JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

Human Action Recognition in Military Obstacle Crossing Using HOG and Region-Based Descriptors

Adeola O. Kolawole *, Martins E. Irhebhude, and Philip O. Odion

Department of Computer Science, Faculty of Military Science and Interdisciplinary Studies, Nigerian Defence Academy, Kaduna Nigeria; e-mail : adeolakolawole@nda.edu.ng; mirhebhude@nda.edu.ng; podion2012@gmail.com

* Corresponding Author: Adeola O. Kolawole

Abstract: Human action recognition involves recognizing and classifying actions performed by humans. It has many applications, including sports, healthcare, and surveillance. Challenges such as a limited number of classes of activities and variations within inter and intra-class groups lead to high misclassification rates in some of the intelligent systems developed. Existing studies focused mainly on using public datasets with little focus on real-life action datasets, with limited research on HAR for military obstacle-crossing activities. This paper focuses on recognizing human actions in an obstacle-crossing competition video sequence where multiple participants are performing different obstacle-crossing activities. This study proposes a feature descriptor approach that combines a Histogram of Oriented Gradient and Region Descriptors (HOGReG) for human action recognition in a military obstacle crossing competition. The dataset was captured during military trainees' obstacle-crossing exercises at a military training institution to achieve this objective. Images were segmented into background and foreground using a Grabcut-based segmentation algorithm, and thereafter, features were extracted and used for classification. The features were extracted using a Histogram of Oriented Gradient (HOG) and region descriptors from segmented images. The extracted features are presented to a neural network classifier for classification and evaluation. The experimental results recorded 63.8%, 82.6%, and 86.4% recognition accuracies using the region descriptors HOG and HOGReG, respectively. The region descriptor gave a training time of 5.6048 seconds, while HOG and HOGReG reported 32.233 and 31.975 seconds, respectively. The outcome shows how effectively the suggested model performed.

Keywords: Histogram of oriented gradient; HOGReG; Human action recognition; Neural Network Classifier; Obstacle crossing competition; Region descriptor.

Received: January, 20th 2025
Revised: February, 13th 2025
Accepted: February, 14th 2025
Published: February, 21st 2025

Copyright: © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) licenses (https://creativecommons.org/licenses/by/4.0/)

1. Introduction

Human Action Recognition (HAR) is one of the most attractive study areas in computer vision applications. For machines to understand human activity, automatic methods for analyzing and interpreting actions in videos and images are important[1]. The ability of computers to comprehend images and videos on their own is one of the trends in the computer vision industry. The core of computer vision is image processing[2]. Vision-based action recognition and prediction from video image tasks involve activity recognition relating to human actions, i.e., the present state, and action prediction, which predicts human action, i.e., the future state. These two tasks are used in real-world applications such as surveillance, human-computer interaction, autonomous vehicles, healthcare, and video retrieval. With the human vision system, the actions and purposes of an actor can be easily understood and known when an individual is performing a particular task, and the actor can confidently guess whether or not such actions comply with the instructions given [3].

Human actions are considered to be a predetermined set of physical movements a person performs over time to complete a given task. In some scenarios, an object or interaction with other people is required to complete a task, and as a result, the level of complexity of the action varies. Some complex actions can occur over a longer period, which requires a

larger number of consecutive frames (video sequence)[4]. The purpose of HAR systems is to correctly identify human actions by analyzing scenarios in the real world[5]. HAR enables computer-based applications to assist users in executing tasks and improving their lifestyle, such as posture monitoring when exercising and remote care for older adults living alone[6]. HAR technologies can be divided into vision-based and sensor-based[7]. Vision-based HAR research is divided based on the data type, such as RGB and RGB-depth. Both data are recorded with cameras, which are highly affordable and produce rich texture data.

HAR research has been applied in a wide range of activities in different environments, such as action recognition and localization in realistic sports videos[8], activity recognition in mountain climbing with variations in terrain[9], tracking of illegal activities, and the surveillance of restricted military environment[10]. Handcrafted and deep-learned features have been utilized by researchers for the classification of human actions, with few considerations on the fusion of features[11]. However, few works have focused on recognizing human actions in military training activities. Most studies have focused on using public datasets with little or no focus on real-life military exercises. According to Gahtan et al. [12], analyzing a soldier's activity is important. This analysis is important in understanding physiological data, which is quite challenging as information on such activities is not always readily available or clear. In obstacle crossing, assessment is done based on precision i.e., Judgment of any obstacle is based on specific action and how the action was carried out.

This paper explored the recognition of different human actions from visual images in an obstacle-crossing exercise in a military training institution. It proposed a feature-based approach that combined HOG and region attributes for classification using a neural network classifier. The study independently evaluated HOG and region descriptors and their combination (HOGReG). This study is relevant in the performance analysis of trainees during military exercises or activity recognition in remote and dangerous environments. The study is also important in assessing military trainee exercises and competitions and can be used as instant feedback and assessment of a trainee's performance, which is useful for instructors.

The paper is structured as follows: section 2 covers the literature review and other concepts. Section 3 discusses the proposed methodology involving the dataset used, HOG feature extraction, and classification. The results of the implemented work are discussed and analyzed in Section 4. Finally, the conclusion and future work is discussed in section 6.

2. Literature Review

Human activity recognition recognizes human actions in a video sequence [13], [14]. The main objective of the HAR system is to successfully understand ongoing events and observe and analyze human activities by retrieving and processing data to classify human behavior into different actions such as jumping, climbing, and crawling. Identifying the unknown human actions in videos is achieved by analyzing frames from such videos, which serve as data that can be classified using models whose efficiency is tested in terms of accuracy, speed, and simplicity[15]. Research on action recognition has used various data modalities like RGB images, videos, and skeletal images, using several techniques that have achieved impressive results[16]. However, However, some of its limitations include lightning, occlusion, and cluttered backgrounds. To overcome these challenges, most authors considered local spatial features, global features, and temporal features for HAR[17].

Alex et al. [18] examined the effectiveness of four selected supervised machine learning algorithms: Naive Bayes, RF, Multi-Layer Perceptron (MLP), and Support Vector Machine (SVM) on their ability to recognize human activities. Five categories of actions were considered namely; category I (asleep or not), category II (eating), category III (walking), category IV (falling), and category 5 (talking on the phone). Feature extraction used HOG, Local Binary Pattern, and Bag Of features. SVM achieved an average accuracy of 86%, while RF MLP and Naive Bayes achieved 70%, 72%, and 71.8%, respectively.

Kumar and Bhavani [19] proposed human activity recognition based on a filtering technique and watershed segmentation algorithm using HOG, colour, GiST, and a combination of all features for feature extraction. The experimental dataset contained 20 activities that were recorded in four different scenarios. The activities include riding on the elevator down, riding on the elevator up, riding on the escalator down, riding on the escalator up, walking, sitting, walking downstairs, walking upstairs, drinking, eating, making phone calls, texting, cycling, doing push up, doing sit up, running, organizing files, reading, working on pc, and

writing sentences. Experimental results obtained using SVM gave 69.23% with HOG, GiST, colour, and combination of all features reported 77.43%, 64.88%, and 79.5%, respectively. A further experiment was conducted using RF classifier which yielded 75.21% on HOG, 80.52% with GiST, colour, and a combination of all features gave 66.88% and 82.45%, respectively.

Patel et al. [13] proposed a feature descriptor model for HAR to explore the dissimilarities in actions, HOG, Regional Features from Fourier HOG (R_FHOG), displacement in object position (OBJ_DISP), and velocity of an object (OBJ_VELO) features were used for extracting feature descriptors. The authors explored a fusion of all these features to improve class classification and diversity. Artificial Neural Network (ANN), SVM, multiple kernel learning (MKL), late fusion approach, and Meta-cognitive Neural Network (McNN) were classifiers used to classify human actions on public datasets; KTH [20] and Weizmann. Using KTH datasets. The KTH dataset contained six types of human actions (walking, jogging, running, boxing, handwaving, and hand clapping). In contrast, the Weizmann contained ten different human actions (run, walk, skip, jumping-jack, jump forward on two legs, jump in place on two legs, galloping sideways, wave two hands, wave one hand, and bend). The experiment achieved 84.12% and 92.32% results for early fusion with ANN and SVM respectively, 93.03% and 95.85% for MKL with ANN and MKL with SVM respectively, and 100% for McNN. The Weizmann dataset gave 91.943% and 94.34% for early fusion with ANN and early fusion with SVM, 92.09% and 93.89% on MKL with ANN and MKL with SVM respectively, while MCN also gave 100%. The detection of moving objects from constrained videos was also considered by the model, which can still be applied to videos where multiple actions are performed. However, for further improvement, overlapping feature extraction can be implemented.

Khan et al.[11] designed a framework combining conventional handcrafted and deep features for human action recognition from video frames. The shape features were extracted with HOG, while the pre-trained Caffe AlexNet model was used for deep feature extraction. Simulations were performed on five publicly available benchmark datasets, achieving recognition rates between 99.6% and 100% for each dataset.

Tan et al. [21] proposed a combination of deep neural networks and handcrafted methods to recognize human actions in video. Sparse autoencoder was employed to train the filter from images, while HOG was utilized to extract texture and shape features from the filtered images. For classification, the Modified Hausdorff Distance was applied. The experiment used two publicly available datasets and one self-gathered dataset; Weizmann, CAD-60 and MMU human action dataset, yielding recognition rates of 100%, 88.24%, and 99.5%, respectively. The MMU consists of 10 classes of human action: walking, answering the phone, surrender, pointing, squads, throwing, running, writing, kicking, and punching.

Singh [22] classified human actions into five categories: boxing, jogging, and handclapping, which were obtained from the KTH database while jumping and bending were obtained from the Weizmann database. The human silhouette was obtained from videos using the optical flow technique algorithm, and then features were extracted using HOG features and SVM as a classifier. The overall accuracy was 86.65% on the KTH database and 85.60% on the Weizmann dataset.

Elharrouss [23] proposed a CNN model for action-based video summarization by recognizing multiple human actions from different scenes. Two techniques were employed: cosine similarity measure of the HOGs of the Temporal Difference Map (CS-HOG-TDMap) and CNN classification of actions from the TDMap images (CNN-TDMap). Experiments were conducted using the following datasets: the Weizmann, KTH, UCF-ARG multi-view action dataset, UT-Interaction (which includes six classes: shake-hands, point, hug, push, kick and punch.), IXMAS (contains 10 actions, including boxing, walking, running, hand waving, hand clapping, jogging, carrying, standing, backpack carrying, and two persons fighting). On the KTH, Weizmann, and UCF-ARG datasets, the CS-HOG-TDMap achieved a recognition rate of 98%, while CNN-TD-Map recognized 99% of the actions. The UT-Interaction dataset gave 87% and 98%, respectively, while the IXMAS achieved 99%.

Sahoo et al. [24] used a Bag of histogram of optical flow (BoHOF) to differentiate actions varying with speed of action using features calculated from segmented human objects. HOG features were also extracted and combined with BoHOF features. KTH datasets were used with SVM as a classifier to assess the effectiveness of the proposed fusion, yielding an overall accuracy of 96.7%.

Gundu and Syed [25] employed a hybrid model of HOG, mask-regional convolutional neural network (Mask-RCNN), and bidirectional long short-term memory (Bi-LSTM) for recognizing human action using YouTube aerial data containing Band Marching, Biking, Cliff Diving, Golf Swing, Horse-riding, Kayaking, Skateboarding, and activities. The model achieved 99.25% accuracy.

Javed et al.[26] used a smartphone accelerometer sensor to recognize six daily activities: standing, sitting, downstairs, walking, upstairs, and jogging. Three machine learning models (decision tree, logistic regression, and multi-layer perceptron) were implemented to train the model with MLP, which had the highest overall accuracy of 93%. Although these studies have used different approaches and technologies for activity recognition, no known work has focused on the wide range of activities, especially in the military obstacle-crossing competition, which has not been recognized.

Ha [27] proposed a human video activity recognition architecture using a 3D Convolutional Neural Network for extracting spatial and temporal features from video actions. The performance of the proposed model was evaluated using publicly available datasets: UCF101, HMDB51, and Traffic Police (TP) dataset to achieve average accuracies of 98.3%, 80.7%, and 97.6%, respectively.

2.1. Histogram of Oriented Gradient (HOG)

Histogram of oriented gradient (HOG) is a feature descriptor proposed by [28]. It is used in computer vision for object detection. The HOG is a highly effective feature for human detection [13]. Feature is important in recognition as it represents some hidden information that helps to facilitate learning for easier human interpretations[22]. HOG can describe local objects' appearance and shape well, which is the distribution of local intensity gradients or edge directions, even without knowledge of the corresponding edge or gradient position. It focuses on the structure or shape of an object while counting the occurrence of gradient orientation in each local region. For the region images, it uses the magnitude and orientations to create histograms[29]. HOG implementation is done by dividing an image window into small regions called cells, with each cell accumulating a 1-D histogram of gradient directions or edge orientation over pixels of that cell, see Figure 1[30]. A combination of the histogram entries forms the representation. To ensure invariance to illumination and shadowing, contrast normalization is done on all cells in the block to form a HOG descriptor.

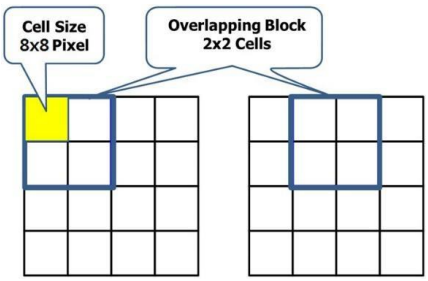

Figure 1. Cells and Overlapping Blocks in HOG[30]

Dalal and Triggs evaluated four block normalization schemes [28]. Let v be the unnormalized descriptor containing histograms in a given block, $\| v \|_k$ be it k- norm form for $k = 1,2$ and e be a small constant. The normalization factor can be any of the schemes shown in Equations (1) to (3).

$$L1 - norm: f = \frac{v}{\| v \|_1 + e} \tag{1}$$

$$L1 - sqrt: f = \sqrt{\frac{v}{\| v \|_1 + e}} \tag{2}$$

$$L2 - norm: f = \frac{v}{\sqrt{\| v \| + e^2}} \tag{3}$$

Followed by clipping (limiting the minimum values of v to 0.2) and renormalizing. From the experiment conducted by [28], $L2-hys$, $L2-norm$ and $L1-sqrt$ schemes performed well while $L1-norm$ gives slightly less performance.

A similar experiment was performed by Irhebhude et al.[31] using HOG features for vehicle type recognition, recording a good accuracy rate of 95%. Patel et al.[13] recognized human actions in a video sequence using HOG-based fusion of features, achieving recognition accuracies between 84% and 99%. Ayalew et al. [32] proposed a hybrid CNN and HOG method for diagnosing COVID-19 from chest X-ray images of patients, which achieved 99.97% and 99.67% training and testing accuracy, respectively. Leveraging the strength of HOG as a powerful feature descriptor on computer vision and detection in image analysis, Bhattarai et al. [33] applied HOG for generating pseudo labels in medical image segmentation and achieved 70.2% accuracy.

2.2 Region Descriptors

Descriptors are similar to sets of numbers produced to describe an image, i.e., they attempt to quantify shape in a way that agrees with human thinking or specific task requirements [34]. Regions are the boundary-based properties of an object; Region descriptors generate a numeric feature vector or non-numeric syntactic description word that characterizes the properties of a described region[35].

Data or object properties can be described as boundary or complete regions based on representation and region description. Boundary descriptors are more applicable when emphasizing external shape characteristics like corners, while region-based descriptors are better appropriate for texture or skeletal shape properties; using a combination of boundary and region descriptors is also a common practice. Some measures used as region descriptors, as described by Gonzalez et al. [36], are;

1. Area (A) is the total number of pixels in an image.
2. Perimeter (P) is the number of pixels or lengths of the boundary of an image.
3. Compactness of a region (CR)is defined by the formula as shown in Equation (4);

$$P^2/A \tag{4}$$

4. Circularity ratio R_c is the ratio of a region's area to a circle's area having the same perimeter. The circularity ratio is expressed in Equation (5).

$$R_c = \frac{4\pi A}{P^2} \tag{5}$$

5. Eccentricity is the ratio of the distance between the center and the length of the major axis of an ellipse. The eccentricity of a region is calculated as shown in Equation (6)[37].

$$E = \sqrt{1 - \left(\frac{MINr}{MAXr}\right)^2} \tag{5}$$

Where $MINr$ is the minimum distance between the boundary of an image and its center, and $MAXr$ represents the maximum distance between the center and boundary of an image.

6. Major axis length: the length in pixels of the major axis of the ellipse of the region[37].
7. Minor axis length: the length in pixels of the minor axis of the region's ellipse region[37].
8. EquivDiameter (ED) calculates the diameter of a circle with the same area as the region of interest, computed as shown in Equation (7).

$$ED = \sqrt{4 \times \frac{A}{\pi}} \tag{7}$$

9. Euler Number (EN) is the number of connected object components (C) in the region minus the number of holes(H) in the objects as defined in Equation (8).

$$E = C - H \tag{8}$$

10. ConvexArea: is the number of pixels in a convex image.
11. Maximum Feret diameter measures the maximum distance between any two boundary points on the vertices that fully enclose the object[38].
12. Minimum Feret diameter measures the minimum distance between any two boundary points on the vertices that fully enclose the objec[38].
13. MaxFeretAngle: is the angle of the maximum Feret diameter[38].

In Irhebhude et al. [31], area, perimeter, and other region descriptors were used for selecting discriminative features for automatic vehicle type recognition, resulting in improved recognition accuracy. Other simple measures are mean, median of intensity levels, minimum and maximum intensity values, and number of pixels with values above and below the mean [36]. In obstacle crossing, actions are usually based on specific body movements that will eventually form shapes. The HOG and region descriptors are useful in capturing the shape, counting the occurrence of gradient variation, and identifying global patterns in the action region.

2.3 Multi-layer Perceptron (MLP)

The MLP is a feedforward ANN and the most commonly used type of neural network for classification and prediction, mapping input to appropriate output sets [39]. It is a nonlinear mapping between input and output vectors and consists of nodes connected by weights and output signals. They are all a function of the sum of input to the node and are modified by a nonlinear simple transfer or activation function. The architecture of MLP consists of the input, hidden, and output layers. It is called fully connected when each node is connected to every node in the next and previous layer. The input layer transfers the input vector to the network; no computation is done at this layer. The input and output vectors are referred to as the input and output of the MLP and can be represented as single vectors[40].

Javed et al. [26] applied the MLP classifier for recognizing physical activity. They achieved 93% accuracy, which is higher than other existing methods and shows the proposed model's ability to increase the recognition rate of physical activities.

3. Proposed Method

This section describes the proposed methodology as depicted in Figure 2. The aim is to correctly find and recognize human actions using a fusion of features extracted from HOG and Region descriptors called the HOGReG model. This technique is based on grabcut segmentation, feature extraction using two techniques, and classification with a neural network classifier. The recognition system involves different steps and techniques for extracting features from images relating to human activities in obstacle-crossing exercises. The steps in the proposed methodology involve the input image of size 448×252, after which segmentation divides the images into separate constituent parts: the background and foreground. The segmented images give two outputs of masked and black/white images, while the segmented black/white images are in the foreground, giving the image shape description. Figure 3 describes the different actions during obstacle-crossing exercises and the class number to which they belong.

3.1. Dataset Collection

The dataset used consists of videos of locally captured trainees (cadets) participating in obstacle-crossing exercises in a military training institution. The data in the form of videos were captured during the cadet's obstacle crossing exercise in the Nigerian Defence Academy (NDA), Nigeria, using the Mavic 2 Enterprise drone. The drone has a high-resolution visual camera and thermal sensors with top ports for mounting accessories and 24GB onboard memory storage. Frames, which are referred to as still images, are extracted from the videos and used as input images for the experiment. The video clips were framed into images, which served as input. The dataset contains 15 different activities, namely, clear jump, barbed wire crawling, 6/ft wall climbing, scrabble net, hand or monkey bridge crossing, Tarzan rope, 9/7ft ditch, tunnel, Niger bridge, balancing, rough & tumbling, high wall with the ladder, high wall tire ladder minefield, and horizontal & vertical wall. Each activity class contains 500 images, totaling 7500 used for the experiment. Figure 3 shows selected sample images of the actions their description used as input captured, which are considered private and restricted due to the classified nature of military training exercises.

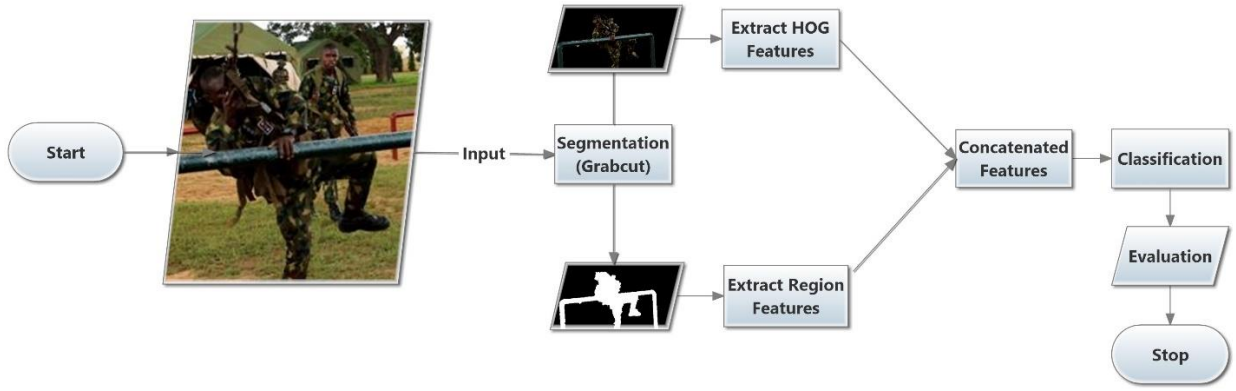

Figure 2. Proposed Methodology

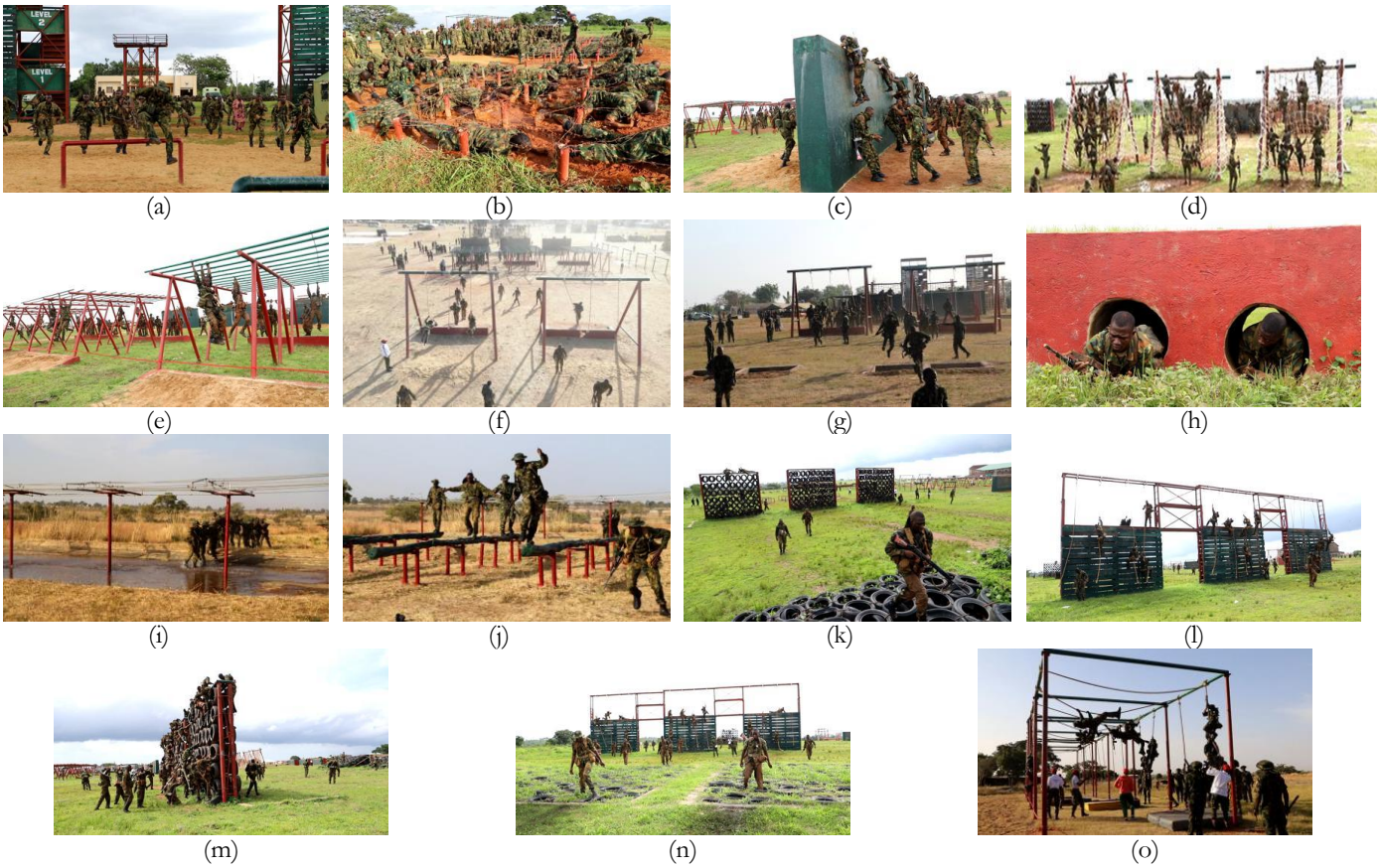

Figure 3. Sample image of human actions in obstacle crossing in military environment; (**a**) Clear jump; (**b**) Barbed wire crawling; (**c**) 6/Ft wall climbing; (**d**) Scrabble net; (**e**) Hand or monkey bridge crossing; (**f**) Tarzan rope; (**g**) 9/7ft ditch; (**h**) Tunnel; (**i**) Niger bridge; (**j**) Balancing; (**k**) Rough and tumbling; (**l**) High Wall with ladder; (**m**) High wall tire ladder; (**n**) Minefield; (**o**) Horizontal and vertical wall.

The first preprocessing involves framing the videos with a video framing algorithm[30] done using MATLAB SOFTWARE due to the large video dimension. The first stage of pre-processing is done, which involves resizing the original video dimension of 1280×1720 while maintaining the aspect ratio to 448×252, and this was achieved by framing the obtained video samples using the video framing algorithm. The resizing was necessary because of the limited computing resources and for easy framing.

3.2. Segmentation

Segmentation involves partitioning the input image into constituent parts or objects[2]. It involves grouping similar pixels to represent meaningful regions in an image. The mapping

out and separation of different parts of the input image is achieved using the Grabcut algorithm implemented in MATLAB R2023a through the image segmenter application. This determines if the individual pixels from the input image are part of the background or object to be identified. The algorithm selects one individual at a time and segments the input image into masked and Black/White (BW) Images. The masked segmented images represent the foreground, while the BW images give the shape description of the foreground. The procedures of the GrabCut algorithm are given as follows[41]:

1. Input the image. The segmentation process is initialized by drawing a shape around the selected labeled image U that connects all the edges, and this defines the foreground region. The region inside U are the foreground objects called F, while the region located outside are the background objects B.
2. For each pixel $p, p \in F$, assign a label $L_p = 1$, and for pixel $p, p \in B$, assign label $L_p = 0$
3. Using the K-means clustering algorithm, F and B Object region is clustered into K kinds of pixels.
4. Initialize the GMM of the foreground and the background with the two sets of labels $L_p = 0$and $L_p = 1$, respectively (the GMM of the foreground and the background both have Gaussian components), and the parameters (π, μ, θ) of the two GMMs are obtained. Where π is the weight of each Gaussian component, μ is the mean vector of each Gaussian component, and θ represents the covariance matrix.
5. Substituting each pixel p in the foreground object region F into the two obtained GMMs will obtain the probability of which pixel belongs to the foreground object region or the background region, respectively. The highest probability generates the pixel p,(that is, the Gaussian component K_p of the pixel). The probability is in the form of the negative logarithm to obtain the regional term F.
6. The Euclidean distance (i.e., the two norms) between all two neighboring pixels in F is calculated V is obtained as the boundary term.
7. Obtain the minimum value of energy min $E\ (A, k, \theta, P)$ using the maximum flow minimum cut algorithm. The calculated result is reassigned to the set of pixels $L_p = 0$and $L_p = 1$ in the foreground object region.
8. Repeat steps 4 through 7 until the convergence and output image.

The algorithm segments the input image into two categories based on the foreground and background, which are then utilized in the next stage and further used for feature extraction.

3.3. Feature Extraction

From the proposed methodological flow diagram in Figure 2, HOG and Region descriptors extract features from the segmented images. The segmented masked is the input to the HOG model, while the segmented BW image is used as input for the region descriptor.

The region descriptors (RD) extract the following statistical feature measurement from the segmented BW images using MATLAB function "regionprops"; Area (A), major axis length(MaL), minor axis length(Mil), eccentricity (Ec), orientation (O), convex area(Ca), circularity ratio(Cr), filled area(Fa), Euler number (En), Equidiameter(Ed), perimeter(P), perimeter old(Po), maxferet diameter(MaD), maxferet angle (MaA), miniferet diameter (MiD) and miniferert angle(MiA). The RD measures the properties of the image regions and describes the characteristics of the image as shown in Equation (9)[42].

$$RD = [A, MaL, MiL, Ec, O, Ca, Cr, Fa, En, Ed, P, Po, MaD, MaA, MiD, MiA] \tag{9}$$

The HOG feature extraction technique makes it easier to count the occurrences of oriented gradients in localized portions of an image[18]. HOG descriptor focuses on local shape information, texture, and shape extraction for easy object detection. The HOG features are extracted from the segmented masked images using the following algorithm[29]:

1. Prepare the input image – take the input image and resize it to size 42×42

2. Calculate the image's gradient by combining the image's magnitude and angle. Considering a block with the input size, firstly, the g_x and g_y is calculated using the formula given in Equations (10) and (11).

$$g_x(r,c) = I(r,c+1) - I(r,c-1) \tag{10}$$

$$g_y(r,c) = I(r-1,c) - I(r+1,c) \tag{11}$$

Where r and c are the rows and columns, respectively.
The magnitude and angle of each pixel is calculated from the gradient g_x and g_y is using formulas in Equations (12) and (13).

$$Magnitude(\mu) = \sqrt{g_x^2 + g_y^2} \tag{12}$$

$$Angle\ (\theta) = |tan^{-1}\left({}^{g_y}/_{g_x}\right)| \tag{13}$$

3. Divide the gradient matrices to form a block - the gradient matrices (magnitude and angle matrix) are divided into 8×8 cells to form a block. A nine-point histogram is calculated for each block, and a histogram with nine bins is developed, with each bin having an angle range of 20 degrees.
4. Calculate the j_{th} Bin - For each cell in a block, the j_{th} bin is calculated, and then the value that will be provided to the j_{th} and $j+1_{th}$ bin respectively.
5. Calculate the arrays for each Pixel - An array is taken as a bin for a block, and values are appended in the array and j_{th} and $j+1_{th}$bin calculated for each pixel.
6. Histogram Computation is complete for each block.
7. After the histogram computation over all blocks, four blocks from the nine-point histogram matrix are clubbed together to form a new block (2×2). This clubbing is done overlappingly with a stride of eight pixels.
8. Calculate the L2 Norm
9. Normalize the value
10. Obtain HOG features

The output of HOG extracted features is returned as described as HOG feature-length (N). The feature length is based on the image size and the function parameter values described in Equations (14) and (15).

$$BPI = \left\lfloor \left(\frac{\left(\frac{AS(I)}{CS} - BS\right)}{BS - BO} + 1 \right) \right\rfloor \tag{14}$$

$$N = \prod([BPI, BS, NB]) \tag{15}$$

Where BPI = blocks per image; AS = size of array; I = input image; CS = size of the HOG cell; BS = number of cells in the block; BO = number of overlapping cells between adjacent blocks; N represents the HOG feature-length, and NB is the number of orientation histogram bins, here 576 features are extracted.

The extracted HOG and region features fused by concatenating with the horzcat function in MATLAB, horzcat indicates a function for concatenating arrays horizontally[43]. The horzcat function combines the extracted features from the region and HOG. The syntax for concatenation is shown in Equations (16) and (17)

$$c = horzcat(A, B) \tag{16}$$

$$c = horzcat(A_1, A_{2,\ldots,} A_n) \tag{17}$$

Equation 16 concatenates A horizontally to the end of B when A and B have compatible sizes (the lengths of the dimensions match except in the second dimension). In contrast, equation 17 horizontally concatenates the arrays, with all inputs from vectors and matrices having the same number of rows[44]. The horzcat algorithm used for concatenation is described as follows:

1. Create two matrices input specified as a scalar, vector, matrix, multidimensional array, table, or timetable, and horizontally append the second matrix to the first by using square bracket notation.
2. Horizontally concatenate the second matrix to the first using horzcat() function.
3. horzcat omits an empty array from the output when concatenating it to a nonempty array.
4. If all the input arguments are empty and have compatible sizes, horzcat will return an empty array whose size is the same as the output size when the inputs are not empty. Applying the algorithm concatenates the combined features (C) of region and HOG, as shown in Equation (18).

$$C = horzcat(RD, N) \tag{18}$$

3.3. Classification using Artificial Neural Network

The MLP classifier is used for classification in this paper, as it maps out input data sets to a set of given outputs. After getting the already preprocessed data from the previous blocks, the model is trained using the MLP network. Different parameters were used, such as hidden layer size, activation function, number of epochs, and algorithm for weight optimization node. The algorithm's performance is measured using accuracy, confusion matrix, and Receiver Operating Characteristic (ROC) curve. The three models used for the experiment, HOG, Region features, and a combination of HOG and region (HOGReg), had the same hyperparameter tuning and values shown in Table 1.

Table 1. Hyperparameters for HOG, Region descriptor, and HOGReg.

Hyperparameter	Value
Preset	Wide Neural Network
Number of fully connected layer	1
First layer size	100
Activation	ReLU
Iteration Limit	100
Regularization strength (Lambda)	0
Standardized data	Yes

3.3. Performance Evaluation Metrics

For easy view and evaluation of the performance of the model, confusion matrix, accuracy, receiver operating characteristics (ROC) curve, true positive rate, false positive rate, positive predictive rate, and false discovery rate were used because of their evaluation efficiency[45], [46].

Confusion matrix – confusion matrices are tables generally used to visualize the performance of a model through a collection of datasets with known true values[47]. The rows in the matrix display the instances of predicted values, while the columns show actual values. True predictions are represented by the diagonal, while others display the errors. This makes it easy to check for errors in the model.

Accuracy measures the percentage of correctly identified cases out of the total, as shown in the formula in Equation (19)[46].

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{19}$$

Where True Positive (TP) is the number of correctly categorized classes belonging to the positive class, while True Negative (TN) is the number of correctly categorized classes belonging to the negative class. False Positive (FP) and False Negative (FN) represent the

number of incorrectly categorized classes belonging to the positive and the negative classes, respectively[48].

True Positive Rate (TPR) is calculated as the number of accurate positive predictions (TP) divided by the total number of positives (P), as shown in Equation (20). It is also called sensitivity or recall. The best TPR value is 1.0[49]. The False negative rate (FNR) - is the proportion of positive samples that were incorrectly classified as defined in Equation (21) [50]. Positive prediction value (PPV) - is the proportion of correctly identified positive samples to the total number of positive predicted samples as shown in Equation (22)[50]. False Discovery Rate (FDR) - is the number of false positive results divided by all positives, as shown in Equation (23)[49].

$$\text{TPR} = \frac{TP}{TP + FN} \tag{20}$$

$$\text{FNR} = \frac{FP}{FP + TN} \tag{21}$$

$$\text{PPV} = \frac{TP}{TP + FP} \tag{22}$$

$$\text{FDR} = \frac{FP}{FP + TP} \tag{23}$$

ROC Curve – a graph that shows the difference or relationship between the true positive rate (TPR) and the false positive rate (FPR). The true and false positive rates are calculated and plotted in one graph. For each threshold, a higher TPR and low FPR show better performance, and the more the curve tilts to the left, the better the classifier performance is [49]. The area below a ROC curve is the area under the curve (AUC), which always has a value between 0 and 1 that indicates how good or bad the ROC curve performs. A value of 1 indicates a perfect performance.

4. Experimental Analysis

The proposed methodology was implemented in MATLAB R2023a, using Grabcut as the segmentation algorithm. Some selected results from the segmentation are shown in Figure 4.

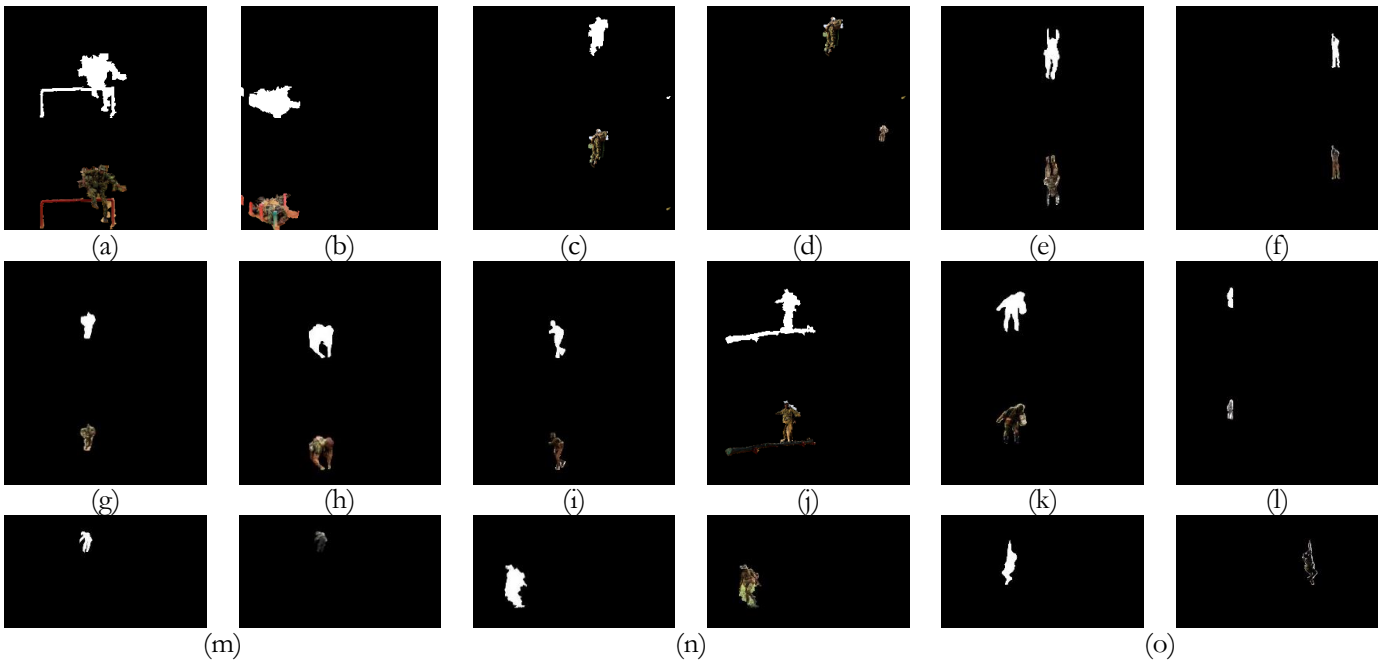

Figure 4. Segmentation results BW images and Masked images; (**a**) Clear jump; (**b**) Barbed wire crawling; (**c**) 6/Ft wall climbing; (**d**) Scrabble net; (**e**) Hand or monkey bridge crossing; (**f**) Tarzan rope; (**g**) 9/7ft ditch; (**h**) Tunnel; (**i**) Niger bridge; (**j**) Balancing; (**k**) Rough and tumbling; (**l**) High Wall with ladder; (**m**) High wall tire ladder; (**n**) Mine field; (**o**) Horizontal and vertical wall.

Performance evaluation was done on the experiment using the proposed feature extraction technique, a combination of HOG and region (HOGReG) with a neural network classifier. 70% of the dataset was used for training, while 30% was used for testing. The confusion matrix curve, TPR/FNR, PPV/FDR, and accuracy values present the results. The confusion matrices for the HOGReG model are presented in Figures 5 to 7. The results obtained are discussed as follows:

4.1. Evaluation with HOG and Region (HOGReG)

The recognition results using the proposed model HOGReG features are presented in Figures 5 to 7. A total of 592 individual features were selected to achieve a maximum accuracy of 86.4% with a training time of 31.975secs. The confusion matrix in Figure 5 visually represents the results and the effect of using a combination of HOG and region features.

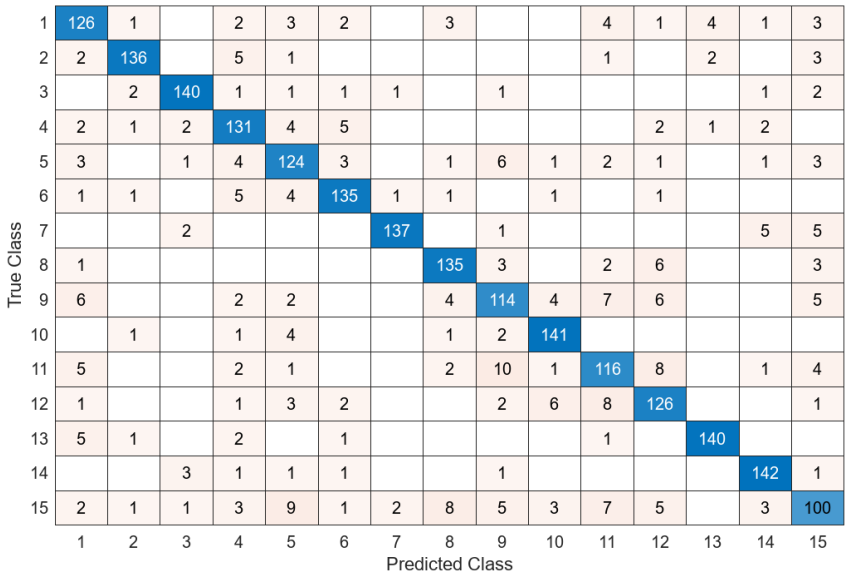

Figure 5. Confusion Matrix with HOGReG

The highest number of correct observations was obtained by Class 14, with 142 correct positive observations, as shown in Figure 5. For the horizontal and vertical action, class 15 recorded the lowest number of correct true classes (100). This is a similar pattern across the other two models. The high wall tire ladder (class 13) and 6/ft wall action (class 3) had the same number of correct true classes, which stood at 140 each.

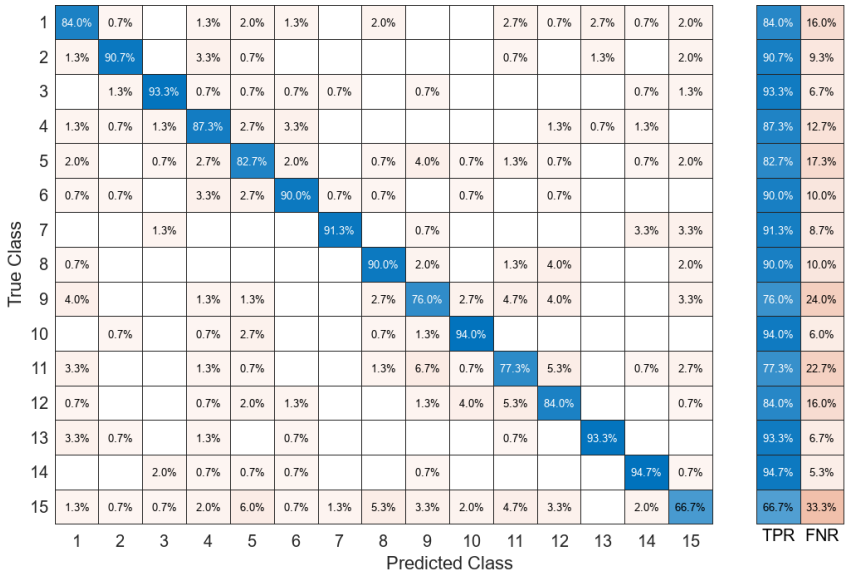

Figure 6. Confusion Matrix Showing TPR and FNR with HOGReG

The confusion matrix showing the TPR/ FNR and PPV/FDR values for the proposed HOGReG features is shown in Figures 6 and 7. With a TPR of 94.7%, as seen in Figure 6, class 14 had the highest percentage of positive correct classification, which aligns with the confusion matrix of the number of observations reported in Figure 5. The highest percentage of incorrect actions was recorded by class 15, with an FNR of 33.3%, which shows a high rate of misclassification spread across other classes. Class 7, with a PPV of 97.2%, had the most correct positive samples predicted to the right class, while class 15 had the highest percentage of wrong prediction with an FDR value of 23.1%, as shown in Figure 7.

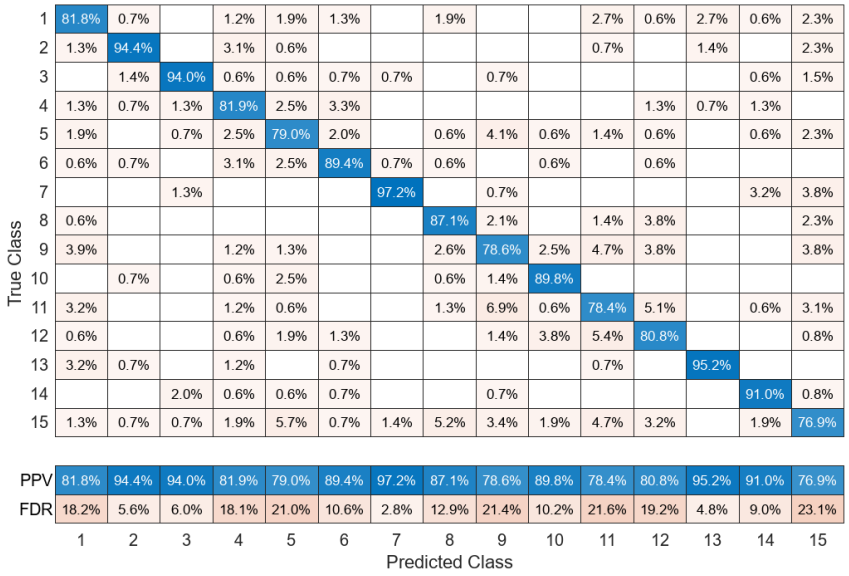

Figure 7. Confusion Matrix Showing PPV and FDR with HOGReG

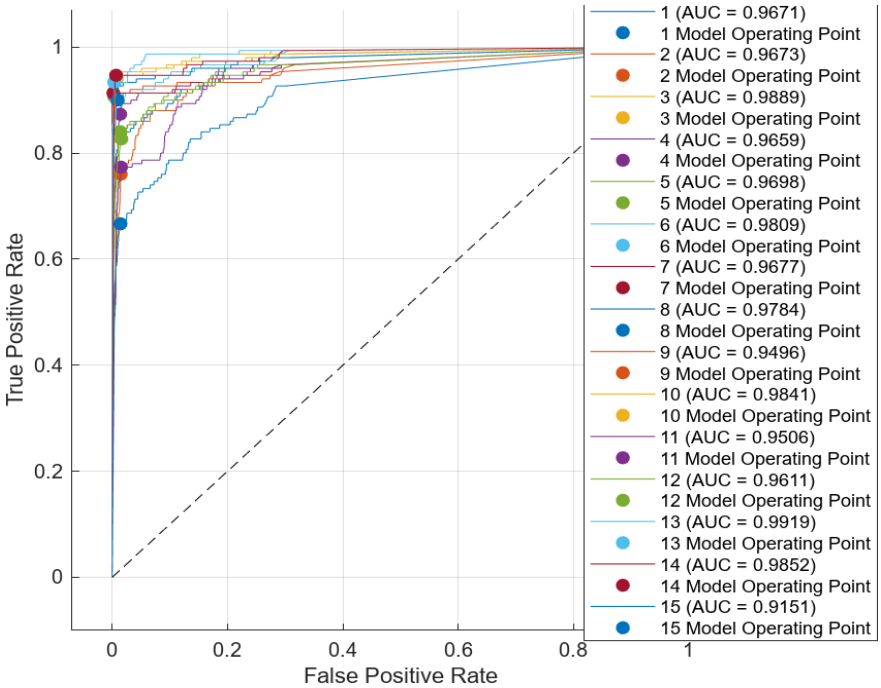

Figure 8. ROC for HOGReG Features

From Figure 8, the value of AUC for each class is shown, with class 12 having the highest AUC of 0.9919, which means the model has a 99.19% probability of distinguishing between actions in class 12 and the other classes. The action recognition was also evaluated using the individual extracted features of region and HOG. The evaluation is reported in the next subsection.

4.2. Evaluation with Region Features (ReG)

A total of 16 individual features were extracted, and an overall accuracy of 63.8% was achieved during the 5.6048 sec training time. Class 13 (High wall tire ladder) had the highest number of correctly classified actions; 135 true positive instances of actions were correctly classified to belong to the true class, followed by class 2 (barbed wire) and class 3 (6/ft wall climbing) which had 132 and 131 correct number of observations respectively. This result demonstrates how well the classifier understood the distinction between class 13 and the other classes. Classes 4 (scrabble Net) and 12 (horizontal and high wall with ladder) have the highest misclassification rate, with the misclassified actions spread across other classes of actions. From the confusion matrix, it is clear that the region descriptor performs better in handling high wall tyre ladder, barbed wire and 6/ft wall climbing actions compared to other classes due to the similar action movement in these classes.

Class 13 also recorded the highest percentage of positive correct classes with a TPR value of 90%. The highest percentage of incorrect actions was recorded by class 12 with an FNR value of 59.7%, which shows a high rate of misclassification spread across other actions. Similarly, class 13 had the most correct PPV of 79.94%, while class 9 (Niger Bridge) had the highest percentage of wrong prediction with an FDR value of 54.1%, indicating that the pattern and shape of movement while performing this action made it difficult for the classifier to identify these activities. Class 11 had the highest AUC value recorded as 09594, which indicates a better performance than other classes, while class 4 recorded the lowest AUC of 0.8068.

4.3. Evaluation with HOG Features (HOG)

For experimental results with HOG descriptors, 576 features were extracted. These features were fed into the learning algorithm, and classification was done with a neural network. A training time of 32.233 seconds was recorded with an overall accuracy of 82.6% achieved which is an improvement from the Region Descriptor. Class 14 (Minefield) had the highest number of correctly predicted classes (144), followed by class 3 (6/ft wall) with 143 true positive, class 7 (9/ft wall) and class 13 (high wall) with 142 true positive number of observations each.

The Horizontal & vertical action (class 15) recorded the lowest number of correctly classified classes (90), with 60 misclassified classes. It showed that this class has a high rate of confusion that spreads across other class actions. The overall performance shows HOG had an improved recognition rate. Even with the increased training time, the model still performed well.

With a TPR value of 96.0% class 14 had the highest percentage of positive correct classified action class. Class 15 recorded the highest percentage of incorrect actions with an FNR value of 40.0%, with a high rate of misclassification spread across other actions. Class 14 also had the highest positive samples predicted to the correct class with a PPV value of 94.7% while class 11 had the highest percentage of wrong prediction with a FDR value of 35.5%.

Class 3 reported the highest AUC of 0.9891, meaning the model has 98.91% probability of distinguishing between class 3 and the other classes, while class 15 reported the lowest AUC value of 0.8916.

4.4. Analysis of Results

Table 2 presents a comparative performance analysis between the proposed technique and the other models used. The evaluation results presented from the experiment conducted using region descriptor and HOG as individual features and a combination of the two features HOGReG, which is the proposed model, showed that the HOGReG features attained the overall highest accuracy of 86.4% with a training time with a reduced training time of 31.975 seconds, this is summarized in Table 2. The reported TPR/FNR, PPV/FDR, and ROC are the average computed scores reported in Figures 6 to 8. The proposed HOGReG recorded lower training time, AUC, and accuracy scores than HOG because of its more robust feature representation of the various actions classified. The experimental findings clearly show that the proposed technique recorded better performances except for the TPR, where HOG performed slightly better.

Table 2. Comparison of Feature Techniques Results on Action Classification in an Obstacle Crossing Competition.

Features	Accuracy (%)	TPR/FNR (%)	PPV/FDR (%)	AUC	Training Time(secs)
ReG	63.8	63.7/36.2	64.2/35.7	0.9122	5.6048
HOG	82.6	82.6/21.7	82.7/17.3	0.9542	32.233
HOGReG	86.4	81.1/13.6	86.4/13.6	0.9682	31.975

Reg feature is more efficient in classifying actions in classes 11 and 13, HOG is more efficient in classes 3, 13, and 14, while HOGReG is more efficient in classes 7, 13, and 14. It is also observed that among all the classes of activities, the three feature extraction techniques found it easier to handle and differentiate actions in class 13 (High wall tire ladder activity). This class activity involves actions with speed and multiple actions taking place, i.e., climb and jump movements. Generally, class 15 had low performance across the three models due to more complexity in the pattern of the activity as it involves the use of crawling movement with both legs and hands and jumping.

The HOGReG features, the proposed model for recognizing human actions in obstacle crossing activities, show its ability to give more accurate prediction with an overall accuracy of 86.4% compared to the other results obtained from the experiment and also agree with the results obtained by Patel [13]. The increase in performance lies in the fusion of the features extracted using HOG and region descriptors, and the type of actions performed determines how well the model can easily recognize them.

5. Conclusions

In conclusion, this paper used a neural network classifier to classify human action in a military obstacle-crossing competition into their correct categories by concatenating HOG and region descriptor features. With an overall accuracy of 86.4%, the proposed model, HOGReG, gave better accuracy with reduced training time compared to results obtained from individual HOG and regions, which gave 82.6%. These findings demonstrate the suggested model's ability to recognize human actions during military training exercises such as obstacle-crossing activity. However, a few instances of misclassification could result from similar actions performed in these classes. A more robust feature extraction technique can be utilized in the future, and a deep learning model can be introduced to improve accuracy further.

Author Contributions: The authors hereby provide information regarding contribution Conceptualization: Adeola O. Kolawole and Martins E. Irhebhude; methodology, Adeola O. Kolawole; software: Martins E. Irhebhude; validation: Adeola O. Kolawole., Martins E. Irhebhude and Philip O. Odion; formal analysis: Martins E. Irhebhude; investigation: Adeola O. Kolawole; resources: Philip O. Odion; data curation: Adeola O. Kolawole; writing—original draft preparation: Adeola O. Kolawole; writing—review and editing: Martins E. Irhebhude; visualization: Adeola O. Kolawole; supervision: Martins E. Irhebhude; project administration: Philip O. Odion; funding acquisition: Adeola O. Kolawole All authors have read and agreed to the published version of the manuscript

Funding: Please add: This research received no external funding.

Acknowledgments: Authors will like to acknowledge the support by Nigerian Defence Academy for providing the enabling environment and data support during the course of this study

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] I. Jegham, A. Ben Khalifa, I. Alouani, and M. A. Mahjoub, "Vision-based human action recognition: An overview and real world challenges," *Forensic Sci. Int. Digit. Investig.*, vol. 32, p. 200901, Mar. 2020, doi: 10.1016/j.fsidi.2019.200901.

[2] A. Kumar, "What is image processing? meaning, techniques, segmentation & important facts to know," *Simplilearn*, 2024. https://www.simplilearn.com/image-processing-article#:~:text=One of the most common,distance between the eyes%2C etc.
[3] Y. Kong and Y. Fu, "Human Action Recognition and Prediction: A Survey," *Int. J. Comput. Vis.*, vol. 130, no. 5, pp. 1366–1401, May 2022, doi: 10.1007/s11263-022-01594-9.
[4] K. Host and M. Ivašić-Kos, "An overview of Human Action Recognition in sports based on Computer Vision," *Heliyon*, vol. 8, no. 6, p. e09633, Jun. 2022, doi: 10.1016/j.heliyon.2022.e09633.
[5] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang, and Z. Li, "A Review on Human Activity Recognition Using Vision-Based Method," *J. Healthc. Eng.*, vol. 2017, pp. 1–31, 2017, doi: 10.1155/2017/3090343.
[6] N. Gupta, S. K. Gupta, R. K. Pathak, V. Jain, P. Rashidi, and J. S. Suri, "Human activity recognition in artificial intelligence framework: a narrative review," *Artif. Intell. Rev.*, vol. 55, no. 6, pp. 4755–4808, Aug. 2022, doi: 10.1007/s10462-021-10116-x.
[7] Y. Li and L. Wang, "Human Activity Recognition Based on Residual Network and BiLSTM," *Sensors*, vol. 22, no. 2, p. 635, Jan. 2022, doi: 10.3390/s22020635.
[8] K. Soomro and A. R. Zamir, "Action Recognition in Realistic Sports Videos," in *Computer Vision in Sports*, T. B. Moeslund, G. Thomas, and A. Hilton, Eds. Cham: Springer International Publishing, 2014, pp. 181–208. doi: 10.1007/978-3-319-09396-3_9.
[9] B. Russell, A. McDaid, W. Toscano, and P. Hume, "Moving the Lab into the Mountains: A Pilot Study of Human Activity Recognition in Unstructured Environments," *Sensors*, vol. 21, no. 2, p. 654, Jan. 2021, doi: 10.3390/s21020654.
[10] S. Patil, S. Shelke, S. Joldapke, V. Jumle, and S. Chikhale, "Review on Human Activity Recognition for Military Restricted Areas," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 603–606, Dec. 2022, doi: 10.22214/ijraset.2022.47926.
[11] M. A. Khan, M. Sharif, T. Akram, M. Raza, T. Saba, and A. Rehman, "Hand-crafted and deep convolutional neural network features fusion and selection strategy: An application to intelligent human action recognition," *Appl. Soft Comput.*, vol. 87, p. 105986, 2020, doi: 10.1016/j.asoc.2019.105986.
[12] B. Gahtan, S. Funk, E. Kodesh, I. Ketko, T. Kuflik, and A. M. Bronstein, "WearableMil: An End-to-End Framework for Military Activity Recognition and Performance Monitoring," *ArXiv*. Oct. 07, 2024. [Online]. Available: http://arxiv.org/abs/2410.05452
[13] C. I. Patel, D. Labana, S. Pandya, K. Modi, H. Ghayvat, and M. Awais, "Histogram of Oriented Gradient-Based Fusion of Features for Human Action Recognition in Action Video Sequences," *Sensors*, vol. 20, no. 24, p. 7299, Dec. 2020, doi: 10.3390/s20247299.
[14] E. Sansano, R. Montoliu, and Ó. Belmonte Fernández, "A study of deep neural networks for human activity recognition," *Comput. Intell.*, vol. 36, no. 3, pp. 1113–1139, Aug. 2020, doi: 10.1111/coin.12318.
[15] H. E. Azzag, I. E. Zeroual, and A. Ladjailia, "Real-Time Human Action Recognition Using Deep Learning," *Int. J. Appl. Evol. Comput.*, vol. 13, no. 2, pp. 1–10, Dec. 2022, doi: 10.4018/IJAEC.315633.
[16] B. Ren, M. Liu, R. Ding, and H. Liu, "A Survey on 3D Skeleton-Based Action Recognition Using Learning Method," *Cyborg Bionic Syst.*, vol. 5, p. 100, Feb. 2020, [Online]. Available: http://arxiv.org/abs/2002.05907
[17] M. M. Islam, S. Nooruddin, F. Karray, and G. Muhammad, "Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects," *Comput. Biol. Med.*, vol. 149, p. 106060, Oct. 2022, doi: 10.1016/j.compbiomed.2022.106060.
[18] P. M. D. Alex, A. Ravikumar, J. Selvaraj, and A. Sahayadhas, "Research on Human Activity Identification Based on Image Processing and Artificial Intelligence," *Int. J. Eng. Technol.*, vol. 7, no. 3.27, p. 174, Aug. 2018, doi: 10.14419/ijet.v7i3.27.17754.
[19] K. P. S. Kumar and R. Bhavani, "Human activity recognition in egocentric video using HOG, GiST and color features," *Multimed. Tools Appl.*, vol. 79, no. 5–6, pp. 3543–3559, Feb. 2020, doi: 10.1007/s11042-018-6034-1.
[20] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2004, vol. 3, pp. 32-36 Vol.3. doi: 10.1109/ICPR.2004.1334462.
[21] P. S. Tan, K. M. Lim, and C. P. Lee, "Human Action Recognition with Sparse Autoencoder and Histogram of Oriented Gradients," in *2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, Sep. 2020, pp. 1–5. doi: 10.1109/IICAIET49801.2020.9257863.
[22] D. K. Singh, "Human Action Recognition in Video," in *Advanced Informatics for Computing Research*, D. Singh, P.-A. Hsiung, K. B. G. Hawari, P. Lingras, and P. K. Singh, Eds. Singapore: Springer Singapore, 2019, pp. 54–66. doi: 10.1007/978-981-13-3140-4_6.
[23] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, A. Bouridane, and A. Beghdadi, "A combined multiple action recognition and summarization for surveillance video sequences," *Appl. Intell.*, vol. 51, no. 2, pp. 690–712, Feb. 2021, doi: 10.1007/s10489-020-01823-z.
[24] S. P. Sahoo, R. Silambarasi, and S. Ari, "Fusion of histogram based features for Human Action Recognition," in *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, Mar. 2019, pp. 1012–1016. doi: 10.1109/ICACCS.2019.8728473.
[25] S. Gundu and H. Syed, "Vision-Based HAR in UAV Videos Using Histograms and Deep Learning Techniques," *Sensors*, vol. 23, no. 5, p. 2569, Feb. 2023, doi: 10.3390/s23052569.
[26] A. R. Javed, M. U. Sarwar, S. Khan, C. Iwendi, M. Mittal, and N. Kumar, "Analyzing the Effectiveness and Contribution of Each Axis of Tri-Axial Accelerometer Sensor for Accurate Activity Recognition," *Sensors*, vol. 20, no. 8, p. 2216, Apr. 2020, doi: 10.3390/s20082216.
[27] M. Ha, "Top-Heavy CapsNets Based on Spatiotemporal Non-Local for Action Recognition," *J. Comput. Theor. Appl.*, vol. 2, no. 1, pp. 39–50, May 2024, doi: 10.62411/jcta.10551.
[28] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 886–893. doi: 10.1109/CVPR.2005.177.
[29] M. Tyagi, "Histogram of Oriented Gradients: An Overview," *builtin*, 2024. https://builtin.com/articles/histogram-of-oriented-gradients
[30] M. E. Irhebhude, "Object Detection, Recognition and Re-identification in Video Footage," Loughborough University, 2015. [Online]. Available: https://repository.lboro.ac.uk/articles/thesis/Object_detection_recognition_and_re-identification_in_video_footage/9406961?file=17024141

[31] M. E. Irhebhude, A. Nawahda, and E. A. Edirisinghe, "View invariant vehicle type recognition and counting system using multiple features," *Int. J. Comput. Vis. Signal Process.*, vol. 6, no. 1, pp. 20–32, 2016.
[32] A. M. Ayalew, A. O. Salau, B. T. Abeje, and B. Enyew, "Detection and classification of COVID-19 disease from X-ray images using convolutional neural networks and histogram of oriented gradients," *Biomed. Signal Process. Control*, vol. 74, p. 103530, Apr. 2022, doi: 10.1016/j.bspc.2022.103530.
[33] B. Bhattarai, R. Subedi, R. R. Gaire, E. Vazquez, and D. Stoyanov, "Histogram of Oriented Gradients meet deep learning: A novel multi-task deep network for 2D surgical image semantic segmentation," *Med. Image Anal.*, vol. 85, p. 102747, Apr. 2023, doi: 10.1016/j.media.2023.102747.
[34] B. S. Morse, "Lecture 9: Shape description (regions)," *Brigham Young University (1998–2000)*. 2000. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/region-props-and-moments.pdf
[35] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 4th ed. Boston, MA: Cengage Learning, 2014. doi: 10.1007/978-1-4899-3216-7.
[36] R. C. Gonzalez and W. Richard E, *Digital image processing*, 3rd ed. United States of America: Pearson Education Ltd., 2008.
[37] P. Sharma and R. S. Anand, "Depth data and fusion of feature descriptors for static gesture recognition," *IET Image Process.*, vol. 14, no. 5, pp. 909–920, Apr. 2020, doi: 10.1049/iet-ipr.2019.0230.
[38] L. Kandlbauer, K. Khodier, D. Ninevski, and R. Sarc, "Sensor-based Particle Size Determination of Shredded Mixed Commercial Waste based on two-dimensional Images," *Waste Manag.*, vol. 120, pp. 784–794, Feb. 2021, doi: 10.1016/j.wasman.2020.11.003.
[39] R. Kruse, S. Mostaghim, C. Borgelt, C. Braune, and M. Steinbrecher, "Multi-layer Perceptrons," in *Computational Intelligence*, Cham: Springer, Cham, 2022, pp. 53–124. doi: 10.1007/978-3-030-42227-1_5.
[40] M. Banoula, "An Overview on Multilayer Perceptron (MLP)," *Simplilearn*, 2023. https://www.simplilearn.com/tutorials/deep-learning-tutorial/multilayer-perceptron
[41] Z. Wang, Y. Lv, R. Wu, and Y. Zhang, "Review of GrabCut in Image Processing," *Mathematics*, vol. 11, no. 8, p. 1965, Apr. 2023, doi: 10.3390/math11081965.
[42] Mathworks, "regionprops," *Mathworks*, 2024. https://www.mathworks.com/help/images/ref/regionprops.html
[43] X. Cai and G. Han, "Background Subtraction Based on Modified Pulse Coupled Neural Network in Compressive Domain," *IEEE Access*, vol. 8, pp. 114422–114432, 2020, doi: 10.1109/ACCESS.2020.3003724.
[44] Mathworks, "horzcat; Concatenate arrays horizontally," *Mathworks*, 2024. https://www.mathworks.com/help/matlab/ref/double.horzcat.html
[45] M. E. Irhebhude, A. O. Kolawole, and F. Abdullahi, "Northern Nigeria Human Age Estimation From Facial Images Using Rotation Invariant Local Binary Pattern Features with Principal Component Analysis," *Egypt. Comput. Sci. J.*, vol. 45, no. 1, 2021.
[46] M. E. Irhebhude, A. O. Kolawole, and H. K. Goma, "A Gender Recognition System Using Facial Images with High Dimensional Data," *Malaysian J. Appl. Sci.*, vol. 6, no. 1, pp. 27–45, Apr. 2021, doi: 10.37231/myjas.2021.6.1.275.
[47] R. Raj and A. Kos, "An improved human activity recognition technique based on convolutional neural network," *Sci. Rep.*, vol. 13, no. 1, p. 22581, Dec. 2023, doi: 10.1038/s41598-023-49739-1.
[48] D. Bhattacharya, D. Sharma, W. Kim, M. F. Ijaz, and P. K. Singh, "Ensem-HAR: An Ensemble Deep Learning Model for Smartphone Sensor-Based Human Activity Recognition for Measurement of Elderly Health Monitoring," *Biosensors*, vol. 12, no. 6, p. 393, Jun. 2022, doi: 10.3390/bios12060393.
[49] Ž. Đ. Vujovic, "Classification Model Evaluation Metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.
[50] A. Tharwat, "Classification assessment methods," *Appl. Comput. Informatics*, vol. 17, no. 1, pp. 168–192, Jan. 2021, doi: 10.1016/j.aci.2018.08.003.