28] are as follows:

$$\begin{aligned} \mathbf{f}_{\mathbf{s}}^{\mathbf{t}} &= \sum_{\mathbf{j}=1}^{\mathbf{J}} \sum_{\mathbf{P}} \mathbf{W}(\mathbf{P}). \left\| \mathbf{S}_{\mathbf{j}}^{\mathbf{t}}(\mathbf{P}) - \mathbf{S}_{\mathbf{j}}^{*}(\mathbf{P}) \right\|_2^2, \\ \mathbf{f}_{\mathbf{L}}^{\mathbf{t}} &= \sum_{\mathbf{c}=1}^{\mathbf{C}} \sum_{\mathbf{P}} \mathbf{W}(\mathbf{P}). \left\| \mathbf{L}_{\mathbf{c}}^{\mathbf{t}}(\mathbf{P}) - \mathbf{L}_{\mathbf{c}}^{*}(\mathbf{P}) \right\|_2^2, \end{aligned} \tag{3}$$

where $(\mathbf{S}^*)$ and $(\mathbf{L}^*)$ represent the ground-truth confidence map and relative affinity field, respectively. W (P) is the window function that gives zero output in the case where the annotation is missing in the image location P. This whole process is pictorially represented in Figure 2. (a) The network takes images as input, (b) calculates the confidence maps for each limb, (c) the second stage periodically find the part affinity field, (d) information from b and c is used to join the relative body parts for each candidate, and (e) assemble them into the full-body pose.

2.2. Formulation of Feature Vector. The extracted features are the combination of body parts in the form of affinity vectors and the confidence maps as discussed above. We calculated the body parts which can work together to perform the motion. The movement or orientation of joints can also help in forming the action. We decomposed the body parts with the help of affinity vectors and encoded them in a set of 18 key points associated with "joints" and 17 lines connecting the joints are associated with "limbs" of the body. Our main aim is to capture the motion of each limb and joint as a vector of coordinates location and its orientation. The coordinates provide the location of each limb in each frame/image and the orientation encapsulates the direction of motion.

We encode each limb with its x and y locations for the position and orientation of limbs in each frame with 34 points (17 for both x and y coordinates). We make a set of 68 points that is successive points in the two consecutive frames. To track the movement of each limb separately, the formed skeleton and the calculated coordinates are shown in Figure 3. Our approach is robust because of the very low count of interest points per frame as compared to recent approaches [8, 13, 33–35]. This representation allows encoding both position and orientation of the body parts. To take into account possible differences in the frame sizes, we use coordinates related to the center instead of the usual image coordinates (the center of the frame is 0). An example of the visual representation from a single frame is shown in Figure 4. It is also important to note that the pose estimation approach sometimes will not be able to extract full pose from the frame and there might be different numbers of extracted poses on different frames, mainly due to occlusion.

2.3. Action Formulation with Time Series. The next task is to combine the information at each frame with the help of joints and limbs to form an action. We get results only for each frame individually. It is important to connect them to get continuous movement individually. For each actor, we compute a centroid of all computed points as a point for comparison. For consecutive frames, estimations are connected if their centroids are closer to the frames. For the formulation of actions, few things might happen, like:

(i) Partial occlusion or self-occlusion of the body parts: In this case, pose estimation cannot produce all the key points and only partial information about the pose can be obtained.

(ii) Disappearing from the frame: Detected actors may disappear from the camera view during the video segment.

(iii) Incorrect pose estimation: The pose can be incorrectly detected, and it does not belong to any actor in the video segment. An example of such misdetection is shown in Figure 5.

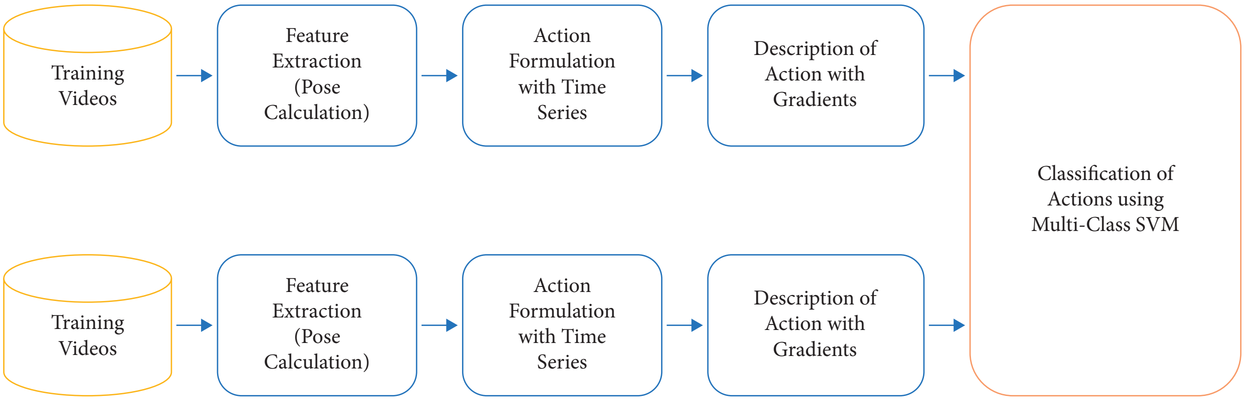

Training Videos
Feature Extraction (Pose Calculation)
Action Formulation with Time Series
Description of Action with Gradients
Training Videos
Feature Extraction (Pose Calculation)
Action Formulation with Time Series
Description of Action with Gradients
Classification of Actions using Multi-Class SVM

Figure 1: Block diagram of the proposed approach.

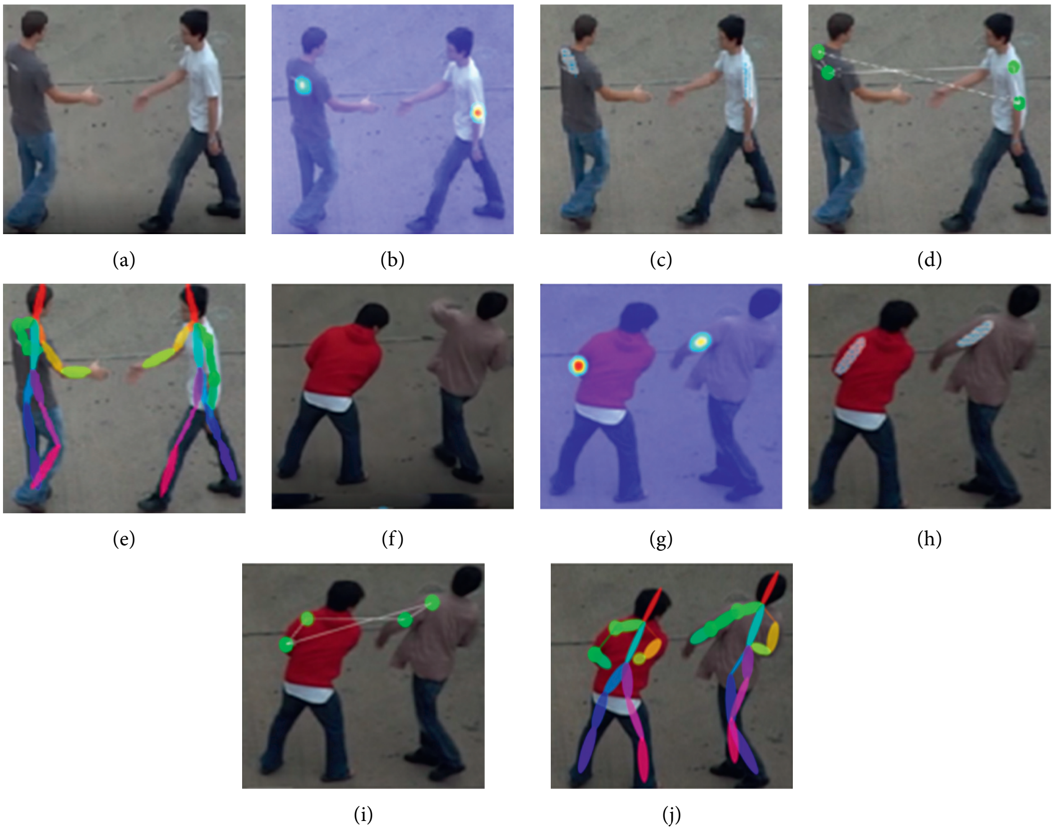
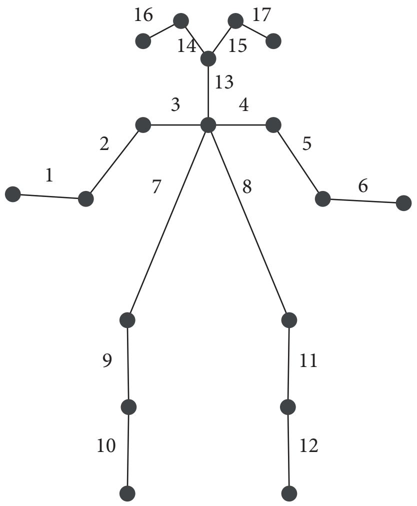


Figure 2: Overall Process of feature extraction in steps.

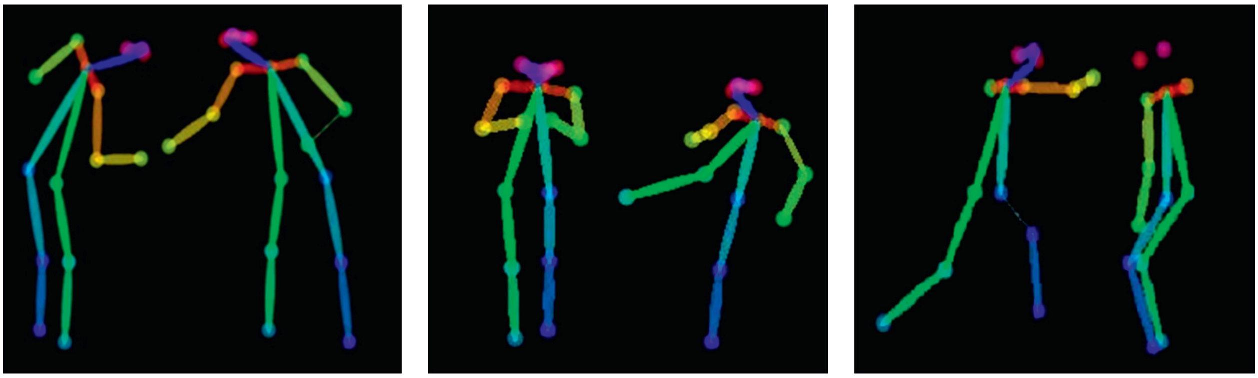
To construct a time series [36] to track the motion of the actors in the video, we start with the first frame and compute the number of interest points present in the frame. If there is only one actor, then we will have 17 points and their co-ordinate values, which will be used in the next frame to track the motion of an actor. Similarly, in the next frames, we will use the previous coordinates and current location of our interest points (joints) which makes the total of 68 points in the consecutive frames. This process will continue until all the frames are completed. Now, after the completion of actions, they are evaluated by comparing them with the original input videos. We first checked the average motion in the video of the original video and then performed the same operation on our time series-based feature extracted video as shown in Figure 6. The upper part depicts the average motion of three videos comprising different actions, and the lower part of the figure contains the average motion of the same video but with the help of feature extracted points. Here, we can predict the action by looking at the lower portion of the figure and the environmental noise is removed as our descriptor is only dependant on the body parts and their relative motion.

The descriptor is further evaluated by calculating the Motion History Image (MHI) [37], it represents the motion



1 - points P1: 13 (x), 14 (y); P2: 15 (x), 16 (y)
2 - points P1: 9 (x), 10 (y); P2: 11 (x), 12 (y)
3 - points P1: 1 (x), 2 (y); P2: 3 (x), 4 (y)
4 - points P1: 5 (x), 6 (y); P2: 7 (x), 8 (y)
5 - points P1: 17 (x), 18 (y); P2: 19 (x), 20 (y)
6 - points P1: 21 (x), 22 (y); P2: 23 (x), 24 (y)
7 - points P1: 25 (x), 26 (y); P2: 27 (x), 28 (y)
8 - points P1: 37 (x), 38 (y); P2: 39 (x), 40 (y)
9 - points P1: 29 (x), 30 (y); P2: 31 (x), 32 (y)
10 - points P1: 33 (x), 34 (y); P2: 35 (x), 36 (y)
11 - points P1: 41 (x), 42 (y); P2: 43 (x), 44 (y)
12 - points P1: 45 (x), 46 (y); P2: 47 (x), 48 (y)
13 - points P1: 49 (x), 50 (y); P2: 51 (x), 52 (y)
14 - points P1: 53 (x), 54 (y); P2: 55 (x), 56 (y)
15 - points P1: 61 (x), 62 (y); P2: 63 (x), 64 (y)
16 - points P1: 57 (x), 58 (y); P2: 59 (x), 60 (y)
17 - points P1: 65 (x), 66 (y); P2: 67 (x), 68 (y)

Figure 3: Stick figure model of a human body with points and lines representing joints and limbs.



(a) (b) (c)

Figure 4: Visualisation of the results of the pose estimation algorithm on a single frame of the video. (a) Shaking of hands of actors. (b) Kicking action. (c) Punching.

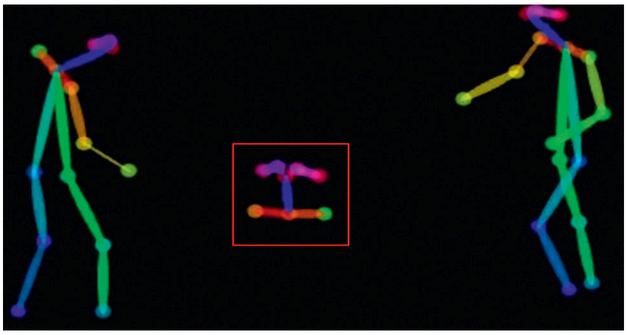
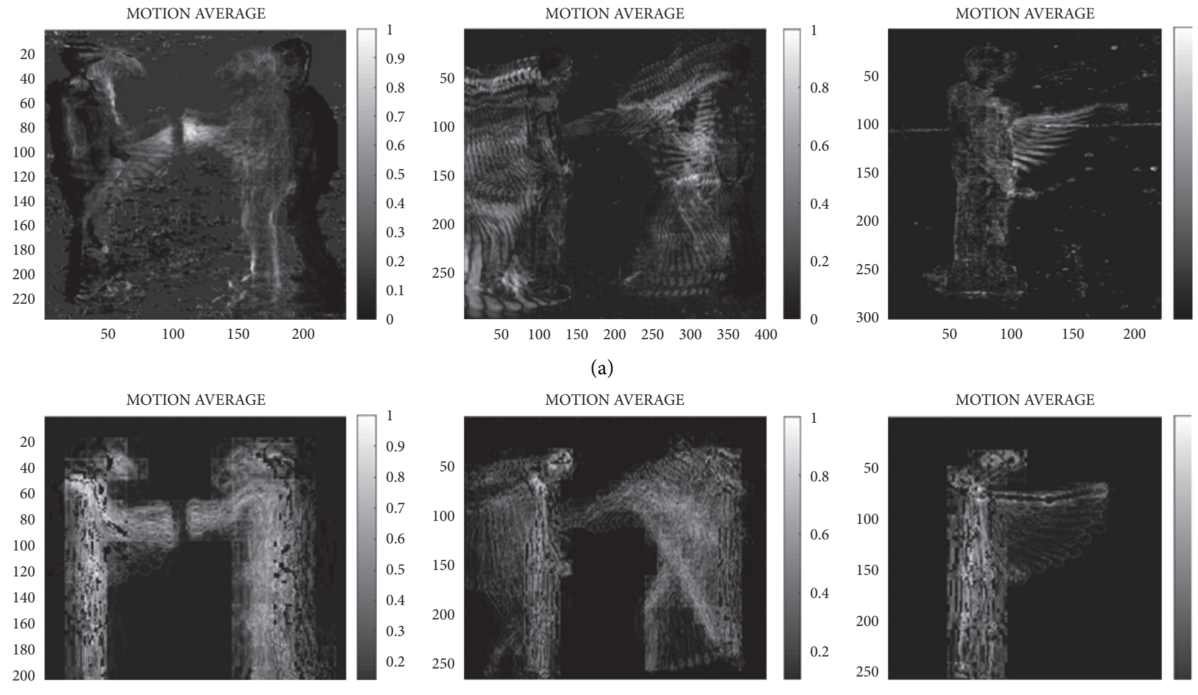


Figure 5: Example of False detection.

or movement in the single static template. It highlights the location and path of the motion along with the frames and represents the motion history at a particular location. The brighter values signify the most current motion. MHI is widely used in video processing, where video sequence is decomposed into a single image. It contains the motion flow and moving parts of the action video. The MHI of the original video and feature extracted video can be depicted in Figure 7. The upper portion of the figure represents the original videos of different actions and the lower portion shows the MHI of the feature extracted video. The features only contained the motion of moving body parts and did not contain any other information. The MHI of extracted video only contains the relative information of actions happening in the video.

Next, we calculate a gradient of the values in the time series. The segments with the high magnitude of the gradient will indicate that an actor performed a lot of movements at this point. On the other hand, the segments with the gradient close to zero will indicate a part of the video when an actor remained still. It will allow us to remove the part of the video at the beginning and the end of the video segments when actors did not perform any actions leaving only the localized part of the video. It helps in more accurate action segmentation in the video. The average gradient can be visualized in Figure 8 and the following equation:

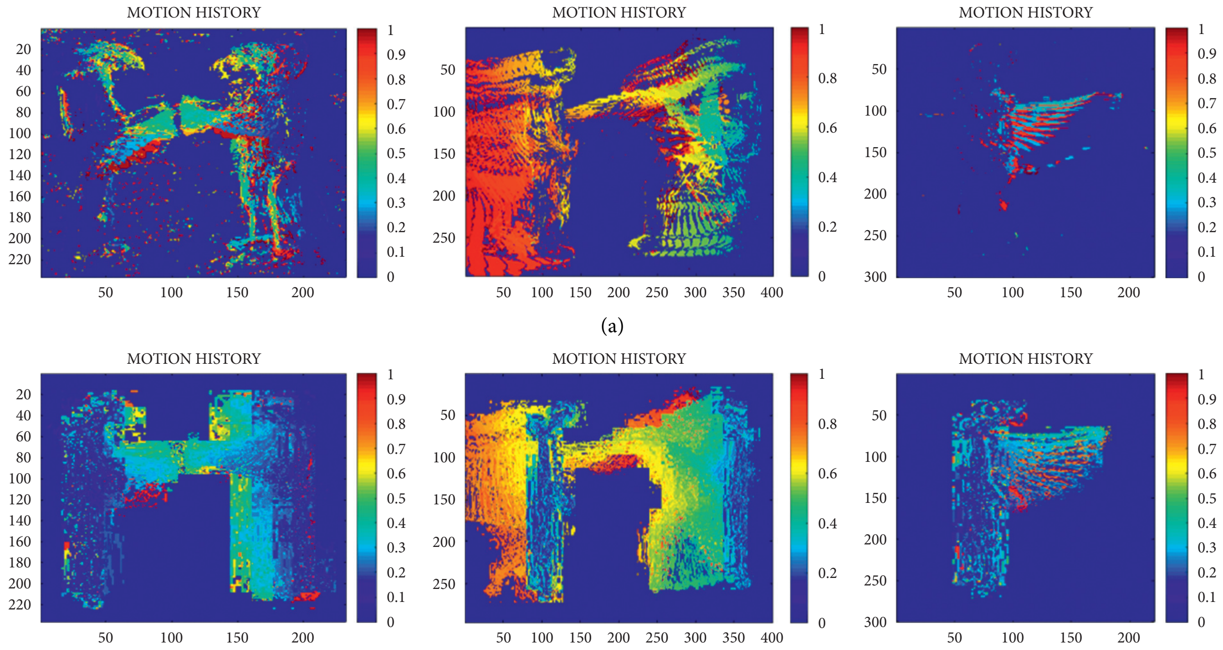
$$\mathbf{D}_n = \left(\sum_{\mathbf{n}}\left(\frac{1}{\mathbf{t}}\right) * \sum_{\mathbf{t}} \nabla\left(\mathbf{S}_{\mathbf{t}}, \mathbf{L}_{\mathbf{t}}\right)\right), \qquad (4)$$



(a)

(b)

Figure 6: Depiction of Average Motion. The motion-captured from the original videos (a). Motion Captured after the feature extraction (b).



(a)

(b)

Figure 7: Motion History Image (MHI). MHI computed from the original dataset videos (a). MHI taken with the help of extracted features (b).

where t is the total number of interest points present in frames and D signifies the ensemble of interest point coordinates and orientations along n (no. of frames). The frame with very low average gradients depicts either no motion or very small motion. We can exclude a couple of frames, which not only improves the classification but also reduces computation time.

The resultant Vector $D = \{D_1, D_2, \ldots, D_n\}$ represents the motion or actions in the video. We combined the descriptor along the temporal axis to make a time series to

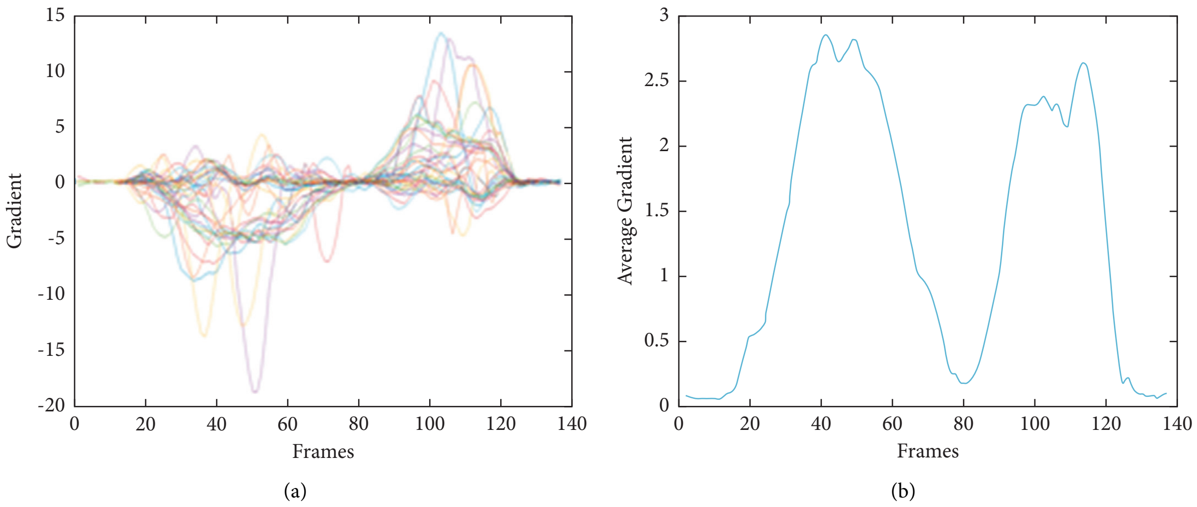


Figure 8: (a) Gradients of interest points along with each frame. (b) Average gradient along frames.

analyze the action more discreetly. In case of any missing frames or key points occluded by objects, we check if the part of the body is not visible all the time. It is removed from the consideration. For the rest, we fill small gaps with the closest previous or future value. We apply the Savitsky-Golay filter [38] to exclude random spikes and get smooth time series for the movement. This filter increased the precision without altering any interest points and as a result, we have the vector of 68 gradients of interest points with the total number of frames in the video.

2.4. Action Classification with Proposed Descriptor. We have calculated feature vectors from each video, as shown in Figure 9; the extracted features showing the human movement along the frames. We trained a classifier that will distinguish between different actions in the video. We used a set of SVM classifiers [39] for each class of actions with varying sigmoid kernel functions to best fit our featured data. The performance of the SVM classifier for the best kernel function can be seen from Figure 10. The graph clearly represents the kernel function with a 0.5 value gives better results for the UT-interaction dataset. Each classifier will estimate the probability that the action performed in the analyzed video belongs to the specific category.

Classifiers for the first class of actions are trained on the entire training set with two labels, “first action” and “not first action,” assigned to the video segments. Then, the video segments are excluded from the dataset, and the classifier for the second class is trained on the remaining data with labels, “second action,” and “not second action” and so on. In the case of the N class of actions, there will be $N-1$ classifiers, and the last classifier will distinguish between actions “$N-1$” and “N.” Additionally, we use a sequential feature selection technique for each classifier to reduce the number of predictors used in the classification. This will allow us to use only information about the movements of the body parts that are the most relevant for this class of actions.

3. Experiments and Discussion

We evaluated the proposed method using a leave-one-out cross-validation technique and a k-fold (10-fold) validation. A classifier, as described above, is trained using the training set and used to predict action in the video from the prediction set. The procedure is repeated for each of “N” videos in the dataset, and the results of all the predictions are collected and represented in the form of a confusion matrix.

3.1. Datasets. The algorithm was assessed on four action datasets: the UT-Interaction dataset [40], the YouTube action dataset [41], the CASIA dataset [42], and HCA [43].

The UT-Interaction dataset has been considered a standard for human interaction. The dataset contains six classes of two-actor interactions, which include handshaking, hug, kick, point, punch, and push.

The YouTube action dataset (also known as UCF11) contains 11 action categories with about a hundred video segments in each category and a total of 1595 videos. Most of these videos are extracted from YouTube and contain camera motion, cluttered backgrounds, different appearances of the actors, different illumination conditions, and viewpoints of the camera.

The CASIA action database contains various activities captured in an outdoor environment with different angles of view. The database contains various types of human actions (walking, running, bending, etc.). There are seven types of interaction involving two actors (robbing, fighting, follow-always, follow-together, meeting-apart, meeting-together, and overtaking). We selected only interaction videos for action recognition with respect to suspicious activities.

HCA dataset contains video sequences for six criminal actions. Each category contains a set of videos to depict a particular action. Each video is recorded with different actors performing various actions under numerous environmental conditions. There are 302 videos in total. Actions include fight, kick, push, punch, gun shooting, and sword-fighting.

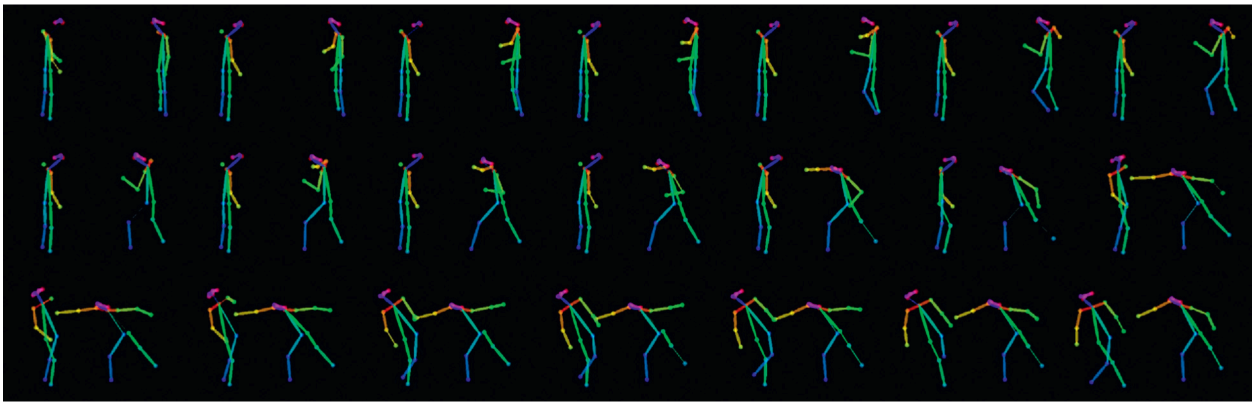


Figure 9: Formulation of action with the help of extracted features.

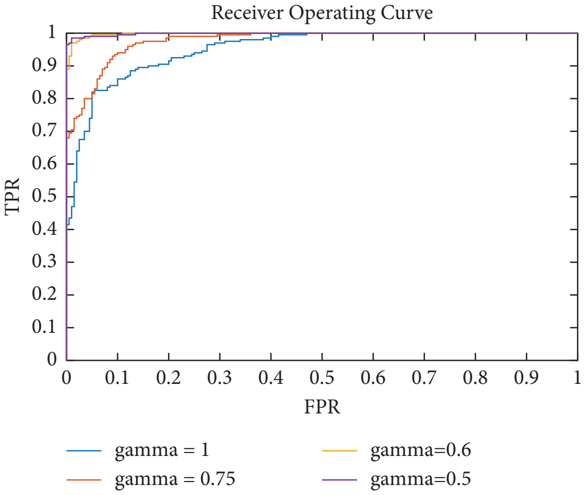


Figure 10: ROC curve for classification.

3.2. Results and Comparison with Current Approaches. The proposed approach is evaluated on four datasets. The system used for the simulation was Intel Xeon Processor, 32 GB RAM, and NVIDIA GTX1660 graphics.

The runtime performance of our approach depends on the number of people present in the video. The number of interest points (body parts) increased with people's count, which is the main factor in calculating the time complexity. The runtime complexity comprises two major parts: one CNN processing time with complexity is O (1), constant with a variable number of people; second, multiperson analyzing time, which takes the complexity of O (n^2), where n represents the number of actors present in the video. The rest is time series approach, whose complexity is O (n).

All four datasets contained a different variety of actions, which helped in better assessing the performance of the proposed approach. The first dataset processed was the UT-interaction dataset with a total number of 20 videos (10 in Set 1 and 10 in Set 2); set 1 contained actions with static backgrounds, which generated close to zero noise, and set 2 videos were captured in a more natural environment with multiple actors in the background. We used the leave-one-out validation and the 10-fold cross-validation separately to train the model, and the confusion matrix is shown in Figure 11. Our approach outperforms the state-of-the-art techniques. The recognition rate for Set 1 was 96.4%. The reason for this high accuracy is that our descriptor extracts the most relevant information about the actions performed, and the full body of actors is visible with minimal occlusions. Similarly, for Set 2, the same results were achieved, regardless of the environmental effects, our approach extracts the motion of the body parts, and it is the least affected by the environmental changes or occlusion. Table 1 shows the action-wise accuracies for each action, which shows good performance on all the actions except few misclassifications in the push and punch actions. The reason for misclassification is due to interclass similarity. The comparisons of the approaches with the state-of-the-art methods; our approach improved the accuracy by 1%, as shown in Table 2.

Our approach successively classifies the Shakehand, Hug, kick, and point actions as all the actions have uniquely defined movement of body parts, which clearly differ from other actions, but for the case of punch and push, we can see few misclassifications as both the actions comprise of similar movement. For example, for the actions of push and punch, one actor remains still, while the other actor approach and executes the action, which results in an impact on the first actor and he leans back (with the effect of push or punch). The hand movement of both actions is quite similar in few cases, which causes misclassification in few cases.

Next, we evaluated the proposed approach on the UCF11 dataset, which contained 11 actions, and most of the videos are taken in a realistic environment, so this dataset was quite challenging with respect to UT interactions due to large viewpoint variations, backgrounds, camera motions, and object appearances. Different body parts remained unseen for a fraction of time, as this dataset contains different viewpoints. Table 2 shows the performance comparisons for the UCF11 dataset with other state-of-the-art. Our approach outperforms because we first detected the poses and shaped the actions by joining the body parts together in the temporal domain. Here, in this dataset, we have multiple occlusions where actor movements are overlapping with other actors, which can cause misclassification. Figure 11(e) shows the confusion matrix for the UCF11 dataset, which

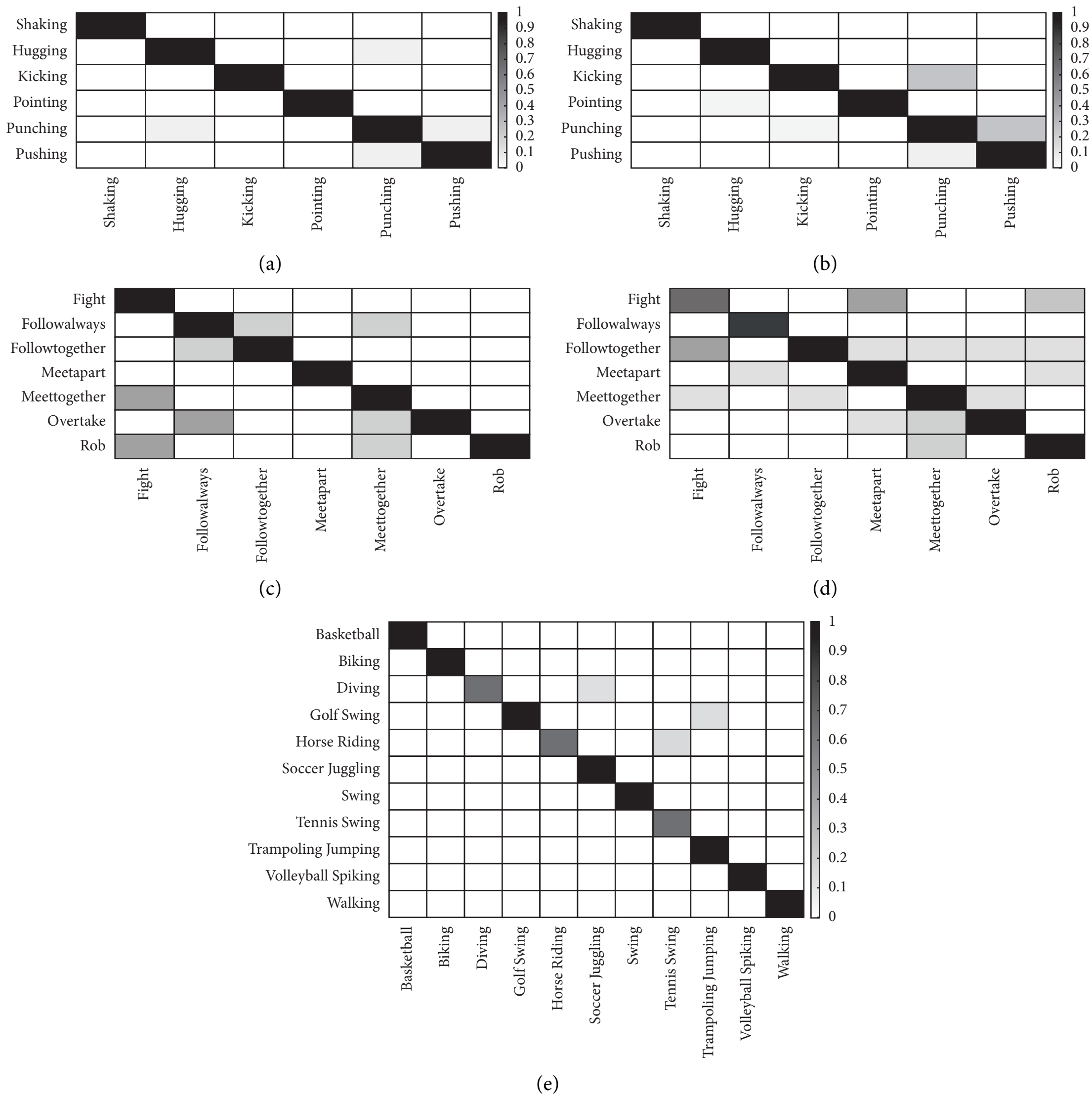


Figure 11: Confusion Matrices of (a) UT-Interaction (Set 1), (b) UT-Interaction (Set 2), (c) CASIA (Horizontal), (d) CASIA (Vertical), and (e) UCF11 dataset.

Table 1: Classwise accuracy on UT-interaction.

Dataset	Action	Accuracy
UT-interaction	Shake hand	96%
	Hug	97.46%
	Kick	94.35%
	Point	100%
	Punch	92.12%
	Push	93.80%

represents few misclassifications. The reason for misclassification is specifically in spiking and basketball shooting where the actor information is occluded and hand movement is overlapped with other actors. Therefore, it is crucial not to lose the body parts for a long-time. To overcome this issue, we picked the centroid of an actor and followed its motion relative to other body parts so that we do not lose the action attributes. As we are extracting the motion of the body parts, the background jitter did not have much of an effect.

For the CASIA dataset, there are three viewpoints, the angle view, the horizontal view, and the top view. The horizontal and angle viewpoints are better than the vertical (bird-eye) viewpoints. We have picked interaction videos to test the performance of our approach. Figure 11 shows the confusion matrices for horizontal (c) and vertical (d). Our approach requires the visualization of body parts for the maximum duration to extract the information about the action. In the vertical viewpoints, most of the actions look comparatively similar, and most of the body parts are hidden, so the transformation of the pose into motion will cause misclassifications as only the head, shoulder, and arms are highlighted for a very small period.

Our approach performed best for horizontal viewpoint and achieved an accuracy of 98%. Table 3 shows the activity-wise accuracies. The results only show the horizontal view as

Table 2: Performance comparison of the proposed approach with the state-of-the-art.

Dataset	**Paper**	**Results**		
		LOOV set 1	LOOV set 2	10-fold
UT-interaction	**Proposed method**	**96.4%**	**96%**	**93% (average)**
	Afrasiabi et al. [35]	93%	93%	—
	Ahmad Jalal et al. [34]	88%	87%	—
	Vahdat et al. [44]	93.3%	91.3%	—
	Ngoc et al. [45]	95%	96%	—
UCF11	**Proposed method**	**99.1%**		**98%**
	Cheng [46]	98.30%		—
	Yadav et al. [47]	91.30%		—
	Wang et al. [48]	98.7%		—
	Nazir et al. [14]	96.68%		—
	Amin Ullah [49]	97.17%		
CASIA	**Proposed method**	**98%**		**95%**
	Tian et al. [42]	94%		—
	Abinta et al. [43]	—		73.33%
HCA	**Proposed method**	**88.72%**		**86.28%**
	Abinta et al. [43]			80.79%

our approach performs best where most of the body part is visible in the entire video. The actions in this dataset are different from one another. Therefore our approach does not suffer from interclass similarity issues as we faced in UT-interaction (push and punch).

However, few misclassifications were observed in the “following” and “overtaking” actions where actors were overlapped in most of the frames and the approach was not able to classify the exact action. For the vertical viewpoint, most of the body cannot be seen due to camera position. Therefore, we can see many misclassifications. To better understand the actions in the vertical viewpoints and similar cases, optical flow and trajectory information was helpful for the translation of the poses into actions.

The last dataset selected for evaluation is Hybrid Criminal Action (HCA) dataset. This dataset comprised videos from different datasets; push, punch, and kick action videos were taken from the UT-interactions dataset; gun shooting and sword-fighting action videos were taken from HMDB51, and fight action videos were taken from the CASIA dataset. The overall experimental parameters change for each criminal action. Therefore, each of the actions is calculated separately and the average value is shown in Table 2.

Table 3: Acitivty-wise accuracy on CASIA (horizontal).

Dataset	Actions	Accuracy
CASIA	Fight	99.2%
	Follow always	97.33%
	Followtogether	95.55%
	Meet apart	98.61%
	Meet together	100%
	Overtake	99.78%
	Rob	96.88%

Table 4: Classwise accuracies on HCA dataset.

Dataset	Actions	Accuracy
HCA	Fight	96%
	Kick	99.28%
	Push	93.45%
	Punch	91.22%
	Gun-fighting	78.4%
	Sword-fighting	76.21%

Previously the proposed approach is evaluated separately on each of the actions and per-class accuracy for a fight, kick, push, punch, gun-fighting, and sword-fighting are 96%, 99%, 93%, 91%, 78.4%, and 76%, respectively, as shown in Table 4.

The action videos in this dataset contain very low background noise, our approach extracts the features and computes the relative descriptor efficiently, but the actions of sword-fighting and gun-fighting were classified with less accuracy. The videos from HMDB51 were collected mostly from movies and web sources. The videos are low quality with camera motion, illumination effects, nonstatic background, changes in position, viewpoint, and occlusion. Our approach misclassifies few videos due to camera motion and viewpoint variations, but the overall accuracy is acceptable.

4. Conclusions

The proposed approach achieved good performance on all the datasets. Our method utilizes the position of the actor and computes the movement of body parts for feature representation. Then, by combining them into actions using a time series approach. The proposed method efficiently computes the features and later formulates the action. This is why the background noise and occlusion do not affect the overall performance of the approach. However, for future directions, additional research is required to refine the extraction of the features from the background information, and vertical viewpoints of the actors, trajectory information, and optical flow can also help in extracting valuable information. Extraction of additional types of features, as well as dimensionality reductions of the feature space using feature selection methods or Principal Component Analysis (PCA), can lead to a higher performance of the system.

Data Availability

The authors have used publicly available datasets (UT-Interaction, CASIA, and UCF-11). However, HCA dataset can be made available on request to the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors are thankful to the Directorate of Advanced Studies, Research and Technological Development (ASRTD), University of Engineering and Technology Taxila for their support. This research work was carried at Swarm Robotics Lab under National Centre for Robotics and Automation (NCRA), funded by Higher Education Commission, Pakistan.

References

[1] T. Lawson, R. Rogerson, and M. Barnacle, "A comparison between the cost effectiveness of CCTV and improved street lighting as a means of crime reduction," *Computers, Environment and Urban Systems*, vol. 68, pp. 17–25, 2018.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE, San Diego, CA, USA, June 2005.

[3] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition*, June 2008.

[4] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proceedings of the Computer Vision-ECCV 2006, 9th European Conference on Computer Vision*, Graz, Austria, May 2006.

[5] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, September 1999.

[6] J. Uijlings, I. C. Duta, E. Sangineto, and N. Sebe, "Video classification with densely extracted hog/hof/mbh features: an evaluation of the accuracy/computational efficiency tradeoff," *International Journal of Multimedia Information Retrieval*, vol. 4, no. 1, pp. 33–44, 2015.

[7] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proceedings of the IEEE International Conference on Computer Vision*, Sydney, Australia, December 2013.

[8] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic image networks for action recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016.

[9] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," 2014, https://arxiv.org/abs/1406.2199.

[10] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," 2019, https://arxiv.org/abs/1904.02811.

[11] F. Perronnin, J. Sanchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proceedings of the Computer Vision-ECCV 2010, 11th European Conference on Computer Vision, Heraklion*, Crete, Greece, September 2010.

[12] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. 34, no. 9, pp. 1704–1716, 2011.

[13] M. F. Aslan, A. Durdu, and K. Sabanci, "Human action recognition with bag of visual words using different machine learning methods and hyperparameter optimization," *Neural Computing & Applications*, vol. 32, no. 12, pp. 8585–8597, 2020.

[14] S. Nazir, M. H. Yousaf, J.-C. Nebel, and S. A. Velastin, "A Bag of Expression framework for improved human action recognition," *Pattern Recognition Letters*, vol. 103, pp. 39–45, 2018.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[16] Y. Zhao, K. L. Man, J. Smith, K. Siddique, and S.-U. Guan, "Improved two-stream model for human action recognition," *EURASIP Journal on Image and Video Processing*, vol. 2020, no. 1, 9 pages, 2020.

[17] K. Pawar and V. Attar, "Deep learning approaches for video-based anomalous activity detection," *World Wide Web*, vol. 22, no. 2, pp. 571–601, 2019.

[18] Y. Du, Y. Fu, and L. Wang, "Representation learning of temporal dynamics for skeleton-based action recognition," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3010–3022, 2016.

[19] Y. Zhang, Y. Zhang, Z. Zhang, J. Bao, and Y. Song, "Human activity recognition based on time series analysis using U-net," 2018, https://arxiv.org/abs/1809.08113.

[20] G. R. Naik, R. Chai, and R. M. Stephenson, "A system for accelerometer-based gesture classification using artificial neural networks," in *Proceedings of the 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, July 2017.

[21] P. Gupta and T. Dallas, "Feature selection and activity recognition system using a single triaxial accelerometer," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 6, pp. 1780–1786, 2014.

[22] S. R. Dyuthi, K. C. Prakash, M. Panwar et al., "CNN based approach for activity recognition using a wrist-worn accelerometer," in *Proceedings of the 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2438–2441, IEEE, Jeju, Republic of Korea, July 2017.

[23] W. Y. Lin, M. Y. Lee, C.-S. Lai, and V. K. Verma, "Levels of activity identification & sleep duration detection with a wrist-worn accelerometer-based," in *Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, Jeju, Republic of Korea, July 2017.

[24] H. Rezaie and M. Ghassemian, "An adaptive algorithm to improve energy efficiency in wearable activity recognition systems," *IEEE Sensors Journal*, vol. 17, no. 16, pp. 5315–5323, 2017.

[25] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Computer Systems*, vol. 81, pp. 307–313, 2018.

[26] Y. Guan and T. Plötz, "Ensembles of deep LSTM learners for activity recognition using wearables," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 2, pp. 1–28, 2017.

[27] J. Wang, Y. H. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: a survey," *Pattern Recognition Letters*, vol. 119, 2018.

[28] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," 2017, https://arxiv.org/abs/1611.08050.

[29] G. Papandreou, T. Zhu, N. Kanazawa et al., "Towards accurate multi-person pose estimation in the wild," 2017, https://arxiv.org/abs/1701.01779.

[30] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: a deeper, stronger, and faster multi-person pose estimation model," in *Proceedings of the European Conference on Computer Vision*, Amsterdam, The Netherlands, October 2016.

[31] A. Ullah, K. Muhammad, T. Hussain, and S. W. Baik, "Conflux LSTMs network: a novel approach for multi-view action recognition," *NeuroComputing*, Elsevier, vol. 435, pp. 321–329, 2021.

[32] A. Toshev and C. Szegedy, "Deeppose: human pose estimation via deep neural networks," 2014, https://arxiv.org/abs/1312.4659.

[33] J. Gall, A. Yao, and L. V. Gool, "2D action recognition serves 3D human pose estimation," in *Proceedings of the European Conference on Computer Vision*, Crete, Greece, September 2010.

[34] N. K. Ahmad Jalal, "Automatic recognition of human interaction via hybrid descriptors and maximum entropy markov model using depth sensors," *Entropy*, vol. 8, no. 22, 2020.

[35] M. Afrasaibi, H. Khotanlou, and T. Gevers, "Spatial-temporal dual-actor CNN for human interaction prediction in video," *Multimedia Tools and Applications*, vol. 29, 2020.

[36] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and machine learning forecasting methods: concerns and ways forward," *PLoS One*, vol. 13, no. 3, p. e0194889, 2018.

[37] M. A. R. Ahad, *Motion History Images for Action Recognition and Understanding*, Springer Science & Business Media, Berlin, Germany, 2012.

[38] S. Karmakar and S. Karmakar, "Image enhancement by the combination of multivalued logic and Savitzky-Golay filter," *International Journal of Electronics Letters*, vol. 7, no. 3, pp. 290–303, 2019.

[39] P. Kaur, G. Singh, and P. Kaur, "Intellectual detection and validation of automated mammogram breast cancer images by multi-class SVM using deep learning classification," *Informatics in Medicine Unlocked*, vol. 16, p. 100151, 2019.

[40] M. S. Ryoo and J. Aggarwal, UT-interaction dataset, ICPR contest on semantic description of human activities (SDHA), 2010.

[41] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wild," in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009.

[42] Z. Dacheng Tao and D. Tao, "Slow feature analysis for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 436–450, 2012.

[43] A. M. Mir, M. H. Yousaf, and H. Dawood, "Criminal action recognition using spatiotemporal human motion acceleration descriptor," *Journal of Electronic Imaging*, vol. 27, no. 6, p. 063016, 2018.

[44] Y. S. Sefidgar, A. Vahdat, S. Se, and G. Mori, "Discriminative key-component models for interaction detection and recognition," *Computer Vision and Image Understanding*, vol. 135, pp. 16–30, 2015.

[45] N. Nguyen and A. Yoshitaka, "Classification and temporal localization for human-human interactions," in *Proceedings of the 2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*, April 2016.

[46] Y. Cheng, Y. Yang, H.-B. Chen, N. Wong, and H. Yu, "S3-net: a fast and lightweight video scene understanding network by single-shot segmentation," 2020, https://arxiv.org/abs/2011.02265.

[47] G. K. Yadav, P. Shukla, and A. Sethfi, "Action recognition using interest points capturing differential motion information," in *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016.

[48] L. Wang, Y. Xu, J. Cheng, H. Xia, J. Yin, and J. Wu, "Human action recognition by learning spatio-temporal features with deep neural networks," *IEEE access*, vol. 6, pp. 17913–17922, 2018.

[49] A. Ullah, K. Muhammad, W. Ding et al., "Efficient activity recognition using lightweight CNN and DS-GRU network for surveillance applications," *Applied Soft Computing*, Elsevier, vol. 103, 2021.