Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.059224

ARTICLE

Drone-Based Public Surveillance Using 3D Point Clouds and Neuro-Fuzzy Classifier

Yawar Abbas[1], Aisha Ahmed Alarfaj[2], Ebtisam Abdullah Alabdulqader[3], Asaad Algarni[4], Ahmad Jalal[1,5] and Hui Liu[6,*]

[1]Faculty of Computing and AI, Air University, Islamabad, 44000, Pakistan
[2]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[3]Department of Information Technology, College of Computer and Information Sciences, King Saud University, Riyadh, 12372, Saudi Arabia
[4]Department of Computer Sciences, Faculty of Computing and Information Technology, Northern Border University, Rafha, 91911, Saudi Arabia
[5]Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul, 02841, Republic of Korea
[6]Cognitive Systems Lab, University of Bremen, Bremen, 28359, Germany

*Corresponding Author: Hui Liu. Email: hui.liu@uni-bremen.de

Received: 30 September 2024; Accepted: 13 January 2025; Published: 06 March 2025

ABSTRACT: Human Activity Recognition (HAR) in drone-captured videos has become popular because of the interest in various fields such as video surveillance, sports analysis, and human-robot interaction. However, recognizing actions from such videos poses the following challenges: variations of human motion, the complexity of backdrops, motion blurs, occlusions, and restricted camera angles. This research presents a human activity recognition system to address these challenges by working with drones' red-green-blue (RGB) videos. The first step in the proposed system involves partitioning videos into frames and then using bilateral filtering to improve the quality of object foregrounds while reducing background interference before converting from RGB to grayscale images. The YOLO (You Only Look Once) algorithm detects and extracts humans from each frame, obtaining their skeletons for further processing. The joint angles, displacement and velocity, histogram of oriented gradients (HOG), 3D points, and geodesic Distance are included. These features are optimized using Quadratic Discriminant Analysis (QDA) and utilized in a Neuro-Fuzzy Classifier (NFC) for activity classification. Real-world evaluations on the Drone-Action, Unmanned Aerial Vehicle (UAV)-Gesture, and Okutama-Action datasets substantiate the proposed system's superiority in accuracy rates over existing methods. In particular, the system obtains recognition rates of 93% for drone action, 97% for UAV gestures, and 81% for Okutama-action, demonstrating the system's reliability and ability to learn human activity from drone videos.

KEYWORDS: Activity recognition; geodesic distance; pattern recognition; neuro fuzzy classifier

1 Introduction

Recognizing human activity from video footage captured from a UAV is a challenging work that entails the analysis of video footage to determine the movement and activity of people from the video stream without intervention. This field of interest finds application in several disciplines, such as video surveillance system improvement, detection of human movement, sports performance, natural human-robot interaction, and rehabilitation [1]. For instance, in rehabilitation healthcare, such as for patients suffering from stroke and affected body extremities, this technology has been shown to help in improving rehabilitation outcomes and

reducing disability. In video surveillance, recognizing activity can assist in alerting security risks within an area, for instance, a person who is a danger to people or even one holding a weapon, thus enhancing safety and minimizing the incidences of criminal activities. The perception of human motion enables robots to analyze and interact with people appropriately in human-robot interactions. In the area of medicine, this technique helps coaches evaluate players' medical status, physical condition, or play, and the efficiency of the teams that they represent, allowing them to make proper decisions and select better teams. Besides, in gaming and entertainment, activity recognition improves the game and entertainment experience to be more interactive. However, activity recognition has some challenges, as follows: the changes in a human pose to different frames, object appearance, movement, speed calculation, and time limitations are some of the conditions that make it challenging to develop a satisfactory algorithm that performs desirable with consistency in different environments. Labeled data for human activity recognition are expensive and time-consuming, respectively, and existing datasets for model training are scarce. Because of the constantly changing backdrop, videos taken using UAVs present other challenges besides motion differences and different camera angles. The requirement for real-time performance, especially for its use in surveillance systems and robotics, further enhances the challenge.

Previous systems employed regular computer vision methodologies along with the algorithms of machine learning on the RGB and depth video stream. This system involves the following steps: splitting frames, reducing noise with bilateral filters, extracting regions using Simple Linear Iterative Clustering (SLIC) segmentation, and estimating body joints through expectation-maximization based on the Gaussian mixture model EM-GMM (Expectation Maximization Gaussian Mixture Model). Still, depth information disadvantages these systems because it is not always practical when dealing with real-world data and environment. We introduce a new method for recognizing human activity from aerial RGB videos, which has two advantages over the previous methods: it does not require depth information. To eliminate depth data and improve accuracy and performance, our proposed system uses a Neuro-Fuzzy Classifier (NFC). It includes processes such as the change of format of aerial video from RGB to frames. This bilateral filter reduces noises but has low computational complexity and separations of background effects. Human detection is done by YOLO, and feature extraction is performed using joint angles, displacement, and velocity of the feature points, Histogram of Oriented Gradient (HOG), three-dimensional coordinates of the feature points, and geodesic Distance. Quadratic Discriminant Analysis (QDA) is used as an optimizer. Here, our contribution is better than other systems in enhancing the overall detected human recognition and action identification. To summarize, the primary findings of our research can be outlined as follows:

- In contrast to previous approaches that use depth information, ours improves human detection and activity recognition solely based on the RGB channel. Since additional sensors are not required, the system is more efficient in low-resource conditions while enhancing the detection rates without compromising speed.
- The entire process is elaborated at a finer grain. It starts with feature extraction and ends with action classification. The Neuro-Fuzzy Classifier (NFC) is used to improve the classifier's learning capability and address the complexities of human actions during dynamic environmental scenarios.
- With the help of the YOLO algorithm, our system guarantees accurate identification of human subjects and allows for successful tracking even in cases of motion or partial occlusion. The recognition of actions actually builds upon YOLO's rapid and precise identification of humans, which greatly increases the system's efficiency.
- The integration of Histogram of Oriented Gradients (HOG) and geodesic distance algorithms helps to achieve a higher level of motion analysis and distinguish between human postures and actions with

increased precision. This sort of integration also improves the system's capacity to identify not only gross movements but even more precise movements.
- Optimization is done by (QDA). This improvement leads to higher recognition rates, much like an increase in classification accuracy, as opposed to previous approaches.
- The main contribution of our proposed system is that we reduce the misclassification between similar classes by using QDA. Reduce the issue of dynamic background by using preprocessing steps. Our system also gets higher accuracy.

2 Literature Review

Recent research in the area of computer vision has brought significant improvement in the various algorithms used in the identification of human activities. According to the literature, there are two main research areas in this study.

2.1 Preprocessing

Our system's preprocessing methodological approach outperforms Arunnehru et al. [2]. Compared to the previous study, which focuses on grayscale conversion and motion feature extraction regarding frame differences, the present system encompasses a more elaborate workflow. We start by preprocessing drone-acquired videos into frames to match those used by image-based algorithms. To improve image quality, we use a bilateral filter that specifically keeps edges sharp and smooth noise at the same time these enhancements allow our system to pay more attention to exhaustive discriminating regions on the body, leading to better reliability and robustness in action recognition applications. Compared to the previous system [3], where the gamma correction was employed to accentuate humans by nonlinear brightness control, the proposed preprocessing approach provides better and finer steps for handling higher-quality inputs for further analysis. Whereas in the prior system, the image is performed gamma correction for blurring the background in order to highlight the human features, here, to eliminate the undesirable post-processing noise effect, a bilateral filter is used that preserves the edges and details of the human subject. This reduces the object-background contrast for the benefit of getting a higher contrast of the subject over its background, which is enhanced using grayscale conversion and background subtraction. Through these steps, preprocessing also ensures that the essential spatial details are retained while, at the same time, eliminating most of the noise with such an outcome, giving our following feature extraction and classification processes a reliable foundation.

2.2 Human Detection and Key Point Detection

The authors in [4] have also developed a deep LSTM (Long Short-Term Memory) network topology wherein all links within the temporal graph are fully connected to recognize human activity from skeletal data. Their study underscored the fact that the articulation of skeletal joints always surfaces essential aspects of human activity. Unlike previous approaches, our method deals with RGB videos, which sidesteps the representation in the skeleton domain and allows us to extract more visual information from aerial images. To solve the issues of previous methods where most metrics were based on short temporal data and failed to capture long-range dependency. We use 15 landmarks of the human body to detect the motion of the human body which helps us to determine human activity. The Quick Shift algorithm [5] is well suited to segment the regions based on color or texture similarity. Our method of training increases the detection of confident bounding boxes surrounding humans and eliminates the instances detected as non-human, enhancing accurate localization. This makes it more appropriate for complex tasks such as the detection of humans in aerial imagery. Shi et al. [6] present a method for the recognition of human activity in videos that have been recorded with the help of a depth camera. It is different from their Super-Normal Vector method, which

is based on the integration of multiple low-level polynomials into discriminative representation. However, this approach depends on in-depth information only and uses RGB data in a limited manner. While our system considers RGB videos, which utilize color and texture as well as depth maps for understanding human activity. Furthermore, the proposed method of work introduced here applies the use of joint regression-based learning to discuss the dynamic appearance of the individual instead of focusing on the entire body. Our model will begin by extracting accurate body features and then utilize a Neuro-Fuzzy Classifier for classification, making it more rigid and flexible.

2.3 Feature Optimization and Classification

Li et al. [7] proposed a new method by the name VLAD (Vector of Locally Aggregated Descriptors) for Deep Dynamics, meaning VLAD3 was employed in order to incorporate multiple layers of video dynamics. For medium-term dynamics, VLAD3 integrates LDS (Laser Direct Structuring) for an interval of one and five times the length of a metric and deep CNN (Convolutional Neural Network) features for short-term dynamics. However, it is still constrained by its heavy reliance on CNNs as well as the LDS model, which is assumed to be linear in nature, meaning that time-based nonlinear relationships may not work well with this proposed model. Muhammad et al. [8] proposed the activity recognition model, which is an extension of the model that is based on a deep bi-directional long short-term memory (BiLSTM) and CNN. They employed ResNet152 to obtain deep features related to video frames, which were passed through the training DB-LSTM (Densely-connected Bi-directional LSTM). According to them, their method provided a better way of doing things as compared to the other available techniques. This has enabled this approach to overcome some of the challenges posed by other approaches while identifying recent progress in human activity recognition.

3 Proposed System Methodology

The challenges highlighted above are well suited to our proposed approach to analyzing RGB videos captured by drones. It includes converting the video into a series of frames and performing several preprocessing on each RGB frame of the video. In preprocessing, the concentration is made on reducing the computational complexity, scaling down image quality, and making foreground objects more conspicuous by eliminating noises in the background. For detecting humans within the frames, we utilize YOLO and achieve the extraction of human skeletal structures as well, identifying specific landmarks that correspond to significant body parts, which include the head, neck, both shoulders, elbows, wrists, hips, knees, feet, and the belly button. These landmarks, which encompass large joints that include the head, wrist, elbow, thighs, knees, and ankles, are essential for calculating normalized positions, joint angles, displacement and velocity, Histogram of Oriented Gradient (HOG), 3D coordinates, geodesic distances, and many more. These features are then maximized using Quadratic Discriminant Analysis (QDA) and later used in a Neuro-Fuzzy Classifier (NFC) for activity classification. Fig. 1 illustrates the proposed architecture.

3.1 Preprocessing

In our proposed system, we employ a data set consisting of drone footage for creating our model. The datasets include UAV-Gesture, Drone-Action, and Okutama which are in the video format, and therefore, the input to the system is a video. Since all the algorithms in our system deal with images, the first conversion is done on the video into frames. These extracted frames are passed through a bilateral filter to remove post-processing noise on the image. We have done it by using Eq. (1).

$$I'(p) = \frac{1}{k(p)} \sum_{q \in N(p)} \varnothing_s (\| \mathrm{p} - \mathrm{q} \|) . \varnothing_r (| \mathrm{I(p)} - \mathrm{I(q)} |) . \mathrm{I(q)} \quad (1)$$

where $I'(p)$ is the resulting filtered intensity at pixel p, $I(q)$ is the intensity of the pixel q in the neighborhood, $N(p)$ is the neighborhood set of pixel p, and $K(p)$ is the normalization factor to ensure all weights sum. It is important to note that after applying the bilateral filter, the images remain in the RGB color space. However, we proceed with the images further since our major concern is presenting the image description rather than color which at times may change the details contained in the image. For this purpose, blurred images are used as inputs, and a grayscale conversion technique is used to minimize noise. In the same manner, with the use of Eq. (2), we subtract or eliminate the background of the human subject, which is regarded as part of the noise [9].

$$Grayscale = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \tag{2}$$

where R is the intensity of the red channel, G is the intensity of the green channel, and B is the intensity of the blue channel. Fig. 2 provides the visual representation of preprocessing techniques employed.

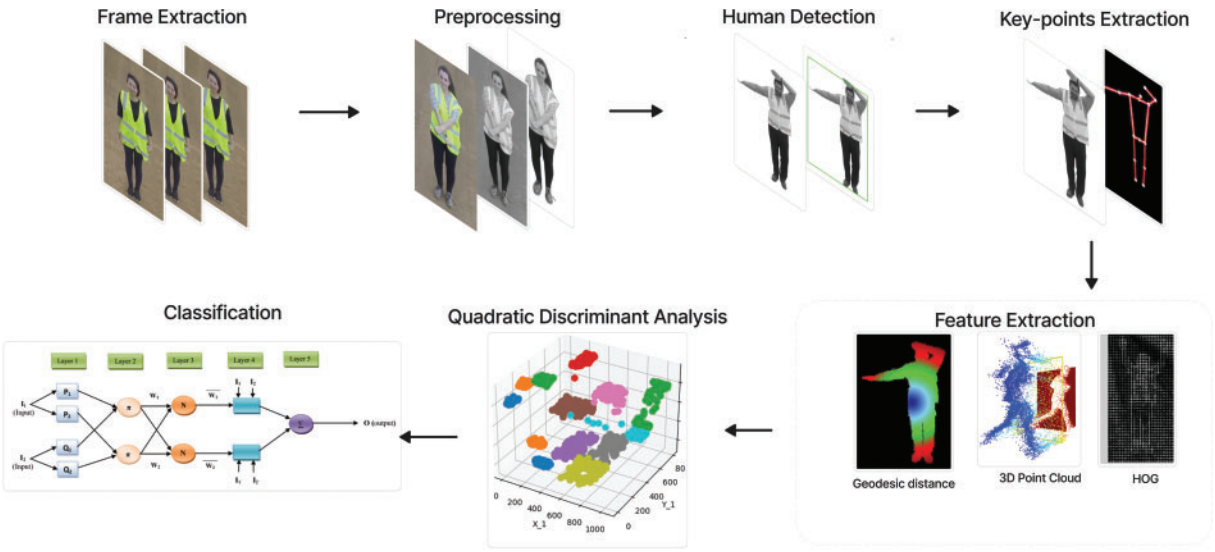


Figure 1: A comprehensive overview of the proposed system

Figure 2: Results from the preprocessing stage for (a) UAV-Gesture and (b) Drone action

3.2 Human Detection

Computer vision and machine learning overlap with the subject of studying and identifying objects or individuals in images, which opens up possibilities for various uses in robotics, unmanned automobiles, and drones. Object detection algorithms are broadly divided into two categories: single-shot detectors as well as two-stage detectors. Out of them, YOLO (You Only Look Once) has given quite a push towards the subject of object detection. Compared to other approaches, such as Faster R-CNN (Region-CNN), in which a region of the proposal network precedes a recognition step, YOLO utilizes a single fully connected layer for its predictions.

In the case of applying the YOLOv7 model, particularly for human detection, the first and foremost concern is to obtain better detection of bounding-box coordinates that are associated with high amounts of confidence levels in encountering human objects. Often it involves tweaking the training process and in some cases changing the structure of YOLO for more focus on the detection of humans. Our modifications are to add human interpretation to the outputs while maintaining the mathematical equations that define the algorithm. The focal point is in predicting the bounding boxes, with a preference for those most likely to hold a person. At the time of inference, all the produced bounding boxes that were not related to the human class were eliminated. We further improve detection precision by reducing the class prediction to the confidence score of the human class with the help of Eq. (3).

$$\text{YOLO Loss } = \lambda_{\text{coord}} \left[\sum_{i=0}^{B} Ii \left((C_i - C_i)^2 \right) + \lambda_{n0_{obj_i}} \sum_{i=0}^{B} I_{n0_{obj_i}} ((C_i - C_i)^2) \right] \tag{3}$$

where C_i are predicted confidence, $I_{n0_{obj_i}}$ are ground truth confidence and λ_{coord} are scaling factors. Results for human detection are illustrated in Fig. 3.

Figure 3: Human detection by applying YOLO (a) UAV-Gesture and (b) Drone action

3.3 Key-Point Extraction

In the case of the YOLO algorithm, the images are selected from videos, and the method is applied to them to identify humans in the identified images; if humans have been detected, specific anatomical landmarks are then selected for further examination. To localize the location of each body part, an Opencv pose estimator is used which is very significant to detect human skeletons in an image. This skeleton is used in identifying the angles and the distances between joints to obtain the correct measurements. Our system focuses on 15 key points: The areas mentioned include the head, neck, shoulders left and right, elbow, wrist, hips, knee, ankle, and navel. These points are critical in identifying the heights, and our system matters

at that. Nevertheless, there are several drawbacks: Opencv does not point to the neck, the belly button, or some other peculiarities exceeding the head. To overcome this limitation, we estimate these points by taking the midpoint.

Firstly, since some companies are in more than one country and the borders of some countries are not very clear, we assume these points are midpoints. For instance, the position of the neck is approximated as the mid-point of the left and right shoulder. This midpoint will involve averaging the x and y coordinates that correspond to the two important points. These midpoints can be calculated by using Eq. (4).

$$K_i = \left(\frac{\sum_{j=1}^{n} x_j}{n}, \frac{\sum_{j=1}^{n} y_j}{n} \right) \tag{4}$$

where k_i represents the i-th key point, x_j and y_j are the coordinates of the detected key points, and n is the total number of key points considered. Visual representation of extracted key points are presented in Fig. 4 and Fig. 5 provides a summary of identified landmarks belonging to various categories.

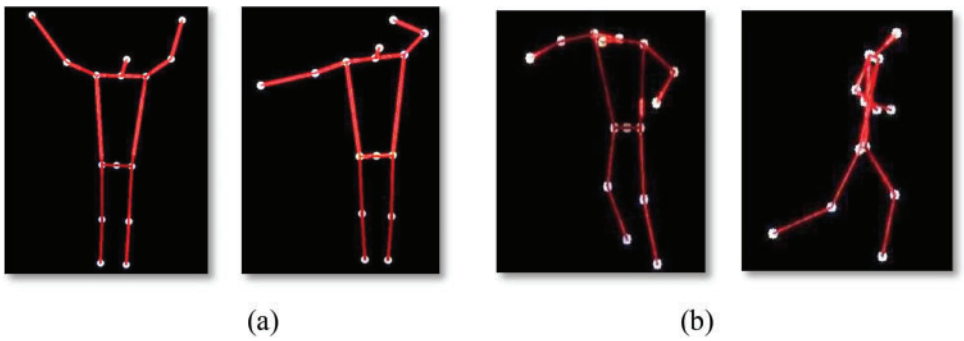


Figure 4: Extraction Key-points of the human body (a) UAV-Gesture and (b) Drone action

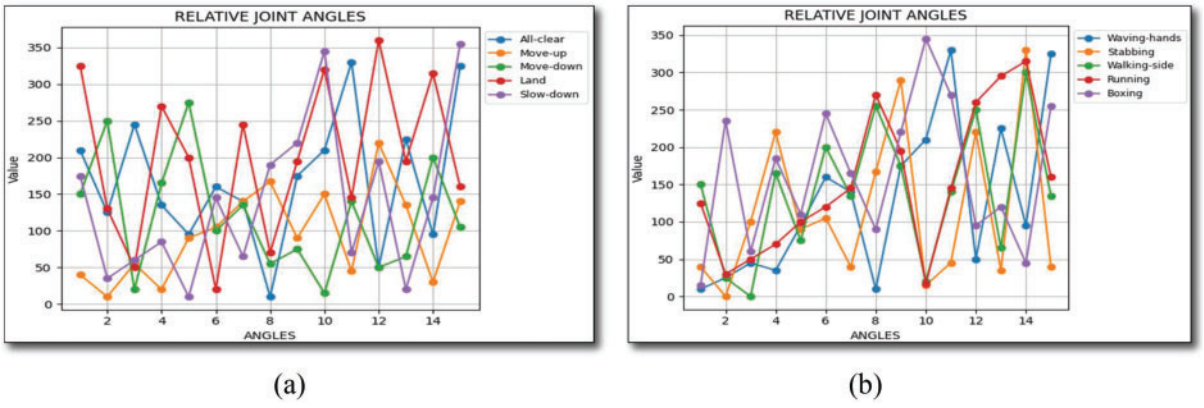


Figure 5: Relative joint angles (a) UAV-Gesture and (b) Drone action

3.4 Feature Extraction

In system development, special attention is paid to the selection of features that meet the intended goal or objective accurately. Feature selection serves a purpose the most crucial one because choosing the wrong

features directly affects the system's overall performance and the result that is being looked at. Furthermore, these features must be both independent and dependable at the same time. From the images, we extract various features and then gather their numerical values in one file for analysis.

3.4.1 Relative Joint Angles

The relative joint angles refer to the positions of the limbs in reference to each other during an activity of a specific limb. Observing these angles could help to improve the accuracy of activity recognition, as highlighted above. To calculate the angle between two points, the following Eq. (5) was used:

$$\theta = \cos^{-1}\left(\frac{x_1.x_2 + y_1.y_2}{\sqrt{x_1^2 + y_1^2}.\sqrt{x_2^2 + y_2^2}}\right) \tag{5}$$

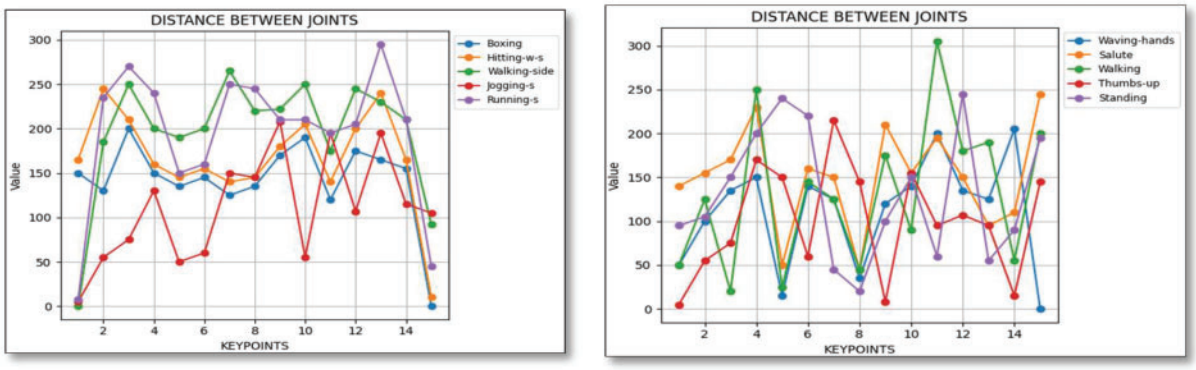
where (x_1, y_1) and (x_2, y_2) are the coordinates of the two points being under consideration.

3.4.2 Distance between Joints

During the subject performing an activity, the parts of the body in question are always in motion. The system proposed uses the speed of this transition and Distance each key point moves across frames from one frame to another as attributes that define the activity [9]. When calculating the estimations of Distance traveled, there are two frames, one preceding the current frame as well as the current frame. The Distance between the key point's previous and current positions is computed using Eq. (6):

$$\text{Distance} = \sqrt{\Delta x^2 + \Delta y^2} \tag{6}$$

where Δx is the difference between x-coordinates $(x_2 - x_1)$, and Δy is the difference between y-coordinates $(y_2 - y_1)$. Fig. 6 provides visual representation of relative distance.



(a) (b)

Figure 6: Relative distance (a) UAV-Gesture and (b) Drone action

3.4.3 Geodesic Distance

In this approach, human activity is represented by geodesic wave maps. These maps are developed by calculating the geodesic Distance—the shortest Distance found by the usage of the Fast-Marching Algorithm

(FMA). First, the center of the human figure is defined, and its Distance is set to 0 value. The distance initialization for the boundary points of the human silhouette is represented in Eq. (7):

$$d_{boundary}\,(p) = \begin{cases} 0, if\ p\ is\ the\ center\ point\ of\ the\ silhouette \\ d_0,\ if\ p\ is\ on\ the\ center\ of\ the\ silhouette \end{cases} \tag{7}$$

This point needs to be marked as visited before analyzing the second article. For the rest of the pixel points that are located in the boundary of the human silhouette, an initial distance value is set, and it is labeled that the pixel point has not been visited. The Distance is updated iteratively for each pixel based on the values of its neighboring points, computed by using Eq. (8):

$$d_{update}\,(p) = \min(d\,(p)\,,\ \min_{q\in \mathrm{N(p)}}\,(d(q) + \Delta \mathrm{d(q,p)})) \tag{8}$$

This continues where the Distance is for each neighboring pixel updated in each iteration of the procedure until all the pixels in an image are identified as visited. The distances computed in every iteration are then compared with the distances computed in the previous iteration. It is a priority-based approach with the priority assigned according to distances; points with small distances are favored. This is computed using the Eq. (9):

$$d\,(p) = \min(d\,(p)\,, d\,(s) + distance(s, p)) \tag{9}$$

where d(p) is the geodesic Distance from the starting point s to the point p, $d(s)$ is the geodesic Distance from the starting point s to itself, *distance*(s, p) is the Distance b/w current s and neighboring point p. Fig. 7 illustrates results for Geodesic distance.

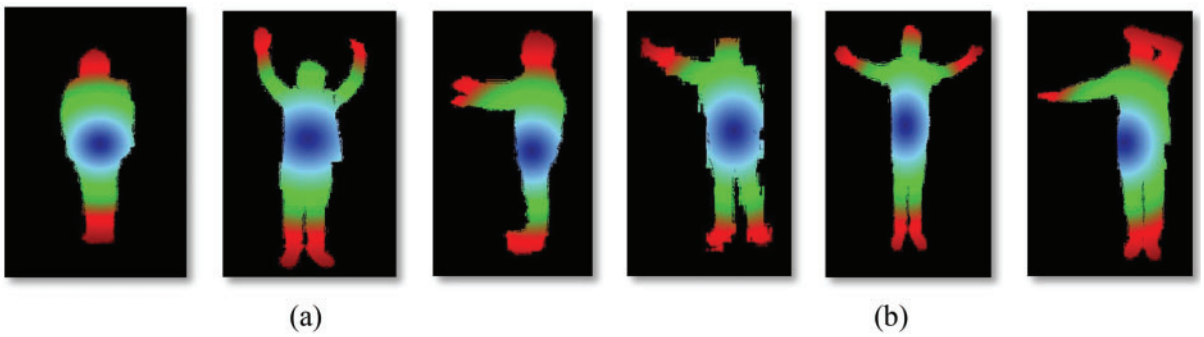


Figure 7: Geodesic distance for (a) UAV-Gesture and (b) Drone action

3.4.4 Histogram of Oriented Gradients (HOG)

It is often used in computer vision and image processing and mostly in the cases of object recognition and detection. It can be defined as a method that includes evaluating the position of gradients or edges in definite areas of an image. It begins by breaking this picture into small, interchangeable sections or what is termed cells. There is then calculated for each cell gradient information. Gradients are then separated into orientation bins and then form orientation histograms of these orientations. Such a process defines the foundations of the feature vector associated with each cell. For human body pose estimation, the representation of the histogram of oriented gradients (hog) can be calculated using Eq. (10):

$$H_k = \sum_{(i,j)\in cell} Magnitude\,(i,j)\,.w_k(\theta(i,j)) \tag{10}$$

where H_k is the histogram value of bin k, *Magnitude* (i, j) is the gradient magnitude at pixel (i, j). Fig. 8 illustrates HOF Features.

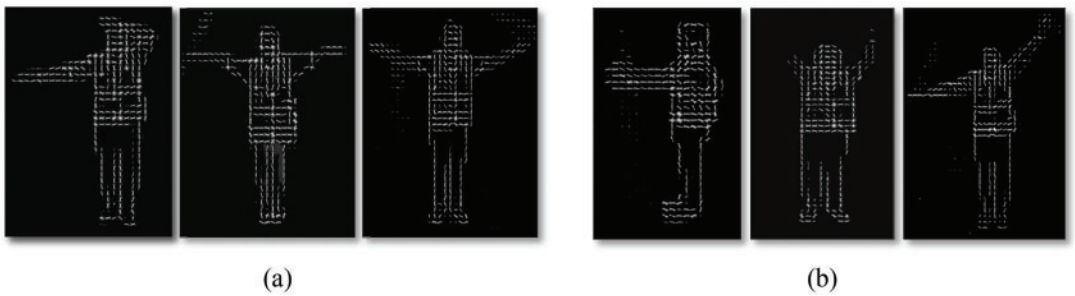


Figure 8: HOG features applied on (a) UAV-Gesture and (b) Drone action

3.4.5 3D Point Cloud

This feature utilizes all the pixel values that a person has in the image. First, we use the YOLO algorithm to detect the human in the image, then we find the central pixel of the bounding box generated by YOLO and iteratively cover all the pixels inside the bounding box. For further processing, we store the values of all Excel files containing human data. The central pixel YOLO bounding box can be calculated as in Eqs. (11) and (12):

$$x_{center} = x_{tI} + \frac{w}{2} \tag{11}$$

$$y_{center} = y_{tI} + \frac{h}{2} \tag{12}$$

where x, y_{center} horizontal and vertical coordinates of bounding box, x_{tI}, y_{tI}, x, y-coordinates of the top left corner of the box and $\frac{h}{2}$ is the half of height.

To find the z dimension of the pixel in the bounding box, we use Eq. (13):

$$Z_{mn} = \frac{d}{f} + \frac{m - C_x}{X} + \frac{d}{f} \times \frac{n - C_y}{Y} \tag{13}$$

Z_{mn} represents depth of pixel (m,n), f is focal length, X, Y are scaling factors and (m,n) are pixel coordinates. Fig. 9 shows results of 3D-point cloud.

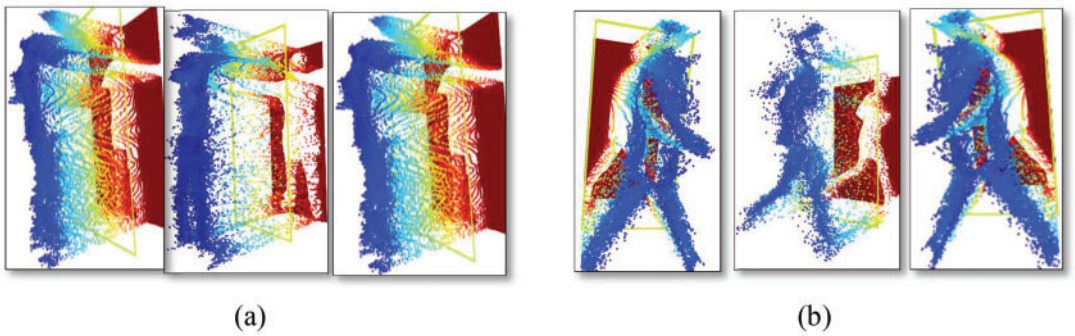


Figure 9: Results of 3D-point-cloud for (a) UAV-Gesture and (b) Drone action

3.5 Quadratic Discriminant Analysis

Quadratic Discriminant Analysis also known as QDA is the method that is utilized in both feature enhancement and prediction. This is done by using a quadratic decision boundary to classify items under different classes. The procedure starts from the assumption that data points within each class are normally distributed having mean vector and covariance matrix pertaining to each class. QDA, on the other hand, assesses the probability of a feature vector given its respective classes by the use of normal density function. The procedure of classification entails the identification of the class with the highest probability of occurrence and is deemed as the predicted category. QDA also applies regularization to make feature optimization better; this is done by adding a small positive value to the diagonal elements of the covariance matrix. This adjustment helps reduce over fitting and helps create a better model as the earlier vertical cavity surface emitting laser (VCSEL) model is much more complex. Mathematically, by using Eq. (14):

$$p\left(C_k|x\right) = \frac{p(C_k)p(C_l)}{\sum_{j=1}^{K} p(x|C_j)p(C_j)} \tag{14}$$

where $p(C_k)$ is the probability of class C_k, and K is the number of classes. Fig. 10 is a visual plot for optimized feature allocation through Quadratic Discriminant Analysis.

3.6 Neuro-Fuzzy Classifier

A Neuro-Fuzzy Classifier is a tool that is the result of merging characteristics of neural networks and fuzzy logic to address classification problems that are characterized by uncertainty. Neural networks are phenomenal in modeling complex patterns using weight, Bias, and iterative techniques of training. On the other hand, fuzzy logic is a progressive improvement of Boolean logic since it avails the concepts of degrees of truth, which is important when dealing with disorders in data. In a Neuro-Fuzzy Classifier, these two approaches work together: They provide better and more consistent classification results as compared to purely deterministic or statistical classification approaches. We used Eq. (15):

$$\hat{y} = \sum_{j=1}^{m} w_j.\mu_j(\mathrm{X}) \tag{15}$$

where y is the predicted output, X is the input given to the classifier w_j are the weight belongs to fuzzy sets. Fig. 10 The Architecture of Neuro-Fuzzy Classifier. Algorithm 1 shows the working of the Neuro-Fuzzy Classifier function (See Figs. 10 and 11).

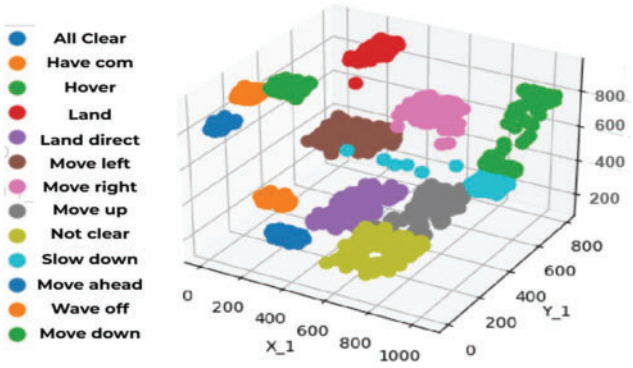


Figure 10: Refined feature allocation through Quadratic Discriminant Analysis on UAV-Gesture dataset

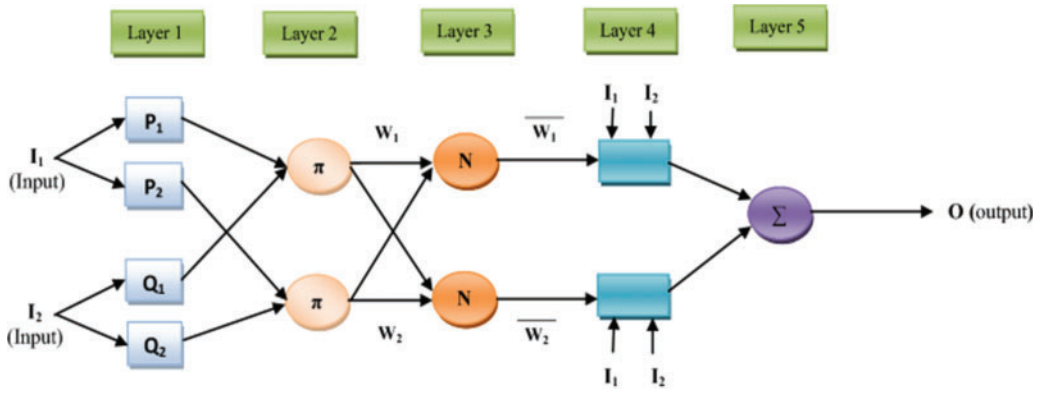


Figure 11: Architecture and operational flow of the Neuro-Fuzzy Classifier

Algorithm 1: Neuro-Fuzzy Classifier function

FUNCTION NeuroFuzzyClassifier(input X):
Step 1: Initialize neural network parameters
INITIATE weight vector W = {w_1, w_2, ..., wm} # Weights for each fuzzy rule
SET bias B = random value # Randomly initialize bias term
Step 2: Fuzzification-Calculate fuzzy membership grades for input X
For each fuzzy rule, j from 1 to m **DO**:
μ_j(X) = MembershipFunction(j, X) # Compute fuzzy membership grade for rule j using fuzzy set
END FOR
Step 3: Apply Neuro-Fuzzy inference
y_hat = 0 # Initialize the output prediction
For each fuzzy rule, j from 1 to m DO:
Multiply the weight of the fuzzy rule by its membership grade and sum the result
y_hat += W[j] * μ_j(X)
END FOR
RETURN some activation of 'value', e.g., sigmoid, tanh, etc.
END FUNCTION

4 Experimental Setup and Datasets

4.1 Experimental Setup

For conducting the experiments described in this study, a laptop with an Intel Core i5 processor and 8 GB of RAM was used. The operating system used here was a 64-bit Windows 10, while PyCharm was used as the development environment for writing code.

4.2 Datasets Description

4.2.1 UAV-Gesture Dataset

The UAV-Gesture dataset includes 119 high-definition RGB videos that were shot in outdoor environments. That makes for a total of 37,151 RGB frames in these videos. They were taken with a drone camera while participants performed some gestures to maneuver UAVs or helicopters. The dataset features 13 distinct

actions: hover, left, up, forward, landing, down, slowed, right, waving, away, order, ambiguous, and, direction of landing [10].

4.2.2 Drone-Action Dataset

Drone-Action is a newly designed dataset for detecting human action through drone video. It comprises 66,919 frames of images and 240 segments of video clips. The dataset comprises different activity types which include jogging while being followed, stabbing, stick hitting, clapping, side walking, punching, kicking, side jogging, bottle hitting, hand waving, side running, following while walking, and following while running [11].

4.2.3 Okutama Dataset

The Okutama-Action dataset consists of 43 videos which are approximately 1775 frames in total. They were taken with two unmanned aerial vehicles (UAVs) with the distance varying from 10 to 45 m from the sample site. These have been made with the camera both at 45-degrees and 90-degrees with the subject. For the Human Activity Recognition (HAR) task [12].

5 Experimental Results

We evaluated the proposed Human Activity Recognition (HAR) system using three datasets: these include UAV-Gesture, Drone-action, and Okutama-Action. The above Table 1 shows the actual confusion matrix of the system working on the UAV-Gesture dataset in which we got a mean accuracy of 97%. As listed in Table 2 below, concerning the Drone-Action dataset, the system achieved a mean accuracy of 93%. Testing the system on Okutama-Action dataset, the prediction accuracy achieved on average was 81% and the results are presented in below Tables 1–3.

Table 1: Confusion matrix of our system on UAV-Gesture Dataset

Cls	Ac	Hc	H0	La	Ld	Ma	Md	Mi	Mr	Mu	Nc	Sd	W0
Ac	97	0	0	1	0	1	1	0	0	0	0	0	0
Hc	0	96	0	0	1	0	1	0	0	0	1	0	1
Ho	0	0	98	0	0	0	0	1	0	0	0	0	0
La	0	0	0	97	0	0	0	0	0	1	0	1	1
Ld	0	0	0	0	97	0	0	0	1	0	1	0	1
Ma	0	1	0	0	1	95	0	1	0	1	0	1	0
Md	0	0	0	0	0	0	98	0	1		0	0	1
Mi	0	0	0	0	0	0	0	98	1	0	0	1	0
Mr	0	0	0	1	0	1	0	0	97	0	0	0	1
Mu	0	1	0	0	1	0	0	0	1	96	1	0	0
Nc	0	0	1	0	0	0	0	0	0	1	97	0	1
Sd	0	0	0	0	0	0	1	0	1	0	0	98	0
Wo	0	0	0	0	0	1	0	1	0	1	0	0	97

Note: Ac = All-clear, Hc = Have-command, H0 = Hover, La = Land, Ld = Landing-direction, Ma = Move-ahead, Md = Move-downward, Mi = Move-left, Mr = Move-right, Mu = Move-upward, Nc = Not-clear, Sd = Slow-down, Wo = Wave-off.

Table 2: Confusion matrix of our system on Drone-Action Dataset

Cls	Bx	Cl	Hb	Hs	Jo-f	Jo-s	Kc	Rb	Rs	Sb	Wf	Ws	Wv
Bx	93	0	0	1	0	1	1	0	1	1	1	0	1
Cl	0	94	1	0	1	1	1	0	0	0	1	1	0
Hb	0	1	92	1	1	0	1	0	1	1	0	1	1
Hs	0	0	1	93	1	1	1	0	0	1	0	1	1
Jof	1	0	1	0	93	0	1	1	1	0	1	0	1
Jos	0	1	1	0	1	95	0	1	0	0	0	1	0
Kc	0	0	0	1	1	1	92	1	1	1	0	1	1
Rb	1	1	0	1	0	1	0	93	1	0	1	1	0
Rs	1	0	1	1	0	0	1	1	92	1	1	0	1
Sb	0	1	0	0	1	1	1	0	1	93	1	1	0
Wf	1	0	1	1	0	1	0	1	0	1	93	0	1
Ws	0	0	0	0	1	0	1	0	1	1	1	94	1
Wv	1	1	1	0	1	1	1	1	0	1	0	0	92

Note: Bx = Boxing, Cl = Clapping, Hb = Hitting with bottle, Hs = Hitting with stick, Jof = Jogging front back, Jos jogging side, Kc = Kicking, Rb = Running front back, Rs = Running side, Sb = stabbing, Wf = walking front back, Ws = walking side, Wv = Waving Hands.

Table 3: Confusion matrix of our system on Okutama Dataset

Class	Ru	Wl	Ly	S	St
Ru	81	4	9	4	2
Wl	7	80	3	5	5
Ly	2	6	82	2	8
S	7	5	3	81	4
St	1	7	3	8	81

Note: Ru = Running, Wl = Walking, Ly = Laying, S = Sitting, St = Standing.

5.1 Precision, Recall, and Accuracy for Locomotion Activities

We use precision, recall, and mean accuracy as evaluation parameters in our research. Tables 4–6 show the precision, recall, and class accuracy of UAV-Gesture, Drone-Action, and Okutama Dataset respectively. Table 7 shows the comparison of our system with existing models. We calculate these by using Eqs. (16)–(18).

$$Precision = \frac{True\ Positives}{True\ Positvies + False\ Positives} \tag{16}$$

Table 4: Classification report on the UAV-Gesture dataset

Classes	Accuracy	Precision	Recall
Ac	0.97	0.96	0.95
Hc	0.96	0.95	0.94
Ho	0.98	0.97	0.96
La	0.97	0.96	0.95
Ld	0.97	0.96	0.95
Ma	0.95	0.94	0.93
Md	0.98	0.97	0.96
Mi	0.98	0.97	0.96
Mr	0.97	0.96	0.95
Mu	0.96	0.95	0.94
Nc	0.97	0.96	0.95
Sd	0.98	0.97	0.96
Wo	0.97	0.96	0.95
Average	**0.97**	**0.96**	**0.95**

Table 5: Classification report on the Drone-Action dataset

Classes	Accuracy	Precision	Recall
Bx	0.93	0.92	0.91
Cl	0.94	0.93	0.92
Hb	0.92	0.91	0.90
Hs	0.93	0.92	0.91
Jof	0.93	0.92	0.91
Jos	0.95	0.94	0.93
Kc	0.92	0.92	0.91
Rb	0.93	0.92	0.91
Rs	0.92	0.91	0.90
Sb	0.93	0.92	0.91
Wf	0.92	0.91	0.90
Ws	0.94	0.93	0.92
Wv	0.92	0.91	0.90
Average	**0.93**	**0.92**	**0.91**

Table 6: Classification report on the Okutama dataset

Classes	Accuracy	Precision	Recall
Running	0.81	0.80	0.79
Walking	0.82	0.81	0.80
Laying	0.80	0.79	0.78
Sitting	0.81	0.80	0.79
Standing	0.81	0.80	0.79
Average	**0.81**	**0.80**	**0.79**

Table 7: Comparisons of the proposed system with other systems

Methods	UAV Gesture	Okutama	Drone action	Percentage improvement
CNN [3]	–	0.78	0.80	3.85% (Okutama), 16.25% (Drone Action)
Multi-feature + CNN [5]	0.95	–	0.90	2.11% (UAV Gesture), 3.33% (Drone Action)
MLP_7j [11]	0.94	–		3.19% (UAV Gesture)
P-CNN [12]	–	–	0.75	24.00% (Drone Action)
3D CNN + BVC + Capsule [13]	–	47.50	–	70.53% (Okutama)
Weighted Temporal Fusion + Inceptionv3 + Pose-Stream [14]	–	72.76	–	11.25% (Okutama)
P-CNN [15]	0.91	–	–	6.59% (UAV Gesture)
SWTF + Pose-Stream [16]	–	–	0.78	19.23% (Drone Action)
3D CNN + Capsule [17]	–	41.87	–	93.54% (Okutama)
Proposed system mean accuracy	**0.97**	**0.81**	**0.93**	

It is the ratio of total identified positive cases to all actual positive cases.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{17}$$

Recall indicates the proposition of correctly classified cases (both positives and negatives) out of the total cases.

$$Accuracy = \frac{True\ Positve + True\ Negative}{Total\ Population} \tag{18}$$

Table 7 shows the comparison of our system with existing models and also shows the relative improvement of our proposed system over each existing method as a percentage, calculated as follows:

$$Percentage\ Improvement = \frac{\text{Proposed System Accuracy} - \text{Existing Method Accuracy}}{Existing\ Method\ Accuracy} \times 100$$

6 Conclusion

The proposed technique offers a new approach to establishing human activity in a drone video, making its detection more manageable. Thus, one of the priorities of the system is to recognize human poses and sort them correctly by selecting certain features. This makes it possible for the users to study and interpret several human activities satisfactorily. The incorporation of a Neuro-Fuzzy Classifier (NFC) improves the system's performance besides analyzing regional changes that are helpful in the differentiation of differences in human motion. This capability is particularly important for applications in which it is crucial to spot accurate action and gesture recognition since it offers a better understanding of the actions within environmental surroundings. Thirdly, some image preprocessing procedures that are integrated into the presented method are of great importance to enhance image quality and exclude interfering backgrounds. One of the biggest advantages of our system is that it effectively minimizes the miss-prediction of data samples placed in

two similar classes through the use of Quadratic Discriminant Analysis (QDA). In addition, the system operates with higher accuracy due to the solution of the problems arising during the photo preprocessing step associated with dynamic backgrounds. In the future, further work plans to increase the flexibility of the system by introducing new functionalities and validating it using a larger set of benchmark datasets. By including different operation scenarios and actions in the training of the model, the allowable operation variations will be expanded. This was in line with our continuous improvement and optimization of the subsequent detection of human actions in drone videos to improve our performance when deployed in the actual world.

Acknowledgement: The authors are thankful to Princess Nourah bint Abdulrahman University Researchers Supporting Project, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: The APC was funded by the Open Access Initiative of the University of Bremen and the DFG via SuUB Bremen. Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R348), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: Study conception and design: Yawar Abbas; data collection: Aisha Ahmed Alarfaj and Ebtisam Abdullah Alabdulqader; analysis and interpretation of results: Ahmad Jalal; draft manuscript preparation: Yawar Abbas, Asaad Algarni and Hui Liu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All publicly available datasets are used in the study.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Hwang H, Jang C, Park G, Cho J, Kim IJ. ElderSim: a synthetic data generation platform for human action recognition in eldercare applications. IEEE Access. 2021;11:9279–94. doi:10.1109/ACCESS.2021.3051842.
2. Arunnehru J, Thalapathiraj S, Dhanasekar R, Vijayaraja L, Kannadasan R, Khan AA, et al. Machine vision-based human action recognition using spatio-temporal motion features (STMF) with difference intensity distance group pattern (DIDGP). Electronics. 2022;11(15):2363. doi:10.3390/electronics11152363.
3. Azmat U, Alotaibi SS, Al Mudawi N, Alabduallah BI, Alonazi M, Jalal A, et al. An elliptical modeling supported system for human action deep recognition over aerial surveillance. IEEE Access. 2023;11:75671–85. doi:10.1109/ACCESS.2023.3266774.
4. Zhu W, Lan C, Xing J, Zeng W, Li Y, Shen L, et al. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. 2016. doi:10.48550/arXiv.1603.07772.
5. Azmat U, Alotaibi SS, Abdelhaq M, Alsufyani N, Shorfuzzaman M, Jalal A, et al. Aerial insights: deep learning-based human action recognition in drone imagery. IEEE Access. 2023;11:83946–61. doi:10.1109/ACCESS.2023.3302353.
6. Shi Y, Tian Y, Wang Y, Huang T. Sequential deep trajectory descriptor for action recognition with three-stream CNN. IEEE Trans Multimed. 2017;19(7):1510–20. doi:10.1109/TMM.2017.2666540.
7. Li Y, Li W, Mahadevan V, Vasconcelos N. VLAD3: encoding dynamics of deep features for action recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun; Las Vegas, NV, USA. p. 1951–60.
8. Muhammad K, Mustaqeem, Ullah A, Imran AS, Sajjad M, Kiran MS, et al. Human action recognition using attention based LSTM network with dilated CNN features. Future Gener Comput Syst. 2021;125(3):820–30. doi:10.1016/j.future.2021.06.045.
9. Abbas Y, Jalal A. Drone-based human action recognition for surveillance: a multi-feature approach. In: 2024 International Conference on Engineering & Computing Technologies (ICECT); 2024 May 23; Islamabad, Pakistan: IEEE. p. 1–6. doi:10.1109/ICECT61618.2024.10581378.

10. Perera AG, Law YW, Chahl J. UAV-GESTURE: a dataset for UAV control and gesture recognition. In: Computer Vision–ECCV 2018 Workshops; 2019; Cham: Springer International Publishing. p. 117–28. doi:10.1007/978-3-030-11012-3_9.
11. Perera AG, Law YW, Chahl J. Drone-action: an outdoor recorded drone video dataset for action recognition. Drones. 2019;3(4):82. doi:10.3390/drones3040082.
12. Barekatain M, Martí M, Shih HF, Murray S, Nakayama K, Matsuo Y, et al. Okutama-action: an aerial view video dataset for concurrent human action detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); 2017 Jul 21–26; Honolulu, HI, USA: IEEE. p. 2153–60. doi:10.1109/CVPRW.2017.267.
13. Algamdi AM, Sanchez V, Li CT. Dronecaps: recognition of human actions in drone videos using capsule networks with binary volume comparisons. In: 2020 IEEE International Conference on Image Processing (ICIP); 2020 Oct 25–28; Abu Dhabi, United Arab Emirates: IEEE. doi:10.1109/icip40778.2020.9190864.
14. Yadav SK, Pahwa E, Luthra A, Tiwari K, Pandey HM, Corcoran P, et al. SWTF: sparse weighted temporal fusion for drone-based activity recognition. 2022. doi:10.48550/arXiv.2211.05531.
15. Chéron G, Laptev I, Schmid C. P-CNN: pose-based CNN features for action recognition. In: 2015 IEEE International Conference on Computer Vision (ICCV); 2015 Dec 7–13; Santiago, Chile: IEEE. p. 3218–26. doi:10.1109/ICCV.2015.368
16. Yadav SK, Luthra A, Pahwa E, Tiwari K, Rathore H, Pandey HM, et al. DroneAttention: sparse weighted temporal attention for drone-camera based activity recognition. Neural Netw. 2023;159(1):57–69. doi:10.1016/j.neunet.2022.12.005.
17. Zhang P, Wei P, Han S. CapsNets algorithm. J Phys: Conf Ser. 2020;1544(1):012030. doi:10.1088/1742-6596/1544/1/012030.