IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 21 July 2023, accepted 31 July 2023, date of publication 7 August 2023, date of current version 14 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3302353

RESEARCH ARTICLE

Aerial Insights: Deep Learning-Based Human Action Recognition in Drone Imagery

USMAN AZMAT[1], SAUD S. ALOTAIBI[2], MAHA ABDELHAQ[3], (Member, IEEE), NAWAL ALSUFYANI[4], MOHAMMAD SHORFUZZAMAN[4], AHMAD JALAL[1], AND JEONGMIN PARK[5]

[1]Department of Computer Science, Air University, Islamabad 44000, Pakistan
[2]Information Systems Department, Umm Al-Qura University, Macca 24382, Saudi Arabia
[3]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[4]Department of Computer and Information, Prince Sultan University, Riyadh 12435, Saudi Arabia
[5]Department of Computer Engineering, Tech University of Korea, Siheung-si, Gyeonggi-do 15073, South Korea

Corresponding author: Jeongmin Park (jmpark@tukorea.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation (NRF) (2021R1F1A1063634) funded by the Ministry of Science and Information & Communications Technology (MSIT), Republic of Korea.

ABSTRACT Human action recognition is critical because it allows machines to comprehend and interpret human behavior, which has several real-world applications such as video surveillance, robot-human collaboration, sports analysis, and entertainment. The enormous variety in human motion and appearance is one of the most challenging problems in human action recognition. Additionally, when drones are employed for video capture, the complexity of recognition gets enhanced manyfold. The challenges including the dynamic background, motion blur, occlusions, video capture angle, and exposure issues gets introduced that need to be taken care of. In this article, we proposed a system that deal with the mentioned challenges in drone recorded red-green-blue (RGB) videos. The system first splits the video into its constituent frames and then performs a focused smoothing operation on the frames utilizing a bilateral filter. As a result, the foreground objects in the image gets enhanced while the background gets blur. After that, a segmentation operation is performed using a quick shift segmentation algorithm that separates out human silhouette from the original video frame. The human skeleton was extracted from the silhouette, and key-points on the skeleton were identified. Thirteen skeleton key-points were extracted, including the head, left wrist, right wrist, left elbow, right elbow, torso, abdomen, right thigh, left thigh, right knee, left knee, right ankle, and left ankle. Using these key-points, we extracted normalized positions, their angular and distance relationship with each other, and 3D point clouds. By implementing an expectation maximization algorithm based on the Gaussian mixture model, we drew elliptical clusters over the pixels using the key-points as the central positions to represent the human silhouette. Landmarks were located on the boundaries of these ellipses and were tracked from the beginning until the end of activity. After optimizing the feature matrix using a naïve Bayes feature optimizer, the classification is performed using a deep convolutional neural network. For our experimentation and the validation of our system, three benchmark datasets were utilized i.e., the UAVGesture, the DroneAction, and the UAVHuman dataset. Our model achieved a respective action recognition accuracy of 0.95, 0.90, and 0.44 on the mentioned datasets.

INDEX TERMS Convolutional neural network, expectation maximization, Gaussian mixture model, quadratic discriminant analysis, quick-shift segmentation, video processing.

I. INTRODUCTION

Recognition of the actions performed by the humans is concerned with processing of the images in a way to collect

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

features about human motion and automatically identify the actions they are performing. The ability to recognize human actions has a wide range of applications, including video surveillance and human-robot interaction, as well as sports analysis and entertainment. Action recognition can be used in video surveillance to automatically identify and indicate potential security hazards, such as individuals carrying weapons or engaged in doubtful behavior. This could serve to increase public safety and lower the likelihood of criminal behavior. While working with human-robot interaction, action recognition can help robots better comprehend and respond to human behavior, making them more helpful and effective in various scenarios ranging from manufacturing and logistics to healthcare and personal assistance [1], [2], [3], [4], [5], [6], [7]. When applied to sports analysis, action recognition has the ability to provide coaches and analysts with considerable insights into player performance and team dynamics, allowing them to make better educated decisions to improve the team's overall success. Moreover, in the entertainment industry, action recognition can enhance virtual reality and gaming experiences by enabling more natural and intuitive interaction between humans and machines, which makes such experiences more immersive and enjoyable [8], [9], [10], [11], [12], [13].

Researchers have faced many challenges while working on the task of recognizing human actions. The first is that human actions may vary significantly in terms of pose, appearance, speed, and context [14], [15], [16]. This makes developing robust algorithms that can accurately recognize actions in different settings quite difficult. Another problem is the limited data: collecting large amounts of labeled data for human action recognition is time-consuming and expensive. This limit impacts the size of the datasets that are available and can make it challenging to train accurate models. Actions are inherently temporal; recognizing them requires an understanding of the sequence of movements that comprise them [17], [18], [19]. This introduces additional complexity into the recognition process. Occlusion and clutter may be present in real-world scenes, which makes it difficult to accurately recognize actions. For example, suppose a person is partially occluded by an object or another person. In that case, the action they are performing can be challenging to recognize [20], [21]. The viewpoint of the camera has an impact on the appearance and motion of a human performing an action. Because the camera viewpoint introduces variability, it may be difficult to recognize actions from different camera angles. Lastly, real-time processing is necessary for many action recognition applications, such as surveillance or robotics. Achieving real-time performance while maintaining accuracy can be quite a challenging task [22], [23], [24], [25], [26]. Another great challenge that emerges when videos are recorded by drone-mounted cameras is that the background of the image is not static but instead varies with the drone's motion.

An initial system, referred to in [27], was developed to address the task of human action recognition. It employed traditional computer vision approaches combined with machine learning techniques to automatically recognize and classify human actions based on RGB and depth video data. The system utilized a multi-step process, starting with frame splitting and bilateral filtering for noise reduction. Simple Linear Iterative Clustering (SLIC) segmentation was applied to extract the human region, followed by Euclidean distance transform to identify body joints. The estimation of body parts using RGB and Depth information was performed through the utilization of elliptical modeling over expectation-maximization based on the Gaussian mixture model (EM-GMM). A conditional random field (CRF) enabled labelling of individual pixels of the human body parts. Fisher's discriminant analysis reduced the dimensionality of features extracted from 3D point cloud data and landmarks. Finally, classification was performed using the K-ary tree hashing algorithm. However, the system had limitations and areas for improvement. The reliance on depth information restricted its applicability, as depth data may not always be available or relevant in certain scenarios. Moreover, the system's performance in real-world scenarios was limited due to the challenges posed by varying viewpoints and complex environmental conditions.

To address these limitations and enhance the accuracy of human action recognition, the proposed system was developed. It focuses on recognizing human actions in aerial RGB videos, particularly from drone footage, where depth information might not be readily available or applicable. It adopts a specialized approach that excludes the use of depth data and incorporates advanced techniques, including deep learning architectures. The RGB videos are first sliced into frames and then subjected to blurring using a bilateral filter for noise reduction. Human segmentation is achieved through a quick shift segmentation algorithm, which demonstrates superior performance in isolating humans in aerial imagery. Feature extraction in the proposed system encompasses various aspects, such as normalized joint positions, angular and distance relationships among body joints, and 3D point cloud data. The technique of EM-GMM is utilized for the purpose of delineating elliptical clusters over the pixels, generating an individual representation of the human body parts. Landmarks are located on the boundaries of these ellipses and tracked throughout the action sequence. A noteworthy advancement in the proposed system is the incorporation of convolutional neural networks (CNNs), which are a type of deep learning architecture. By leveraging CNNs, system optimized the feature extraction and improved action classification. Their use in the proposed system enabled the model to learn hierarchical representations from the RGB imagery, enhancing generalization and robustness for the action recognition in drone recordings.

The proposed version of the system demonstrated superior performance as compared to the previous version. The advancements in accurate human segmentation, feature extraction using deep learning architectures, and action classification capabilities make it more versatile, adaptable, and effective for aerial action recognition. By excluding depth information, the system achieves a wider range of applicability, especially in scenarios where depth data is either unavailable or not pertinent to the specific context. The key contributions of our study are listed below:

- The specialized approach fills the gap in existing literature by addressing the challenges of recognizing human actions in aerial videos while taking only RGB videos as input and making the system independent of the depth information as it is not readily available.
- By utilizing CNNs, the system improves feature extraction and action classification. By incorporating them, the model is able to derive hierarchical representations from input RGB frames, which improves the system's robustness and generalizability.
- A cutting-edge image segmentation method known as quick-shift segmentation is used to efficiently isolate a human from its context to identify the human's actions.
- An algorithm for extracting a 3D point cloud data is presented, and it is shown to be a helpful feature for accurately identifying human action.
- A comprehensive comparison is conducted on three openly available datasets encompassing various human behaviors. The experimental outcomes validate that the proposed approach attains higher recognition accuracy compared to other existing techniques.

The hierarchy of the remaining paper follows the flow i.e., section II offers a succinct evaluation of several state-of-the-art approaches. The methodology employed by the system proposed is outlined in section III. In section IV, an evaluation of the effectiveness of our suggested technique is presented based on three benchmark datasets. The advantages and disadvantages of the proposed system are discussed in section V. Lastly, section VI concludes the study and explores potential future directions.

II. RELATED WORK

Researchers have developed many computer vision algorithms for human action recognition in the recent past. The work related to our study is divided into two subsections: Section A includes human action recognition by machine learning, while Section B covers action recognition by deep learning.

A. HUMAN ACTION RECOGNITION BY MACHINE LEARNING

Arunnehru et al. focused on action recognition based on motion patterns by analyzing changes in a subject's location over time. The system started by converting the input video to grayscale and applying noise removal techniques to enhance fine features. To extract motion features, the frame difference method was employed. This method calculated the intensity differences between consecutive frames, highlighting areas of motion. This study employed a technique known as difference intensity distance group pattern (DIDGP) to extract 2D/3D cuboids from action sequences. To each spatio-temporal interest point, transformations such as discrete cosine transform (DCT), discrete wavelet transform (DWT), and a hybrid of DWT+DCT were applied. Finally, for action classification, the system used support vector machine (SVM) and Random Forest classifiers. Despite its achievements, the system suffered from certain shortcomings. Firstly, it focused solely on motion patterns and did not consider other important cues, such as appearance and spatial relationships between body parts. This limitation might affect the system's ability to accurately recognize complex actions that involve subtle appearance changes or rely on spatial context. Moreover, the system's reliance on handcrafted features and traditional machine learning algorithms may restrict its ability to capture complex patterns and generalize well across different action scenarios [28]. The proposed system addresses these limitations by incorporating deep learning architectures. It takes advantage of the spatial and temporal information in aerial RGB videos and utilizes a CNN, for better action classification.

In order to address the difficulties associated with variance within the classes, variability between the speed of the movement, and computing complexity in action recognition tasks, the authors presented a unique hierarchical model for a three-dimensional (3D) action recognition setting its basis on skeleton data. A part-based clustering module, which used a five-dimensional (5D) feature vector concentrating on the most pertinent joints of body parts within each action sequence, was introduced at the first level of the framework. Transitioning to the next level, the architecture contained two modules: motion feature retrieval and action graphs. Within the motion feature extraction module, only the joints relevant to the clusters were considered. Additionally, a novel statistical principle was proposed to determine the optimal time scale for extracting motion features. The system focused solely on skeleton-based features and did not fully exploit other important features, such as appearance or spatial relationships between body parts. This limitation might hinder accurately recognizing complex actions that rely on these additional features [29]. The proposed system captures both appearance and motion information, allowing for a more comprehensive understanding of actions.

Zhen et al. investigated local methods based on spatio-temporal interest points (STIPs), such as sparse coding (SC), bag-of-words (BoW), vector of locally aggregated descriptors (VLAD), Fisher kernels (FK), and the naive Bayes nearest neighbor (NBNN) classifier, to recognize human action in videos. Among the approaches tested, the enhanced Fisher kernels (IFK) produced the best results. Although promising in the image domain, the performance of these local approaches for action recognition may not translate effectively to the video domain [30]. To address this problem, the proposed approach considers the temporal dynamics and spatial relationship found in video sequences, both of which are required for successful action recognition in videos. Yang et al. proposed a ground-breaking framework for recognizing human actions in video sequences captured by depth cameras in their study. The framework's main goal was to effectively capture both local motion and shape information. This was accomplished by extending the surface normal to polynormal, which entailed assembling local neighboring hypersurface normals from a depth sequence. A strategy termed super normal vector (SNV) was utilized to aggregate the low-level polynormals into a discriminative representation, which acted as a reduced version of the FK representation, enabling for the extraction of a compact and informative representation of the depth video data. The architecture is exclusively based on depth information and does not fully utilize additional modalities, such as RGB, which could give complementing features for more robust recognition [31]. By analyzing RGB videos, the proposed system captures not only depth information but also color and texture features, providing a more comprehensive understanding of human activities.

In [32], a framework for human action recognition has been proposed that employed robust multi-features and embedded hidden Markov models (HMMs) to analyze depth maps and human motion tracking in a video. The framework began by applying a temporal motion identification method to segment human silhouettes from the noisy background in depth maps. The depth silhouette area was then computed for each activity, enabling the tracking of human movements. Embedded HMMs were employed to learn, model, train, and recognize the features with active values, allowing for activity recognition. This method sets its basis over the depth data that is not always available while the proposed system tackles this by working over RGB data. Shahroudy et al.'s study offered a novel way of action recognition that utilized a joint sparse regression-based learning technique. This approach combined multimodal characteristics derived from a sparse set of body parts with organized sparsity to model each action. In the study, a wide variety of depth and skeleton-based features were applied to efficiently capture both dynamics and appearance. The study only focused on multimodal features from a sparse set of body parts, and it did not consider the whole-body features [33]. While the proposed model first extracts a comprehensive set of features including both individual body part features and whole-body features and then employs deep learning architecture to final training that makes the system more generalizable and robust.

B. HUMAN ACTION RECOGNITION BY DEEP LEARNING

The authors in [34] proposed an end-to-end fully connected deep long-short-term-memory (LSTM) network for skeleton-based action recognition. A significant observation made in this study is that the co-occurrences of skeleton joints inherently provide essential characteristics of human actions. In order to capture this information, the skeleton was treated as the input at each time slot, and a novel regularization scheme was introduced to learn the co-occurrence features of the skeleton joints. However, it is worth noting that this work focuses specifically on skeleton-based representations and does not consider other modalities such as RGB or depth information. The proposed system overcomes the reliance on skeleton-based representations by working directly with RGB videos, enabling the extraction of rich visual features from aerial imagery. Li et al. [35] introduced a novel approach called VLAD for Deep Dynamics (VLAD3) for action recognition, addressing the limitations of previous methods that primarily focused on short-term temporal information and did not explicitly model long-range dynamics. The VLAD3 combined different levels of video dynamics to capture and represent temporal information comprehensively. It incorporated deep CNN features to capture short-term dynamics. To model medium-range dynamics, linear dynamic systems (LDS) were utilized, which enabled the modeling of temporal dependencies over a longer time span. The limitations of this approach lie in its reliance on pre-trained deep CNNs and the assumption of linearity in the LDS model. Pre-trained networks may not capture all the necessary information for a specific action recognition task, and the linearity assumption in LDS may limit its ability to handle complex non-linear temporal dynamics. In contrast, our system directly works with RGB videos, enabling the extraction of rich visual features and capturing non-linear temporal dynamics without relying solely on pre-trained networks.

Shi et al. [36] presented a novel descriptor called the sequential Deep Trajectory Descriptor (sDTD) that aimed to learn a robust representation of long-term motion. Many existing descriptors struggle to capture motion information effectively, especially over extended time periods. The proposed sDTD addressed this issue by projecting dense trajectories into two-dimensional planes. This transformation allowed for extracting spatial and temporal information from the trajectories. A CNN-RNN network was used to learn a useful representation for long-term motion. The network could recognize both spatial and temporal correlations in the motion data because of the fusion of CNN and RNN. However, the limitations of this approach include the reliance on dense trajectory extraction, which may be sensitive to noisy or cluttered scenes. Our proposed system offers a solution to address these limitations by operating directly on RGB videos and not relying on explicit trajectory extraction. Du et al. [37] proposed an end-to-end hierarchical recurrent neural network (HRNN) for skeleton-based action recognition. The motivation behind using HRNN was its ability to effectively model long-term contextual information in temporal sequences. Instead of considering the entire skeleton as input, the authors divided the skeleton into five parts based on the human physical structure. These five parts were separately fed into five subnets, each responsible for extracting representations from a specific part. The representations retrieved by the subnets were combined in a hierarchical fashion as the number of layers increased to create the inputs for higher layers. With the help of this hierarchical fusion, more abstract and complicated features from the skeletal sequences may be captured. The completed skeleton sequence representations were then input into a single-layer perceptron. The final choice for action recognition was then made using the perceptron's temporally accumulated output. Their approach significantly relied on skeleton data, which was not always readily available.

In their study, Mihanpour et al. introduced a hybrid architecture comprising a deep bidirectional LSTM (DB-LSTM) and CNN. The video frames were first processed using a ResNet152 [38] to extract deep features. These extracted features were then fed into a DB-LSTM for training purposes. The authors' findings supported the claim that their technology outperformed cutting-edge techniques. The accuracy of their study was improved by using the pre-trained ResNET152 to properly adjust the input parameters. The computational complexity of the system was improved by using multiple threads in parallel [39]. Muhammad et al. presented a system that used a mix of strategies to improve human action detection in videos, including a bi-directional long short-term memory (BiLSTM)-based attention mechanism and a dilated CNN (DCNN). The DCNN layers in their system were utilized to extract salient discriminative features from the input frames. By using residual blocks, the system enhanced these features, allowing them to retain more informative details compared to shallow layers. These upgraded features were fed into a BiLSTM, enabling the model to capture long-term dependencies and temporal dynamics associated with human actions [40]. Their system might struggle with recognizing complex actions that involve intricate temporal patterns or subtle motion features. While our system addresses these limitations by incorporating techniques such as quick shift segmentation and skeleton extraction.

III. THE PROPOSED SYSTEM

The system's goal is to analyze input videos and recognize the activities of humans. The video is first segmented into frames. The frames are then denoised with the use of the bilateral filter. The segmentation block receives the denoised frames and the quick shift method processes the frames with a kernel-based strategy to segment the human silhouette out of it. The silhouette is then morphologically eroded in an iterative manner to obtain skeleton. Like a silhouette, the skeleton has proven to be particularly helpful in identifying the key-points of the body. Key-points obtained from the skeleton and utilized to extract features serve as representations of the joints of the human body. The retrieved features in our study cover a wide range of topics. These include the angle between key-points that are adjacent to one another within a frame, the separation between a key-point's current position and its position in the previous frame, the key-point's speed between two subsequent frames, the RGB image's 3D point cloud data and the associated silhouette, and finally the landmarks that identify each body part. The EM-GMM approach is used for elliptical modelling, resulting in the creation of human body parts. Next, a single data frame containing each of these attributes is created and labeled appropriately. A naïve Bayes-based feature optimizer is then used to optimize the features. In order to perform the classification, a CNN model is trained using the optimized data. Fig. 1 illustrates the general design of the proposed system.

A. FRAME EXTRACTION AND DENOISING

The system initially accepts a video as input, but all computational and analytical procedures are run on individual images. Therefore, it is necessary to split the video into its constituent images for further processing. The first step in the image processing is to denoise the images. For this purpose, the proposed system utilizes a bilateral filter [41]. The bilateral filter is a non-linear image processing and computer vision technique that smooths images while preserving edges. By considering both the spatial distance and intensity differences between neighboring pixels, it applies a weighted average to achieve the desired effect. Mathematically it is defined as:

$$BL\left[I_p\right] = \frac{1}{W_p}\sum\nolimits_{q\in S} G_{\sigma_S}\left(||p-q||\right) G_{\sigma_r}\left(\left|I_p - I_q\right|\right) I_q \quad (1)$$

where I_p is the filtered intensity value at pixel p, I_q is the intensity value at pixel q, σ_s is the span of the neighborhood under consideration, σ_r is the minimum amplitude of the edge under consideration, and W_p is the normalization factor for pixel p. The original frame and that frame after it has undergone bilateral filtration are shown in Fig 2.

B. QUICK SHIFT SEGMENTATION

Quick shift is a non-parametric, kernel-based algorithm that can segment an image into regions that have homogeneous color or texture. The key idea behind quick shift segmentation is that it performs mode seeking in a hierarchical manner. It starts by treating each pixel as a region, and then iteratively merges sections with similar colour and texture. A distance metric based on colour difference and spatial distance between pixels is used to determine similarity. Each pixel in the image is moved to an adjacent pixel with a higher density of other pixels by the algorithm. This is accomplished by the use of a density map, which assigns a density value to each pixel based on the number of surrounding pixels with comparable color and texture. The density map is used to determine the image's modes, or the densest regions of pixels in the image. After that, the modes are employed as seeds to enlarge the image regions [42]. The following equation governs the bilateral filtering operation:

$$D\left(x, y\right) = \sqrt{\frac{\left(||x-y||\right)^2 + \left(V_x - V_y\right)^2}{h^2}} \quad (2)$$

where V_x and V_y are the texture features of pixels x and y respectively, and h is a scaling factor that controls the influence of color and texture. Some examples of quick shift segmented images are shown in Fig. 3.

C. SKELETONIZATION AND KEYPOINT EXTRACTION

The human silhouette is extracted from the RGB video frame using quick shift segmentation. The binary silhouette

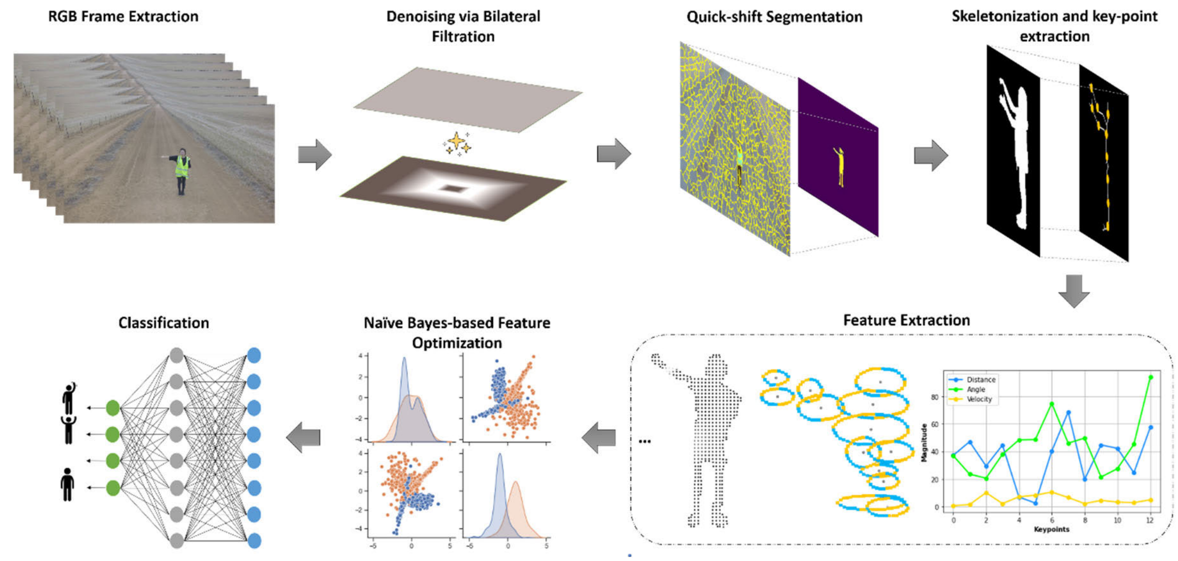

FIGURE 1. Architecture of the proposed system for human action recognition.

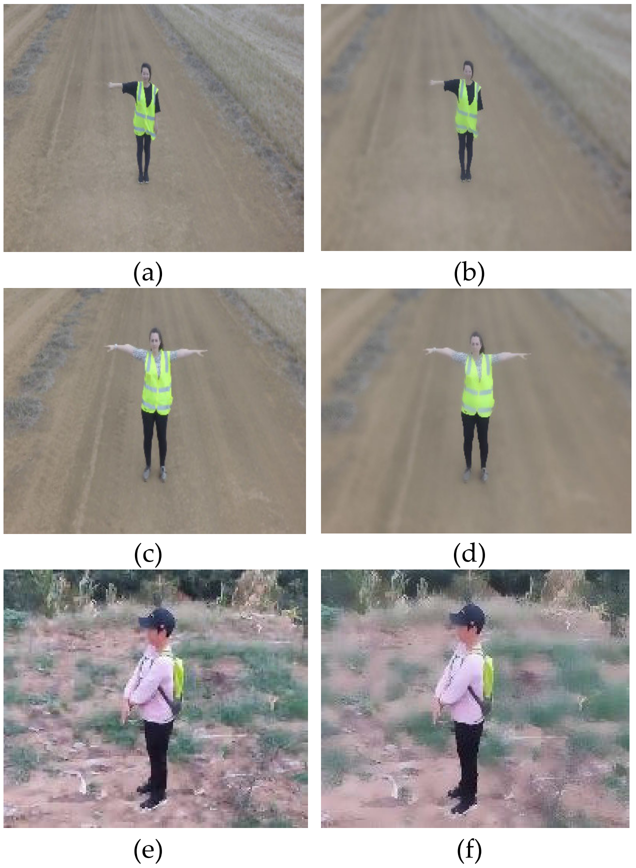

FIGURE 2. Denoised image after application of bilateral filter (a) Input image for Not clear, (b) Denoised image for Not clear, (c) Input image for Hover, (d) Denoised image for Hover, (e) Input image for Land, and (f) Denoised image for Land.

is then subjected to an iterative erosion procedure to produce the skeleton demonstrated by Fig. 4 where part (a) represents the human silhouette for landing direction and part (b) represent the skeleton for the same silhouette. While in part (c), a silhouette for move-up is shown and part (d) shows the its skeleton.

The skeleton is a depiction of the human body that identifies the body's joints. For feature extraction and correctly identifying the action taken by the subject, the joint locations are essential [43], [44]. The classification is more accurate the closer the joint location is to being exact. The skeleton is contoured, and a convex hull is drawn over it to highlight the key-points [45], [46]. The hull's extreme points are then determined. The head, left wrist, right wrist, left ankle, and right ankle can all be visualized as five important points using this method. Eq. 4 is utilized to compute the moment M of the contours, while, in order to get the x and y coordinates of the centroid, which is often located in the lower abdomen region, Eq. 5 is used.

$$M_{ij} = \sum\nolimits_{x} \sum\nolimits_{y} x^{i} y^{j} I(x, y) \tag{3}$$

$$x, y = (M_{10} / M_{00}, M_{01} / M_{00}) \tag{4}$$

Additional key-points are required for proper body depiction during the feature extraction step. Therefore, the two most appropriate points of the six points previously calculated are chosen to determine the midpoint; next, the point on the skeleton that is closest to that midpoint is selected, using the Euclidean distance, to represent the new body key-point. The midpoint of the abdominal and head points, for the key point of the chest, was determined. Then, the Euclidean distances between each point on the skeleton and the recently discovered midpoint were computed. The chest was then represented by the closest skeletal point. In the same way, the points for the left elbow, right elbow, left thigh, right thigh, left knee, and right knee were identified. Eq. 6 is the formula

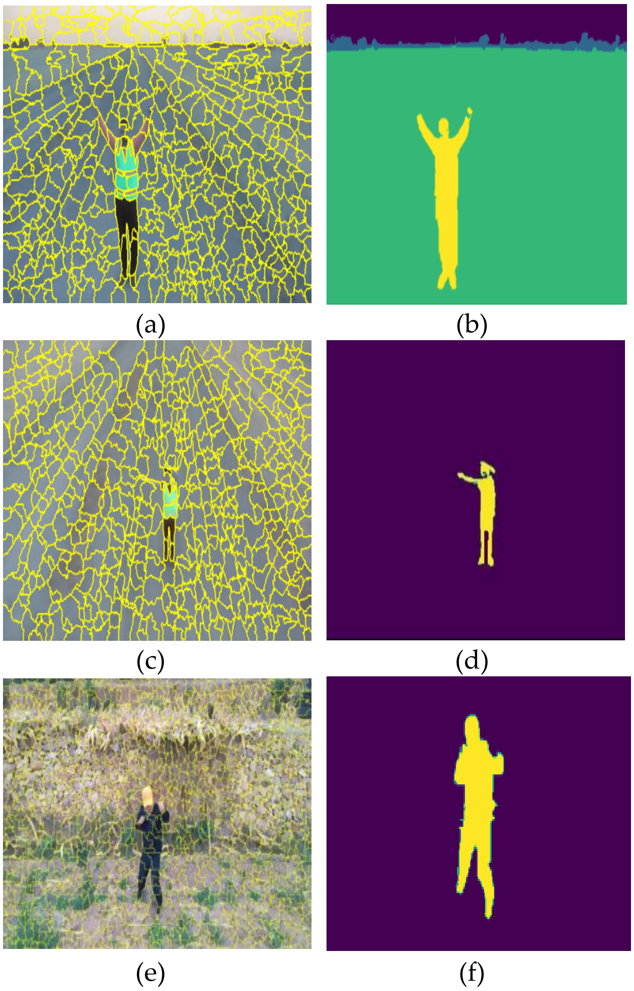

FIGURE 3. Quick shift segmentation (a) Quick shift pixel-regions for Move up, (b) Segmented image for Move up, (c) Quick shift pixel-regions for Move left, (d) Segmented image for Move left, (e) Input image for punching, and (f) Segmented image for punching.

for computing the midpoints.

$$a_m, b_m = \left(\frac{x_i + x_j}{2}, \frac{y_i + y_j}{2}\right) \tag{5}$$

where a_m, and b_m represent the horizontal and vertical coordinate of the midpoint of (x_i, y_i) and (x_j, y_j). The thirteen-key-point model, as described, yielded the best performance and produced satisfactory results for the datasets employed in this study. Fig. 5 provides the visual representations of the key-points for all thirteen body parts i.e., the head, left wrist, right wrist, left elbow, right elbow, torso, abdomen, right thigh, left thigh, right knee, left knee, right ankle, and left ankle.

D. FEATURE ENGINEERING

The features of a system directly impact its effectiveness; a system's intelligence can be boosted by good features. Various features were extracted, including normalized positions and their angular and spatial connections, 3D point cloud data, and landmarks. All of these features are described in depth in the following subsections.

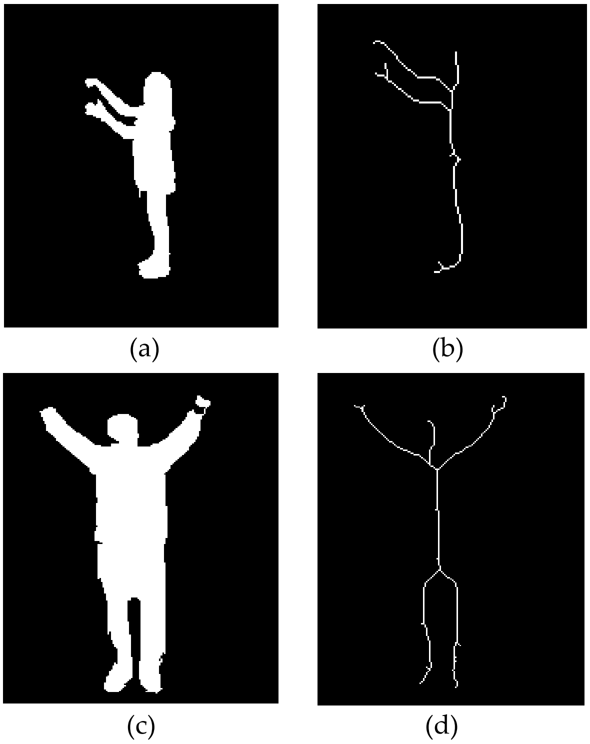

FIGURE 4. Skeleton Extraction (a) Human silhouette for Landing direction, (b) Human skeleton for Landing direction, (c) Human silhouette for Move up, and (d) Human skeleton for Move up.

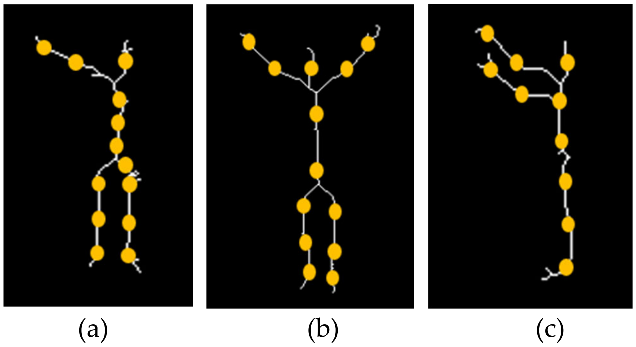

FIGURE 5. Keypoints for (a) All clear, (b) Move up, and (c) Landing direction.

1) ANGULAR RELATIONSHIP OF KEY POINTS

Relative joint angles indicate how the limbs are positioned in relation to one another during an action. The accuracy of action recognition can be improved by tracking these angles [47]. The following relation was employed to obtain the angle between two points:

$$\varphi = tan^{-1}(y_2 - y_1 / x_2 - x_1) \tag{6}$$

(x_1, y_1) and (x_2, y_2) represent the coordinates of the two points being considered. Fig. 6 illustrates the computed angles in the form of one-dimensional signals for all of the activities

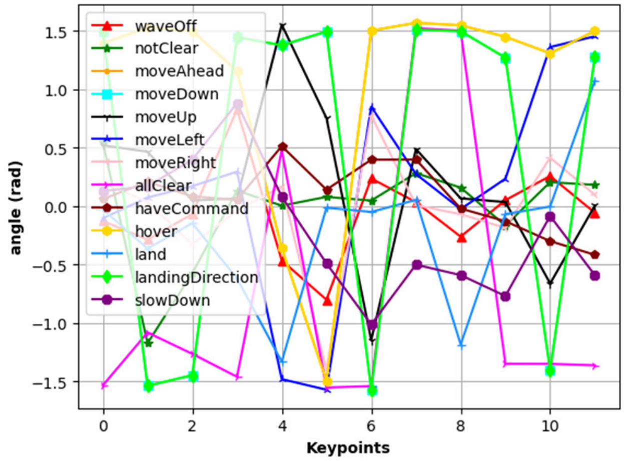

FIGURE 6. Angles between the keypoints for all thirteen activities of the UAVGesture dataset.

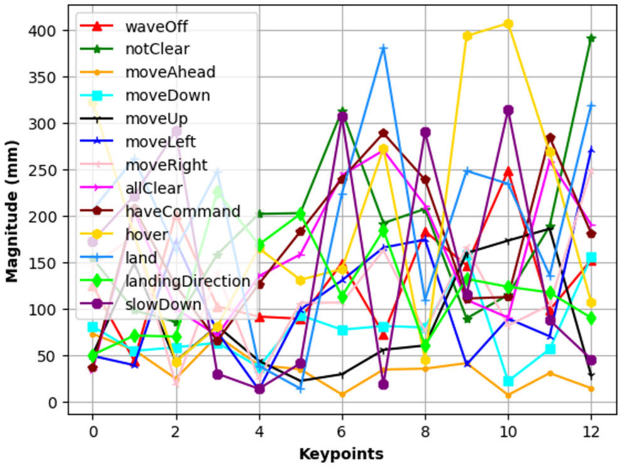

FIGURE 7. Linear displacement of the key-points for all thirteen activities of the UAVGesture dataset.

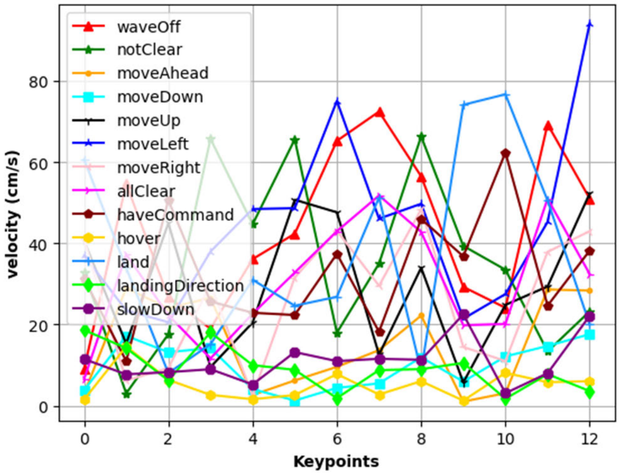

FIGURE 8. Key-points velocity for all thirteen activities of the UAVGesture dataset.

provided in the UAVGesture dataset that are indicated by the legends in the plot.

2) LINEAR DISPLACEMENT AND VELOCITY

The subject's relevant body parts move continuously while the subject is executing an action. The speed at which the transition happens as well as the frame-to-frame distance travelled by each key-point are used by the proposed system as features for the action carried out [48]. To calculate the traveled distance, two successive frames are considered. The following relation is used to calculate the distance between past and present positions of each key-point:

$$distance = \sqrt{(x_2 - x_1)^2 - (y_2 - y_1)^2} \tag{7}$$

To calculate each key-point's individual velocity, the slope of the calculated distance is computed. Mathematically,

$$velocity = \frac{\Delta\, distance}{\Delta\, time} \tag{8}$$

Fig. 7 represents the linear displacement covered by the body parts while going from one frame to the next for all of the activities provided in the UAVGesture dataset. While the unit of the linear displacement is millimeters. On the other hand, Fig. 8 depicts the velocity of the body parts by which they change their position while performing a particular activity. The units of the velocity are centimeters per second.

3) 3D POINT CLOUD DATA

A 3D point cloud data is a depiction of an actual object where each point's x, y, and z coordinates are used to identify it in 3D space [49]. The suggested approach gives the user a 3D point cloud data of an action. The pixels in the image had to be given an additional dimension to make this possible. The additional dimension was created by using the human silhouette. An iteration is then carried out horizontally across the image after calculating the coordinates of the image's center pixel. The point cloud data is defined with specified focal length and scaling factor and the z dimension of the cloud data is determined using both the RGB image and grayscale silhouette in a concurrent manner. This is expressed mathematically as:

$$Z = \frac{1}{SF} * Sil\,[u, v] \tag{9}$$

u and v stand for the x and y coordinates of the pixel under consideration, whereas SF stands for the scaling factor. Following relationships are used to compute the final two dimensions, X and Y:

$$X = \frac{Z}{F} * (u - C_x) \tag{10}$$

$$Y = \frac{Z}{F} * (v - C_y) \tag{11}$$

F stands for the focal length, whereas C_x and C_y stand for the x and y coordinates of the center pixel, respectively.

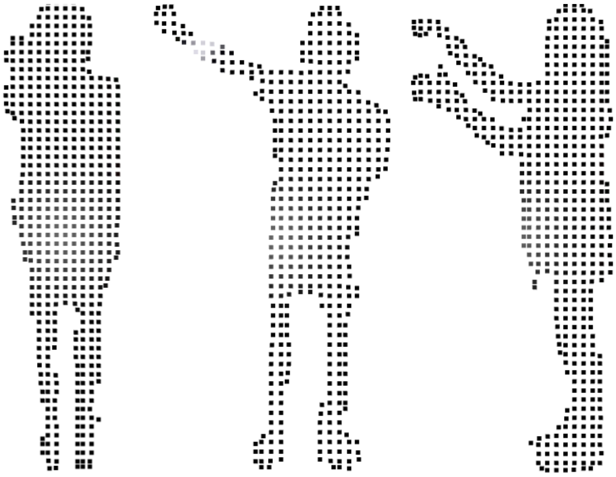

FIGURE 9. 3D point cloud data for (a) Have command, (b) All clear, and (b) Landing direction.

By repeatedly iterating through all of the image's pixels utilizing these relations, a point-cloud-3D is created [50]. A voxel grid filter was utilized in our study to simplify point clouds data despite the fact that they are intrinsically complex. A 3D point cloud data is made simpler using a voxel grid filter by having fewer points in it. Only the points that are contained within voxels that meet specific point density criteria are kept, while others are discarded. This maintains the point cloud's general form and structure while greatly reducing its complexity. A feature vector is then created that contains the down sampled point cloud data for classification. The final depiction of the 3D point cloud data is given in Fig. 9.

4) ELLIPTICAL MODELING

By using the EM method and the restriction that each component's covariance matrix assumes an elliptical form, elliptical modeling applies a GMM to data [51]. The EM technique uses a two-step process to iteratively estimate the GMM parameters. The probability of every data point to belong to every cluster generated by GMM is estimated using posterior probabilities in the expectation step (E-step), which is based on the most recent parameter estimates. Irom:

$$P\left(z_i = k \mid x_i, \theta\right) = \frac{\pi_k{}^{*}N\left(x_i \mid \mu_k, \Sigma_k\right)}{\Sigma_j\left(\pi_j * N\left(x_i \mid \mu_j, \Sigma_j\right)\right)} \tag{12}$$

z_i stands for the latent variable reflecting the data point x_i's elliptical component assignment. $N(xi \mid \mu_k, \Sigma_k)$ is the k^{th} ellipse's probability density function for the normal distribution, and θ, denotes a set containing (π, μ, Σ) parameters of the GMM cluster. The posterior probabilities are computed in the E-step. The parameter estimates are updated by maximizing the projected log-likelihood of all the data using the estimated posterior probability in the succeeding maximization step (M-step).

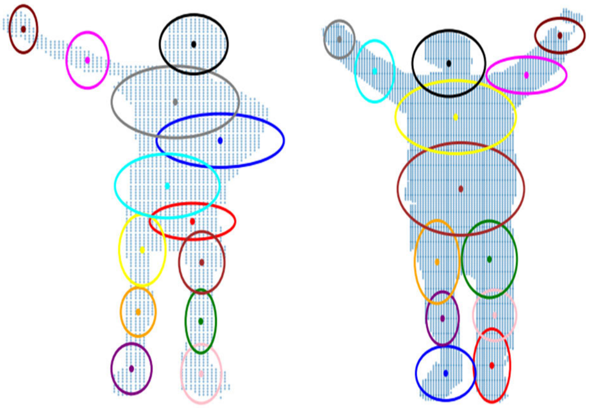

FIGURE 10. EM-GMM-based elliptical modeling (a) All clear, and (b) Move up.

$$\pi_k = \frac{1}{N} * \Sigma_i(P(z_i = k|x_i, \theta)) \tag{13}$$

$$\mu_k = \frac{1}{n_k} * \Sigma_i(P\left(z_i = k \mid x_i, \theta\right) * x_i) \tag{14}$$

$$\Sigma_k = \frac{1}{n_k} * \Sigma_i(P\left(z_i = k \mid x_i, \theta\right) * (x_i - \mu_k)(x_i - \mu_k)^T) \tag{15}$$

N represents gross data points in the cluster, n_k represents the gross points belonging to k^{th} ellipse, and $(x_i - \mu_k)(x_i - \mu_k)^T$ stands for the tensor product of the difference between the vectors x_i and μ_k. The E and M-step iterations are performed until the convergence is reached, which is typically demonstrated by the difference in log-likelihood between subsequent rounds dropping below a particular threshold.

The algorithm takes a binary image containing a silhouette of the human and extracted key-points as input. Initially, circles of the same radius are placed on the silhouette, with the centroids of these circles aligned with the body key-points. There are 13 key points in this scenario, which is the same number of clusters into which the silhouette is divided. The EM-GMM method is applied after the circle assignment, and it iterates over Eq. 10 through Eq. 13 to obtain ellipses that best-fit the silhouettes. A human body model is generated by utilizing separate ellipses to depict each bodily component. This elliptical modeling approach is crucial for enabling independent tracking of the movement of each body part. Fig. 10 depicts the spanning area of all of the body parts discovered by the keypoints in the context of all clear, and move up actions performed by the subjects.

5) LANDMARKS

Landmarks are reference points used in computer vision tasks, serving as specific locations on an object. Various bodily components, including arms, knees, head, and ankles, are tracked by using landmarks. Following a person's movement through time and estimating their stance is made feasible by locating these points in a series of images. After that, the subject's action can be recognized [52].

Using the corresponding ellipsoids produced by the elliptical modeling phase, landmarks that span the boundaries of the distinct body components are located. The ellipsoids are individually scanned in a horizontal manner for this reason. Within an ellipsoid, the interior is represented by a black region. The right border of the ellipsoid is therefore indicated by a point's shift from a high value to a low value. In contrast, a change from a low value to a high value denotes that the point is part of the ellipsoid's left boundary. The mathematical relations for the right and left boundaries of the ellipsoids are given by:

$$RB = rbp_1, rbp_2, \ldots, rbp_m \tag{16}$$

$$LB = lbp_1, lbp_2, \ldots, lbp_n \tag{17}$$

In this context, m denotes the gross points belonging to the right border, and RB denotes set of points that make up the right border. Similar to this, LB denotes the collection of points that make up the border on left, and n denotes the gross points in the left border. The next step is to calculate the local minima and local maxima for both borders after separating the boundary points on left and right sides. if p_{i+1}'s slope is less than zero and p_i's slope is larger than or equal to zero, then p_i is regarded as a local maximum at either the right or left boundary. Similar to this, if the slope at a point p_i is less than or equal to zero and the slope at p_{i+1} is higher than zero, the point p_i is regarded as a local minimum. Mathematically,

$$max = \left\{p_i | p_i{}' \geq 0 \; and \; p_{i+1}{}' < 0\right\} \tag{18}$$

$$min = \left\{p_i | p_i{}' \leq 0 \; and \; p_{i+1}{}' > 0\right\} \tag{19}$$

The slopes of points p_i and p_{i+1} is represented in the preceding equation by $p_i\prime$ and $p_{i+1}\prime$, respectively. These local minimums and maximums serve as the landmarks for the specific body component being analyzed and are tracked from one frame to the next and recorded in a vectorized format. Fig. 11 illustrates the landmarks corresponding to different actions where the orange boxes represent the local maximum points of the body part while the cyan boxes represent the local minimum points for the same body part. These landmarks are spotted for each body part, discovered in the previous stages.

E. FEATURE ANALYSIS AND OPTIMIZATION

For the feature optimization, the naïve Bayes algorithm was implemented as a feature optimizer. Naïve Bayes is a probabilistic model that calculates the probability of a class given a set of features. It can be used to select the most informative features that are relevant to a classification task. This is done by calculating the conditional probability of each feature given the class label and then selecting the features having the highest probabilities [53]. First, the prior probability of the class is calculated using Eq. 20.

$$P_C = \frac{N(C)}{N} \tag{20}$$

where Nc is the number of samples belonging to class C, N is the total number of samples, and Pc is the probability of class C. For feature optimization, the probability of a class is calculated by taking each of the features into consideration one by one.

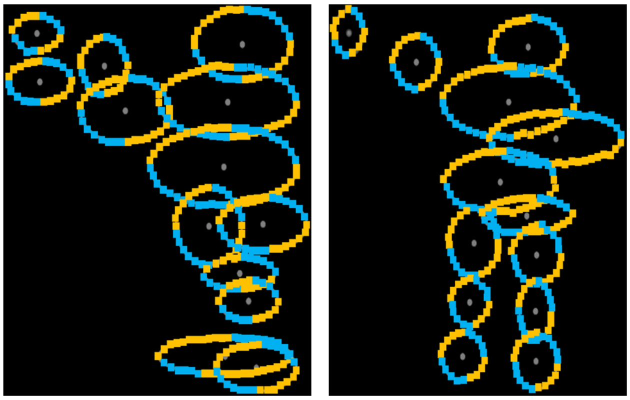

FIGURE 11. Landmarks for (a) Landing direction, and (b) All clear.

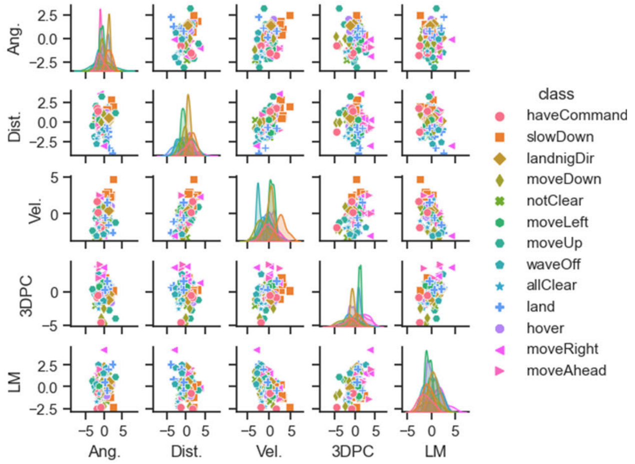

FIGURE 12. Pair plot for the extracted features with respect to all thirteen classes of the UAV-Gesture dataset.

$$P\left(f \mid C\right) = \frac{N(C,f)}{N(C)} \tag{21}$$

where $P(f|C)$ is the conditional probability of class C given feature f, and $N(C, f)$ is the number of samples containing feature f and class C. The final probability of the class considering a given feature is given by:

$$P\left(C \mid x\right) = P_C * P(f \mid C) \tag{22}$$

where $P\left(C \mid x\right)$ is the probability of class C given data point x. The mutual information and correlation between each feature and the class label is computed, and the top k features having better combination of mutual information and correlation are selected. Fig. 12 shows the pair plot of the features extracted in this framework. The diagonal of the plot shows the kernel density estimates of the features that show the separability of the classes with respect to that particular feature. While the off-diagonal scatter plots show the separability of the classes

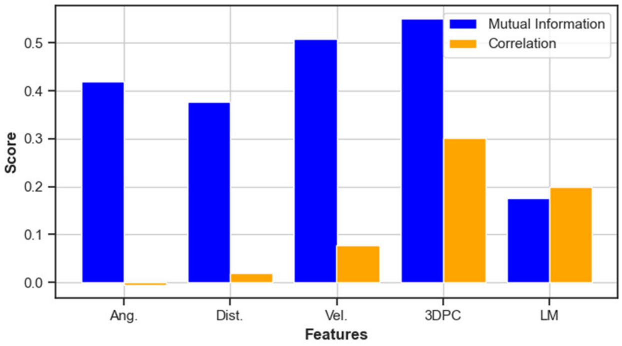

FIGURE 13. Mutual information and Pearson correlation plot for the extracted features.

TABLE 1. Proposed CNN architecture for the action classification.

Layer	Output Shape	Parameters
Conv2D	(None, 62, 62, 62)	320
MaxPooling2D	(None, 31, 31, 32)	0
Conv2D	(None, 29, 29, 64)	18496
MaxPooling2D	(None, 14, 14, 64)	0
Flatten	(None, 12544)	0
Dense	(None, 512)	6423040
Dense	(None, 13)	6669

Total parameters: 6448525 (24.60 MB)
Trainable parameters: 6448525 (24.60 MB)
Non-trainable parameters: 0 (0.00 Byte)

based on two features that are at x-axis and y-axis for the respective scatter plot.

We have also shown the mutual information and correlation of the features in Fig.13 where blue bars represent the mutual information of the features and the classes while the orange bars represent the Pearson correlation between the features and the classes.

F. ACTION CLASSIFICATION USING CNN

For the classification, a CNN is utilized [54]. The generic equation for the convolutional operation in a CNN is given below.

$$Y_{ijk} = \sum\nolimits_{p=0}^{k-1} \sum\nolimits_{q=0}^{k-1} \sum\nolimits_{c=0}^{c_{in}-1} W_{pqck} X_{i+p,j+q,c} + b_k \tag{23}$$

where X denotes the input matrix, W represents the weights, b represents the bias and Y is the output of the convolutional layer. Table 1 shows the proposed architecture of CNN for the classification of human actions. The features are transformed into a shape of (64, 64, 1) and sent at the input of CNN model. It first applied 32 filters of size (3, 3) and stride of 1. Then a max pooling layer with a size (2, 2) was applied that reduced the size of the input to (31, 31, 32). After that, 64 convolutional filters with a size of (3, 3) were applied with a stride of 1 followed by a max pooling layer of size (2, 2). The resulting size at this stage was (14, 14, 64). Afterwards, a flatten layer followed by the dense layer was applied. Finally, softmax generated the probability distribution for the final prediction.

IV. EXPERIMENTAL SETTINGS AND ANALYSIS

In this study, the experiments were conducted on a laptop with the following specifications: a 64-bit version of the Windows 10 operating system, an Intel®CoreTM i7-7500U CPU running at 2.70 GHz and 2.90 GHz, 16.0 GB of memory, and Visual Studio Code as the development environment. To evaluate the performance of the proposed system, three benchmark human action recognition datasets were utilized, including, the UAVGesture, the DroneAction, and the UAVHuman dataset. These datasets consist of recorded RGB videos captured from various perspectives using a drone camera. By implementing 10-fold cross-validation method, we ensured that the research findings were reliable.

A. UAV-GESTURE DATASET

The UAV-Gesture dataset is an outdoor dataset that consists of 119 high definition RGB videos. The videos constitute 37,151 RGB frames in total. The videos were recorded by a drone camera while subjects performed actions to command unmanned aerial vehicles (UAVs) or helicopters. In the dataset, there were a total of 13 different actions including, *hover, move to left, move upward, move ahead, land, move downward, slow down, move to right, wave off, all clear, have command, not clear, and landing direction* [55].

B. DRONE-ACTION DATASET

The Drone-Action dataset is a one-of-a-kind dataset that was collected exclusively for human action detection using drone video. It consists of a total of 66,919 frames and 240 video segments. A total of 10 individuals carried out 13 distinct tasks in the videos. Their activities were captured in a variety of ways, including from the front, from the side, and even as the drone followed the person as they performed the action. The recorded actions by the team included, *jogging-follow, stabbing, hitting-with-stick, clapping, walking-side, punching, kicking, jogging-side, hitting-with-bottle, waving hands, running-side, walking-follow, and running-follow* [56].

C. UAV-HUMAN DATASET

The UAV-Human Dataset serves as a comprehensive benchmark for UAV-based human behavior understanding, with a specific focus on action recognition. It comprises a vast collection of 67,428 multi-modal video sequences captured by a flying UAV in diverse urban and rural districts [56]. The dataset offers an extensive range of variations, including different subjects, backgrounds, illuminations, weathers, occlusions, camera motions, and UAV flying attitudes, providing a challenging and realistic environment for research. For action recognition purposes, the dataset includes 119 subjects performing various actions such as *sit, applaud, stand, wave, kick, thumbs-up, thumbs-down, salute, run, walk, all-clear, not-clear, have-command, move-left, and move-right.*

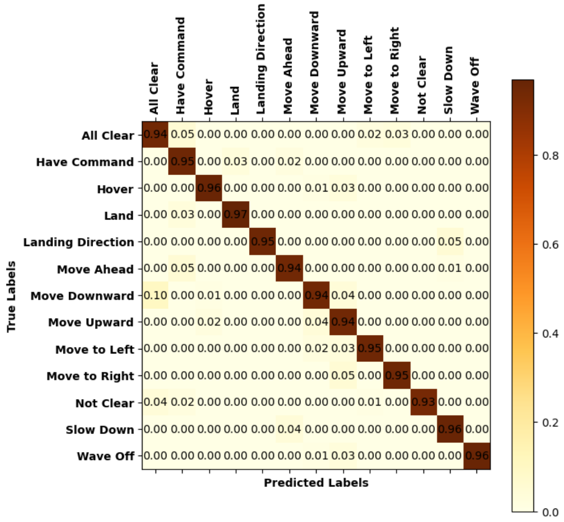

FIGURE 14. Confusion matrix for the UAVGesture dataset.

These fifteen classes have been chosen to cater specifically to human action recognition tasks.

D. RESULTS

On three different datasets, including the UAVGesture, the DroneAction, and the UAVHuman dataset, the proposed approach, which combines quick shift segmentation, EM-GMM-based elliptical modeling, multi-feature extraction, naive Bayes feature optimization, and CNN classification, was painstakingly assessed.

To capture reliable results for human action classification, the experiment was repeated three times. The confusion matrix that represents the ratio of true positives and true negatives for each action class of the dataset is generated for each of the datasets. The confusion matrices for UAVGesture dataset, DroneAction dataset, and UAVHuman dataset are shown in Fig. 14, Fig. 15, and Fig.16 respectively. The confusion matrices display respective action recognition accuracy of 0.95, 0.90, and 0.44. In addition, the system was assessed by its classification report, which consisted of the precision, recall, and F1-score of the system on the mentioned datasets. Table 2 shows the mean precision, mean recall, and mean F1-score for the UAV-Gesture dataset i.e., 0.96, 0.95, and 0.94, respectively. Mean recall, precision, and F1-score for Drone-Action dataset are 0.90, 0.89, and 0.89, respectively, as demonstrated in Table 3. A classification report of system's performance on UAVHuman dataset is shown in Table 4 that demonstrates the mean recall, mean precision, and mean F1-score of the system to be 0.44, 0.43, and 0.43, respectively.

The system's performance has been compared with those of other cutting-edge systems aimed at achieving the same goal. The suggested system performed far better than the leading-edge systems. Table 5 contains the specifics of the system comparison.

To further affirm the performance of the system, we have compared the training and validation results while

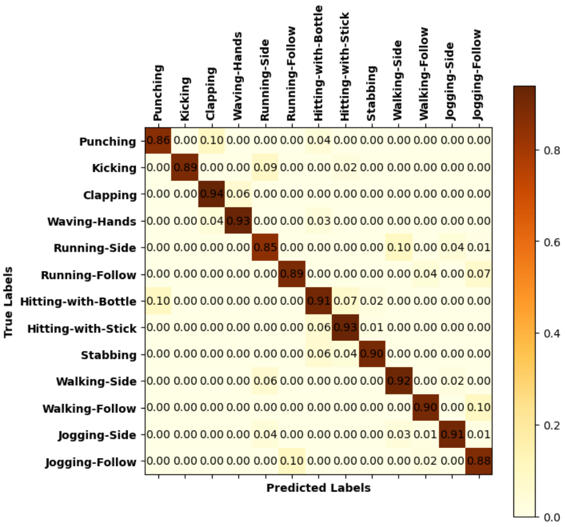

FIGURE 15. Confusion matrix for the DroneAction dataset.

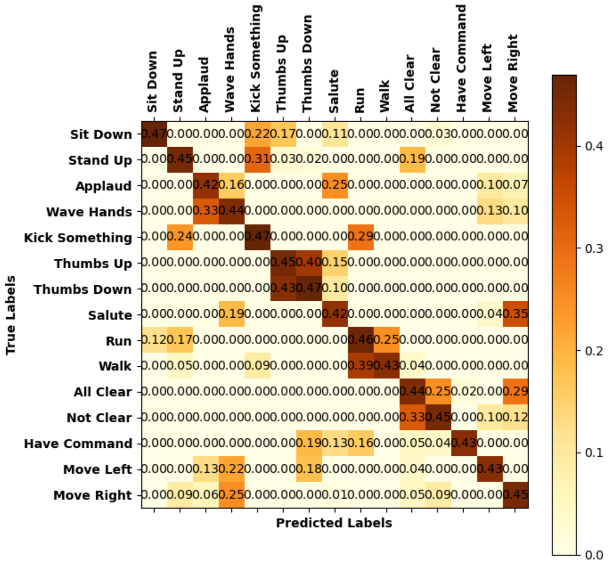

FIGURE 16. Confusion matrix for the UAV-Human dataset.

implementing deep learning architectures other than CNN for the action classification. The deep architectures include, a DenseNet, EfficientNet, MobileNet, and ResNet. The performance comparison is given in Table 6.

The common point among all of the deep architectures other than CNN, was that their time complexity was very high as compared to the CNN architecture that we utilized. Secondly, other deep models got overfit on the data and generated similar or worse results as compared to the chosen architecture of CNN. That makes CNN the best choice for our system architecture.

V. DISCUSSIONS

The proposed system uses deep learning architecture and makes the algorithm depend only on RGB data while

TABLE 2. Classification report on the UAV-Gesture dataset.

A/C	Precision	Recall	F1 score
Acl	0.95	0.95	0.93
Hcd	0.95	0.94	0.94
Hov	0.96	0.96	0.95
Lan	0.96	0.95	0.94
Ldr	0.95	0.95	0.95
Mah	0.96	0.95	0.94
Mdn	0.95	0.95	0.93
Mup	0.96	0.94	0.94
Mvl	0.94	0.94	0.94
Mvr	0.95	0.93	0.93
Ncl	0.95	0.93	0.94
Sdn	0.97	0.97	0.95
Wav	0.98	0.95	0.96
Mean	**0.96**	**0.95**	**0.94**

*Acl = all clear; Hcd = have command; Hov = hover; Lan = land; Ldr = landing direction; Mah = move ahead; Mdn = move down; Mup = move up; Mvl = move left; Mvr = move right; Ncl = not clear; Sdn = slow down; Wav = wave off

TABLE 3. Classification report on the Drone-Action dataset.

A/C	Precision	Recall	F1 score
Jog-f	0.89	0.88	0.87
Sta	0.90	0.89	0.89
Hws	0.92	0.92	0.91
Cla	0.95	0.92	0.92
Wal-s	0.93	0.91	0.91
Pun	0.86	0.86	0.84
Kic	0.90	0.88	0.88
Jog-s	0.92	0.92	0.90
Hwb	0.90	0.90	0.90
Wav	0.92	0.90	0.89
Run-s	0.85	0.85	0.84
Wal-f	0.90	0.90	0.89
Run-f	0.88	0.89	0.88
Mean	**0.90**	**0.89**	**0.89**

*Jog-f = jogging follow; Sta = stabbing; Hws = hitting with stick; Cla = clapping; Wal-s = walking side; Pun = punching; Kic = kicking; Jog-s = jogging side; Hwb = hitting with bottle; Wav = wave hands; Run-s = running side; Wal-f = walking follow; Run-f = running follow

TABLE 4. Classification report on the UAV-Human dataset.

A/C	Precision	Recall	F1 score
Sit	0.47	0.46	0.44
App	0.42	0.39	0.39
Sta	0.45	0.43	0.43
Wav	0.45	0.45	0.44
Kic	0.48	0.47	0.47
Thu-u	0.44	0.44	0.44
Thu-d	0.47	0.46	0.46
Sal	0.42	0.41	0.41
Run	0.46	0.45	0.45
Wal	0.44	0.44	0.41
Acl	0.44	0.44	0.41
Ncl	0.46	0.43	0.43
Hcd	0.45	0.43	0.43
Mvl	0.43	0.43	0.42
Mvr	0.46	0.44	0.44
Mean	**0.44**	**0.43**	**0.43**

*Sit = sitting; App = applaud; Sta = standing; Wav = wave; Kic = kick; Thu-u = thumbs-up; Thu-d = thumbs-down; Sal = salute; Wal = walk; Acl = all-clear; Ncl = not-clear; Hcd = have-command; Mvl = move-left; Mvr = move-right

TABLE 5. Comparison with conventional systems.

Methods	UAV Gesture	Drone Action	UAV Human
P-CNN [56]	-	0.75	-
SWTF + Pose-Stream [58]	-	0.78	-
P-CNN [59][55]	0.91	-	-
MLP_7j [60]	0.94	-	-
Baseline (SGN) [61]	-	-	0.39
MSST-RT [62]	-	-	0.41
Multi-feature + CNN (Proposed)	**0.95**	**0.90**	**0.44**

TABLE 6. Comparison with state-of-the-art deep architectures.

Methods	UAV Gesture (train/val.)	Drone Action (train/val.)	UAV Human (train/val.)
DenseNet-121 [63]	0.99/0.95	0.98/0.90	0.93/0.43
EfficientNet [64]	0.96/0.95	0.94/0.90	0.96/0.42
MobileNet [65]	0.93/0.64	0.97/0.50	0.95/0.36
ResNet-50 [66]	0.98/0.47	0.93/0.49	0.99/0.44
Multi-feature + CNN (Proposed)	**0.97/0.95**	**0.91/0.90**	**0.47/0.44**

excluding the use of depth data. The exclusion of depth data not only makes the proposed system more versatile and adaptable but also enables it to have broader applicability when depth information is not available or applicable. Another notable aspect of the system is the use of quick shift segmentation that proved its significance by excellently segmenting out humans from the videos and meanwhile keeping the system from inducting obnoxious time complexity. The use of CNNs in the proposed system optimized feature extraction, improved action classification, and enhanced generalization and robustness in recognizing human actions in aerial videos. The proposed system demonstrated superior performance compared to the previous system, particularly in accurate human segmentation, feature extraction, and action classification capabilities.

Although the system was able to outperform the state-of-the-art methods with respect to each of the dataset used in this study but the accuracy of the proposed system was significantly lower in case of the UAVHuman dataset. It was due to the fact that the complexity and variability of the UAVHuman dataset was much higher than the other two datasets as it was recorded in a variety of urban and rural areas. Secondly, 119 subjects participated in the collection of the data, which also adds more complexity to the dataset by including different gaits and body postures of the subjects.

VI. CONCLUSION

An action recognition system for humans was proposed in this article. The system used body key-point extraction, quick shift segmentation, EM-GMM-based elliptical modeling, feature optimization using the naïve Bayes feature optimizer, and CNN classification. It is intended for use in a variety of reality-based scenarios, including gesture control systems, sports analysis, robot-human collaboration, surveillance systems, entertainment and gaming, and smart homes. Although the system dropped its accuracy in recognizing some of the activities but it was still able to outperform the most cutting-edge techniques in use.

Our future endeavors will focus on improving the performance of the system on high complexity datasets such as the UAVHuman dataset so as to improve the overall generalizability of the framework. Moreover, we would be exploring alternative features suitable for multi-human-based systems. We also intend to delve into novel approaches and methodologies that effectively capture and analyze the interactions and dynamics among multiple individuals.

ACKNOWLEDGMENT

Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R97), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

REFERENCES

[1] H. Hwang, C. Jang, G. Park, J. Cho, and I.-J. Kim, "ElderSim: A synthetic data generation platform for human action recognition in eldercare applications," *IEEE Access*, vol. 11, pp. 9279–9294, 2023.

[2] A. Jalal, M. A. K. Quaid, and M. A. Sidduqi, "A triaxial acceleration-based human motion detection for ambient smart home system," in *Proc. 16th Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Jan. 2019, pp. 353–358.

[3] A. Jalal, M. A. K. Quaid, and K. Kim, "A wrist Worn acceleration based human motion analysis and classification for ambient smart home system," *J. Electr. Eng. Technol.*, vol. 14, no. 4, pp. 1733–1739, Jul. 2019.

[4] M. Batool, A. Jalal, and K. Kim, "Telemonitoring of daily activity using accelerometer and gyroscope in smart home environments," *J. Electr. Eng. Technol.*, vol. 15, no. 6, pp. 2801–2809, Nov. 2020.

[5] A. Jalal, M. Batool, and K. Kim, "Sustainable wearable system: Human behavior modeling for life-logging activities using K-Ary tree hashing classifier," *Sustainability*, vol. 12, no. 24, p. 10324, Dec. 2020.

[6] M. Javeed and A. Jalal, "Deep activity recognition based on patterns discovery for healthcare monitoring," in *Proc. 4th Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2023, pp. 1–6.

[7] M. Muneeb, H. Rustam, and A. Jalal, "Automate appliances via gestures recognition for elderly living assistance," in *Proc. 4th Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2023, pp. 1–6.

[8] K. H. Cheong, S. Poeschmann, J. W. Lai, J. M. Koh, U. R. Acharya, S. C. M. Yu, and K. J. W. Tang, "Practical automated video analytics for crowd monitoring and counting," *IEEE Access*, vol. 7, pp. 183252–183261, 2019.

[9] K.-P. Chou, M. Prasad, D. Wu, N. Sharma, D.-L. Li, Y.-F. Lin, M. Blumenstein, W.-C. Lin, and C.-T. Lin, "Robust feature-based automated multi-view human action recognition system," *IEEE Access*, vol. 6, pp. 15283–15296, 2018.

[10] A. Jalal and Y. Kim, "Dense depth maps-based human pose tracking and recognition in dynamic scenes using ridge data," in *Proc. 11th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2014, pp. 119–124.

[11] P. Wei, Y. Zhao, N. Zheng, and S.-C. Zhu, "Modeling 4D human-object interactions for joint event segmentation, recognition, and object localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1165–1179, Jun. 2017.

[12] A. Jalal, Y. Kim, and D. Kim, "Ridge body parts features for human pose estimation and recognition from RGB-D video data," in *Proc. 5th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2014, pp. 1–6.

[13] A. Jalal, N. Khalid, and K. Kim, "Automatic recognition of human interaction via hybrid descriptors and maximum entropy Markov model using depth sensors," *Entropy*, vol. 22, no. 8, p. 817, Jul. 2020.

[14] R. Al-Akam and D. Paulus, "Local feature extraction from RGB and depth videos for human action recognition," *Int. J. Mach. Learn. Comput.*, vol. 8, no. 3, pp. 274–279, Jun. 2018.

[15] U. Azmat and A. Jalal, "Smartphone inertial sensors for human locomotion activity recognition based on template matching and codebook generation," in *Proc. Int. Conf. Commun. Technol. (ComTech)*, Sep. 2021, pp. 109–114.

[16] M. Waheed, M. Javeed, and A. Jalal, "A novel deep learning model for understanding two-person interactions using depth sensors," in *Proc. Int. Conf. Innov. Comput. (ICIC)*, Nov. 2021, pp. 1–8.

[17] J. Ji, S. Buch, A. Soto, and J. C. Niebles, "End-to-end joint semantic segmentation of actors and actions in video," in *Proc. ECCV*, 2018, pp. 734–749.

[18] A. Jalal, M. Mahmood, and A. S. Hasan, "Multi-features descriptors for human activity tracking and recognition in indoor-outdoor environments," in *Proc. 16th Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Jan. 2019, pp. 371–376.

[19] M. Mahmood, A. Jalal, and H. A. Evans, "Facial expression recognition in image sequences using 1D transform and Gabor wavelet transform," in *Proc. Int. Conf. Appl. Eng. Math. (ICAEM)*, Sep. 2018, pp. 1–6.

[20] R. Divya Rani and C. J. Prabhakar, "Human action recognition by concatenation of spatio-temporal 3D SIFT and CoHOG descriptors using bag of visual words," in *Proc. Int. Conf. Distrib. Comput., VLSI, Electr. Circuits Robot.*, Shivamogga, India, Oct. 2022, pp. 1–6.

[21] M. Pervaiz and A. Jalal, "Artificial neural network for human object interaction system over aerial images," in *Proc. 4th Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2023, pp. 1–6.

[22] M. Mahmood, A. Jalal, and K. Kim, "WHITE STAG model: Wise human interaction tracking and estimation (WHITE) using spatio-temporal and angular-geometric (STAG) descriptors," *Multimedia Tools Appl.*, vol. 79, nos. 11–12, pp. 6919–6950, Mar. 2020.

[23] M. A. K. Quaid and A. Jalal, "Wearable sensors based human behavioral pattern recognition using statistical features and reweighted genetic algorithm," *Multimedia Tools Appl.*, vol. 79, nos. 9–10, pp. 6061–6083, Mar. 2020.

[24] A. Nadeem, A. Jalal, and K. Kim, "Human actions tracking and recognition based on body parts detection via artificial neural network," in *Proc. 3rd Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2020, pp. 1–6.

[25] R. Cui, G. Hua, A. Zhu, J. Wu, and H. Liu, "Hard sample mining and learning for skeleton-based human action recognition and identification," *IEEE Access*, vol. 7, pp. 8245–8257, 2019.

[26] H. Rahmani and M. Bennamoun, "Learning action recognition model from depth and skeleton videos," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5833–5842.

[27] U. Azmat, S. S. Alotaibi, N. A. Mudawi, B. I. Alabduallah, M. Alonazi, A. Jalal, and J. Park, "An elliptical modeling supported system for human action deep recognition over aerial surveillance," *IEEE Access*, vol. 11, pp. 75671–75685, 2023.

[28] J. Arunnehru, S. Thalapathiraj, R. Dhanasekar, L. Vijayaraja, R. Kannadasan, A. A. Khan, M. A. Haq, M. Alshehri, M. I. Alwanain, and I. Keshta, "Machine vision-based human action recognition using spatio-temporal motion features (STMF) with difference intensity distance group pattern (DIDGP)," *Electronics*, vol. 11, no. 15, p. 2363, Jul. 2022.

[29] H. Chen, G. Wang, J.-H. Xue, and L. He, "A novel hierarchical framework for human action recognition," *Pattern Recognit.*, vol. 55, pp. 148–159, Jul. 2016.

[30] X. Zhen and L. Shao, "Action recognition via spatio-temporal local features: A comprehensive study," *Image Vis. Comput.*, vol. 50, pp. 1–13, Jun. 2016.

[31] X. Yang and Y. Tian, "Super normal vector for human activity recognition with depth cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 1028–1039, May 2017.

[32] K. Kim, A. Jalal, and M. Mahmood, "Vision-based human activity recognition system using depth silhouettes: A smart home system for monitoring the residents," *J. Electr. Eng. Technol.*, vol. 14, no. 6, pp. 2567–2573, Nov. 2019.

[33] A. Shahroudy, T.-T. Ng, Q. Yang, and G. Wang, "Multimodal multipart learning for action recognition in depth videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2123–2129, Oct. 2016.

[34] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," 2016, *arXiv:1603.07772*.

[35] Y. Li, W. Li, V. Mahadevan, and N. Vasconcelos, "VLAD3: Encoding dynamics of deep features for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1951–1960.

[36] Y. Shi, Y. Tian, Y. Wang, and T. Huang, "Sequential deep trajectory descriptor for action recognition with three-stream CNN," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1510–1520, Jul. 2017.

[37] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1110–1118.

[38] R. V. Nezafat, B. Salahshour, and M. Cetin, "Classification of truck body types using a deep transfer learning approach," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, Nov. 2018, pp. 3144–3149.

[39] M. J. Rashti and S. E. Alavi, "Human action recognition in video using DB-LSTM and ResNet," in *Proc. 6th Int. Conf. Web Res. (ICWR)*, Tehran, Iran, 2020, pp. 133–138.

[40] K. Muhammad, A. Ullah, A. S. Imran, M. Sajjad, M. S. Kiran, G. Sannino, and V. H. C. de Albuquerque, "Human action recognition using attention based LSTM network with dilated CNN features," *Future Gener. Comput. Syst.*, vol. 125, pp. 820–883, Jan. 2021.

[41] Y. He, Y. Zheng, Y. Zhao, Y. Ren, J. Lian, and J. Gee, "Retinal image denoising via bilateral filter with a spatial kernel of optimally oriented line spread function," *Comput. Math. Methods Med.*, vol. 2017, pp. 1–13, Jan. 2017.

[42] Q. Zhu, D. Wu, Y. Xie, and L. Wang, "Quick shift segmentation guided single image haze removal algorithm," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2014, pp. 113–117.

[43] J. Mille, A. Leborgne, and L. Tougne, "Euclidean distance-based skeletons: A few notes on average outward flux and ridgeness," *J. Math. Imag. Vis.*, vol. 61, no. 3, pp. 310–330, Mar. 2019.

[44] L. J. Latecki, Q.-N. Li, X. Bai, and W.-Y. Liu, "Skeletonization using SSM of the distance transform," in *Proc. IEEE Int. Conf. Image Process.*, 2007, pp. 349–352.

[45] T.-Q. Yan and C.-X. Zhou, "A continuous skeletonization method based on distance transform," in *Proc. ICIC*, 2012, pp. 251–258.

[46] L. Serino, C. Arcelli, and G. S. Baja, "From the zones of influence of skeleton branch points to meaningful object parts," in *Proc. DGCI*, 2013, pp. 131–142.

[47] M. Batool, A. Jalal, and K. Kim, "Sensors technologies for human activity analysis based on SVM optimized by PSO algorithm," in *Proc. Int. Conf. Appl. Eng. Math. (ICAEM)*, Aug. 2019, pp. 145–150.

[48] H. Xing and D. Burschka, "Skeletal human action recognition using hybrid attention based graph convolutional network," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2022, pp. 3333–3340.

[49] X. Wang, H. Chen, and L. Wu, "Feature extraction of point clouds based on region clustering segmentation," *Multimedia Tools Appl.*, vol. 79, nos. 17–18, pp. 11861–11889, May 2020.

[50] Daavoo. (2023). *Pyntcloud v0.3.1. Commit: 32eea8f*. GitHub. [Online]. Available: https://github.com/daavoo/pyntcloud

[51] A. Arif and A. Jalal, "Automated body parts estimation and detection using salient maps and Gaussian matrix model," in *Proc. Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Jan. 2021, pp. 667–672.

[52] F. Rajbdad, M. Aslam, S. Azmat, T. Ali, and S. Khattak, "Automated fiducial points detection using human body segmentation," *Arabian J. Sci. Eng.*, vol. 43, no. 2, pp. 509–524, Feb. 2018.

[53] D. Mittal and M. Bala, "Hybrid feature selection approach using bacterial foraging algorithm guided by naive Bayes classification," in *Proc. 8th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2017, pp. 1–7.

[54] M. Kanthi, T. H. Sarma, and C. S. Bindu, "A 3D-deep CNN based feature extraction and hyperspectral image classification," in *Proc. IEEE India Geosci. Remote Sens. Symp. (InGARSS)*, Dec. 2020, pp. 229–232.

[55] A. Perera, Y. Law, and J. Chahl, "UAV-GESTURE: A dataset for UAV control and gesture recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2019, p. 11130.

[56] A. G. Perera, Y. W. Law, and J. Chahl, "Drone-action: An outdoor recorded drone video dataset for action recognition," *Drones*, vol. 3, no. 4, p. 82, Nov. 2019.

[57] T. Li, J. Liu, W. Zhang, Y. Ni, W. Wang, and Z. Li, "UAV-Human: A large benchmark for human behavior understanding with unmanned aerial vehicles," 2021, *arXiv:2104.00946*.

[58] S. K. Yadav, A. Luthra, E. Pahwa, K. Tiwari, H. Rathore, H. M. Pandey, and P. Corcoran, "DroneAttention: Sparse weighted temporal attention for drone-camera based activity recognition," *Neural Netw.*, vol. 159, pp. 57–69, Feb. 2023.

[59] G. Chéron, I. Laptev, and C. Schmid, "P-CNN: Pose-based CNN features for action recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3218–3226.

[60] C. Papaioannidis, D. Makrygiannis, I. Mademlis, and I. Pitas, "Learning fast and robust gesture recognition," in *Proc. 29th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2021, pp. 761–765.

[61] L. Xu, C. Lan, W. Zeng, and C. Lu, "Skeleton-based mutually assisted interacted object localization and human action recognition," *IEEE Trans. Multimedia*, early access, May 16, 2022, doi: 10.1109/TMM.2022.3175374.

[62] Y. Sun, Y. Shen, and L. Ma, "MSST-RT: Multi-stream spatial–temporal relative transformer for skeleton-based action recognition," *Sensors*, vol. 21, no. 16, p. 5339, Aug. 2021.

[63] G. Huang, S. Liu, L. van den Maaten, and K. Q. Weinberger, "CondenseNet: An efficient DenseNet using learned group convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2752–2761.

[64] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[65] A. Michele, V. Colin, and D. D. Santika, "MobileNet convolutional neural networks and support vector machines for palmprint recognition," *Proc. Comput. Sci.*, vol. 157, pp. 110–117, Jan. 2019.

[66] Z. Wu, C. Shen, and A. van den Hengel, "Wider or deeper: Revisiting the ResNet model for visual recognition," *Pattern Recognit.*, vol. 90, pp. 119–133, Jun. 2019.

USMAN AZMAT received the B.E. degree in mechatronics and control engineering from the University of Engineering and Technology Lahore, Lahore, Pakistan. He is currently pursuing the M.S. degree in artificial intelligence with Air University, Islamabad, Pakistan. He was a Microcontroller Programmer, from 2015 to 2020. Since 2020, he has been a Research Associate with Air University. His research interests include artificial intelligence, machine learning algorithms, deep learning classification, human locomotion analysis, inertial signals filtration, image and video processing, human action, and interaction recognition.

SAUD S. ALOTAIBI received the B.Sc. degree from King Abdulaziz University, in 2000, the master's degree in computer science from King Fahd University, Dhahran, in May 2008, and the Ph.D. degree in computer science from Colorado State University, Fort Collins, CO, USA, in August 2015, under the supervision of Dr. Charles Anderson. He was an Assistant Lecturer with Umm Al-Qura University, Mecca, Saudi Arabia, in July 2001, where he is currently an Assistant Professor of computer science. He was the Deputy of the IT-Center for EGovernment and Application Services, Umm Al-Qura University, in January 2009. From 2015 to 2018, he was with the Deanship of Information Technology to improve the IT services that are provided to Umm Al-Qura University. He is also with the Computer and Information College, as the Vice Dean for academic affairs. His research interests include AI, machine learning, natural language processing, neural computing IoT, knowledge representation, smart cities, wireless, and sensors.

MAHA ABDELHAQ (Member, IEEE) received the B.Sc. degree in computer science and the M.Sc. degree in securing wireless communications from The University of Jordan, Jordan, in 2006 and 2009, respectively, and the Ph.D. degree from the Faculty of Information Science and Technology, National University of Malaysia, Malaysia, in 2014. She is currently an Associate Professor with the College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Saudi Arabia. Her research interests include vehicular networks, MANET routing protocols, artificial immune systems, network security, and intelligent computational. She is a member of ACM and the International Association of Engineers (IAENG).

NAWAL ALSUFYANI is currently a Full Professor with Prince Sultan University.

MOHAMMAD SHORFUZZAMAN is currently an Associate Professor with Prince Sultan University.

AHMAD JALAL received the Ph.D. degree from the Department of Biomedical Engineering, Kyung Hee University, South Korea. He is currently an Associate Professor with the Department of Computer Science and Engineering, Air University, Pakistan. He is also a Postdoctoral Research Fellowship with POSTECH. His research interests include multimedia contents, artificial intelligence, and machine learning.

JEONGMIN PARK received the Ph.D. degree from the College of Information and Communication Engineering, Sungkyunkwan University, South Korea, in 2009. He is currently an Associate Professor with the Department of Computer Engineering, Tech University of Korea, South Korea. Before joining the Tech University of Korea, in 2014, he was a Senior Researcher with the Electronics and Telecommunications Research Institute (ETRI) and a Research Professor with Sungkyunkwan University. His research interests include high-reliable autonomic computing mechanism and human-oriented interaction systems.