Received: May 29, 2023. Revised: September 1, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Effective Video Event Classification by Optimizing the Hyper-Parameters Using Improved Pelican Optimization and Bi-LSTM Classifier

Susmitha Alamuru[1]* **Sanjay Jain**[1]

[1]*CMR Institute of Technology, Bangalore, India*
* Corresponding author's Email: susmitha.academic@gmail.com

Abstract: In recent years, video event prediction and classification is considered a hot research topic among researchers, due to its social influence and its extensive real-time applications. In this research, a video event classification system is proposed to predict and categorize various day-to-day events. This research proposed an improved pelican optimization algorithm (IPOA) to optimize the hyper-parameters which aids in better classification accuracy. The input data is gathered from three well-known datasets such as University of Central Florida101 (UCF101), Human Motion Database 51 (HMDB51) and columbia consumer video (CCV) dataset. The raw data is pre-processed using the data normalization technique and the DenseNet 201 model is used to extract the features from the pre-processed output. The extracted features are fed into the stage of feature selection using the improved grey wolf optimization (IGWO) algorithm. The hyper-parameters of the selected features are optimized using the proposed IPOA algorithm and classification takes place using bidirectional long short term memory (Bi-LSTM) classifier. The experimental results show that the proposed IPOA with Bi-LSTM achieved a better classification accuracy of 93.91%, 94.19%, and 90.19% for datasets such as UCF101, HMDB51 and CCV respectively which is comparatively higher than the existing techniques such as improved residual convolutional neural network (CNN), Bi-LSTM CNN, two-stream 3D CNN model and gait event detection system.

Keywords: Bidirectional long-short term memory, Convolutional neural network, Improved grey wolf optimization, Improved pelican optimization algorithm, Video events.

1. Introduction

The recognition of events is considered one of the active areas in the field of computer vision due to the wide range of applications like the interaction between humans and computers, analysis of motion, analysis of medical images, etc [1]. The main goal of event recognition is to recognize the events such as parties, weddings, graduation, and also daily activities like shaving the beard, jogging, riding a bike, and so on [2]. In recognition of events, feature extraction plays a significant role in extracting the essential features from the videos [3, 4]. Since some of the videos have a high time duration, recognizing the action from the video is a challenging task when compared with image recognition [5, 6]. The heterogeneous structure of the videos acts as a barricade to provide an effective solution for recognizing the human activities from the video and this heterogeneity is due to messed backgrounds, the varied motion of the camera etc [7]. In video background, recognizing the activity is based on collecting consecutive video frames in which the motion of the individual body should be analyzed based on spatial and temporal information [8].

The recognition of human activities in various events takes place in three stages such as (i) pre-processing which is utilized in removing unnecessary noise from images, (ii) pre-processed information is processed for extraction of features and (iii) in the classification stage, the extracted features are mapped with the respective classes [9, 10]. The configuration and the temporal dependencies act rely as a major problem in video processing applications. The space among the action and activities is combined by encoding the features with mid-level representations [11, 12]. Considering the single concept prohibits the

DOI: 10.22266/ijies2024.0229.03

process of recognizing the individual's event. Additionally, the video sequence related to real-world applications is comprised of noisy surroundings which may affect the classification accuracy [13, 14]. The usage of deep neural networks (DNN) helps to overcome these issues due to its ability in transmitting the data without loss and enhance the detection accuracy [15].

The main contributions of this research are listed as follows:

1. This research proposed an IPOA to optimize the hyper-parameters and classify the video events using Bi-LSTM architecture. Moreover, the Improved GWO technique is utilized to select the relevant features from the extracted data.
2. The performance of the proposed method is evaluated using three familiar datasets such as UCF101, HMDB51, and CCV based on performance metrics such as accuracy, precision, sensitivity and F-1 score.

The rest of this research paper is organized as follows: Section 2 provides some recent year works on the detection and classification of video events. Section 3 describes the proposed work and the results are discussed in section 4. Finally, section 5 provides the overall conclusion of this research paper.

2. Related works

Chen [16] introduced the frame and spatial attention network (FSAN) which was an improvisation of residual convolutional neural network (CNN) to recognize the actions from the video frames. The two-level attention module was utilized to highlight the information of features including temporal and spatial dimensions. The two-level attention module exploits the significant features in the entire video sequence and minimizes the interference caused by similarities among the video. However, the attention mechanism used in FSAN was not robust and had a poor convergence rate.

Zhang [17] introduced an ensemble framework that consists of a convolutional neural network (CNN) and bi-directional long short term memory (Bi-LSTM) to recognize human activities. Moreover, the introduced framework utilized the swarm intelligence (SI) algorithm to detect the optimal hyper parameters and improvise the learning capability of the Bi-LSTM network. The introduced SI algorithm has a higher ability to solve the issues regarding high dimensional optimization functions. However, the introduced ensemble framework was not effective to categorize large-scale computer vision tasks.

Xiong [18] have introduced action sequence optimization and two stream fusion network to recognize human activities. The introduced method was based on short segmentation and weighted sampling which highlight the area with high activities, remove the repeated data, and help to extract the long-range information. The two stream fusion utilized 3D CNN which combines the RGB feature and human skeleton data which provides a deep presentation of human activities. However, the introduced method does not consider the background information when it is considered it may provide higher classification efficiency.

Alamuru and Jain [19] have introduced an ensemble-based algorithm that detects multiple events from the video sequences. Initially, the histogram equalization technique was used to improve the contrast and smoothen the videos obtained from CCV and UCF101 datasets. The introduced ensemble algorithm was used for feature selection which selects the optimistic features and the extracted features were fed as input to the multi support vector machine (MSVM) classifier. At last, appropriate events were recovered by measuring the Euclidean distance. However, the introduced method has increased system complexities due to its multi-event detection.

Akhter [20] introduced the gait event detection (GED) system which consists of 2D- stick model with a hierarchical optimization algorithm. The distinguishing movable features are proposed by concentrating on invariant and localized aspects of human postures in various event classes. Ultimately, to distinguish between complex postures and assign the proper labels to each occurrence, grey wolf optimization and a genetic algorithm were utilized. The introduced system is applied to real-world applications and surveillance systems. However, the dataset used in this research seems to be challenging due to its variation in the movement of cameras and the size of varied objects.

The recent optimization algorithms like guided pelican algorithm (GPA) [21] and puzzle optimization algorithm (POA) [22] were surveyed. GPA was used in the process of optimizing the networks with the help of three stages such as target selection, replacing the location and utilized multiple number of candidates to deal with the problems regarding real world optimization technqiues. The POA was introduced by Fatemeh Ahmadi Zeidabadi and Mohammad Dehghani which was based on simulation process of solving the puzzles and POA has no control parameters and it is not limited by any

Received: May 29, 2023. Revised: September 1, 2023.

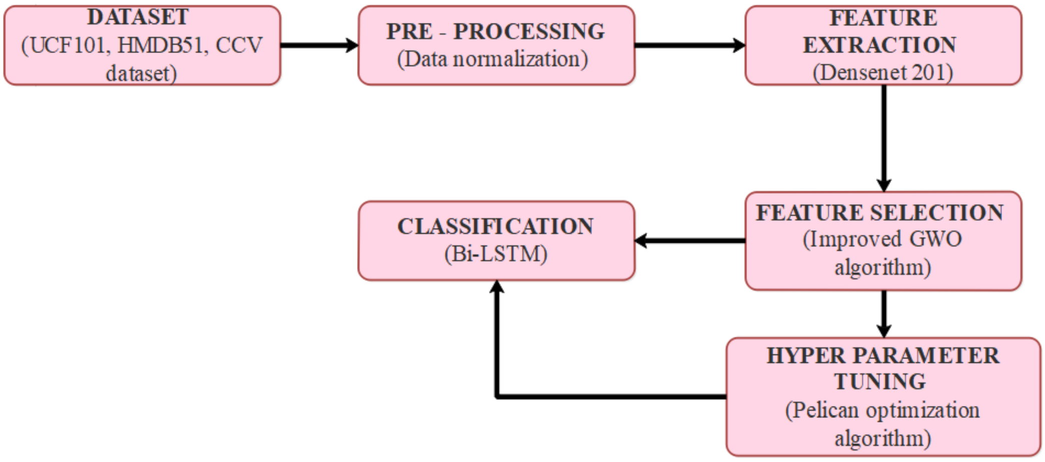

Figure. 1 The overall process involved in video event detection

parameters.

3. Methodology

Video surveillance plays an important role in detecting human activities and different types of events. Moreover, it helps in monitoring traffics, medical application, and so on. This research undergoes six stages in the process of video detection, the six stages include a collection of data from UCF101, HMDB51, and CCV datasets. The input data is pre-processed using the data normalization technique and the DenseNet 201 is used to extract the features from the pre-processed output. The hyper parameters are tuned using the pelican optimization algorithm which initializes the classification performance. Finally, the video events are classified using the Bi-LSTM classifier. The overall process involved in detection of video events is depicted in Fig. 1 as follows:

3.1 Dataset

In this research, the different type of video events is collected from three datasets namely UCF101, HMDB51, and CCV. Moreover, the samples have been taken from these datasets to evaluate the performance of the proposed method. This section provides a brief description of the aforementioned datasets.

UCF101: It is one of the largest datasets [23] which consist of human actions with 13320 video sequence with 101 types of actions such as pole vault, discus throwing, catching the Frisbee, shaving the beard, etc. In the applications regarding the detection of video events, UCF101 is considered one of the challenging datasets due to the unvaried motion of cameras, messed background and low-quality pixels. Moreover, the activities present in UCF101 relies on five classes such as the motion of the body, sports, interaction among humans, interaction among machine and humans, and playing musical instruments.

Columbia consumer video (CCV): The CCV dataset [24] consist of 9317 sequential YouTube videos with 20 class of actions. In the CCV dataset, the major count of sequential videos belongs to events and only fewer categories of action sequence which includes wedding reception, parties, trekking, swimming, soccer etc.

HMD51 dataset: The HMDB51 dataset [25] is comprised of 6849 sequences of videos with 51 classes of activities like laughing, running, and so on. Moreover, in every class, there is 101 video sequence with realistic videos belongs web video and movies.

3.2 Pre-processing

After the stage of data acquisition, pre-processing is performed to remove inappropriate information from the raw data. So, this research utilized a data normalization technique to remove the undesired image portions, discrepancies and noises present in the original database. This section describes the normalization process involved in pre-processing the input image.

3.2.1. Data normalization

Normalization is one of the techniques to pre-process the input image which eases the classification of video events. Generally, normalization is defined as the process of varying the intensities of pixels. This research utilized the min-max normalization technique to scale the input data and this scaling approach adjusts the varied values of numeric columns into a common scale. The normalization effectively eliminates unwanted data and reduces the rate of redundancy. The input image is fed into the normalization function to produce a normalized image. The min-max normalization is evaluated using the Eq. (1) represented as follows:

$$N = \frac{X - V_{min}}{V_{max} - V_{min}} \quad (1)$$

Where, V_{max} and V_{min} is denoted as the minimum and maximum intensity of video pixels correspondingly. The output obtained from normalization is fed as input for the stage of feature extraction.

3.3 Feature extraction

The features are extracted from the pre-processed output using DenseNet 201, which mines out the feature vectors using the learned weights from the datasets along with CNN. The feature concatenation is obtained from DenseNet 201 model and its mathematical representation is shown in Eq. (2) as follows:

$$X_l = H_l([X_0, X_1, \dots, X_{l-1}]) \quad (2)$$

Where the concatenation of the feature map is denoted as $[X_0, X_1, \dots, X_{l-1}]$ and $X_l(.)$ is referred to as the composite function which performs three sequential operations such as batch normalization, ReLu, and 3×3 convolution.

Any variation in the size of feature maps leads to a loss of vulnerability in the concatenation operator (H_l). The significant phase of CNN is the down-sampling layer which varies the size of feature maps and the DenseNet consists of dense blocks which are segregated by transaction layers. The transaction layer contains batch normalization with a 2×2 average pooling layer and the convolutional layer of size 1×1.

The DenseNet201 performs effectively at a smaller growth rate with the feature maps and each layer accesses the maps of preceding layers. In each layer of the feature map, k features are included as input at n^{th} layer and it is represented in Eq. (3) as follows:

$$(FM)^l = k^0 + k(l-1) \quad (3)$$

Where the layer of input channel is denoted as k^0 and the feature map is denoted as FM. The computational efficiency is enhanced by presenting a 1×1 convolutional layer in every 3×3 layer that minimizes the count of FM which seems to be higher than the k feature map. Combining the 1×1 convolutional layer in DenseNet 201 creates 5760 maps that take place in the bottleneck layer. Thus the features are extracted effectively using DenseNet 201 model and from the extracted features, the relevant features are selected using an improved GWO [26] algorithm.

3.4 Feature selection

The size of the feature vectors must be minimized to provide an effective classification, so this research used an improved IGWO algorithm which reduces the dimensionality of the feature vectors without any loss in the data. Generally, gray wolves live in packs and they have strict social predominant. The process involved in reducing the dimensionalities using the IGWO algorithm is represented in the following sections.

3.4.1. Initialization

The features are optimized using the IGWO and the solution is generated by making improvements in the algorithm. While forming the groups, the features are organized based on the execution of the diving request. The wolves with principal features are considered as best ones and every pack of wolves with better performance is distinguished by shading. The features with similar appearances are possessed in the same search space.

3.4.2. Updated feature selection

Detecting the fitness value for every individual block of the image is significant because it relies on it as a reason to make changes in classification accuracy. After the stage of fitness evaluation, the best solution is selected based on the efficacy of grey wolves. The features are arranged and become isolated based on three factors best α, best β, and best γ. Based on these three wolves, enhancement is made to select the idealistic features for the process of classification. The following conditions must be satisfied while selecting the better features which is represented in Eqs. (4) and (5) as follows:

$$A_{\{fe\}} = \left| B.fe_{prey}(t) - fe_{(wolf)}(t) \right| \quad (4)$$

$$fe(t+1) = fe_p(t) - M.G \quad (5)$$

Where the best features are represented as $B.fe$

and the features of the selected at time t are represented as $d(t)$. The co-efficient vectors are represented as M and B whose values are $2ar_1 - a$ and $B = 2r_2$. In this, the random values which lie among the range [0,1] are represented as r_1 and r_2.

3.4.3. Hunting behavior of wolves

The hunting is led by efficient wolves such as $\alpha, \beta,$ and γ with their ordinary companions. This behavior of wolves $\alpha, \beta,$ and γ is based on the following three solutions which are represented in Eqs. (6-8).

$$A_{fe}^{\alpha} = |B_{1.}fe_{\alpha} - fe| \tag{6}$$

$$A_{fe}^{\beta} = \left|B_{2.}fe_{\beta} - fe\right| \tag{7}$$

$$A_{fe}^{\gamma} = \left|B_{1.}fe_{\gamma} - fe\right| \tag{8}$$

Where the position of grey wolves $\alpha, \beta,$ and γ is represented as fe_{α}, fe_{β} and fe_{γ} respectively. It is noted that the preceding position will be irregular inside the circle and GWO considers the two parameters to be maintained equally.

3.4.4. Termination based on feature ranking

The IGWO algorithm implements more iterations to select the best fitness function. The optimal features are chosen based on the ranking method, the feature ranking helps to determine relevant features and enhance the efficiency of classification models. Once the features with optimal fitness value are obtained, then the features are provided to the classification model or it will be terminated.

3.5 Hyper-parameter optimization

The ultimate aim of hyper-parameter optimization is to enhance the video classification efficiency by improvising the hyper-parameters of the Bi-LSTM classifier. The optimization of hyper-parameters is one of the significant aspects of maintaining the model's behaviour. When the hyper-parameters are not tuned properly, then the model results in diminished results and high loss. So, the process of hyper-parameter optimization aids in better classification results. Moreover, the optimization technique is known for its significant applications in reducing time, cost, and so on. In some of the cases, it is essential to enhance the quality and efficiency, this can be attained through the technique of optimization. The traditional methods employed in optimizing the hyper-parameters do not provide effective results. Moreover, it is complex whenever the number of dimensionalities gets enhanced. So, the use of metaheuristic algorithms is considered an alternative to overcome the aforementioned issues. This research considered the following parameters and their ranges. The parameters include dropout [0.1-0.4], Learning rate [0.003-0.1], L2 Regularization [0.003-0.1], and maximum epoch which ranges from 10 epochs to 50 epochs. The metaheuristic algorithms provide optimal solutions for large datasets such as UCF101, HMDB51, and CCV datasets. The problem related to the optimization technique is mathematically expressed in Eq. (9) and the optimal value should be identified to minimize and maximize the objective function $f(x)$.

$$\frac{Maximize}{Minimize}, consider\ x \in X, f(x), x = (x_1, \ldots, x_d) \in R^d \tag{9}$$

Where the vectors of decision variables are denoted as d and the search space is denoted as X. The range of the search space is described by lower boundaries (l_i) and the upper boundaries (u_i) which is presented in Eq. (10) as follows:

$$X = \left\{x \in R^d \middle| l_i \le x_i \le u_{i,} i = 1, \ldots, d\right\} \tag{10}$$

The relationship among the objective function due to maximization and minimization is shown in Eq. (11) as follows:

$$max\, f(x) \Leftrightarrow min - 1\ \times f(x) \tag{11}$$

The efficacy of the metaheuristic algorithms maintains a balance among the phases of exploration and exploitation to obtain optimal solutions. So, this research utilized improved pelican optimization algorithm (IPOA) to optimize the hyper-parameters of feature subsets. The process of hyper-parameter optimization using IPOA is described in section 3.5.1 as follows,

3.5.1. IPOA for hyper-parameter optimization

There are many naturally inspired algorithms to select and optimize the hyper-parameters. In this research, the improved pelican optimization algorithm (IPOA) is utilized with a feature selection technique to optimize the hyperparameters. In IPOA, ω is considered as a controlling parameter that reduces the motion of pelicans at the time of hunting. Moreover, this parameter maintains a balance among the stage of exploration and exploitation of an entire pod of pelicans at the time of hunting, the outer parameter ρ diminishes the improper movement of the pelican at the time of hunting. In IPOA best solution is obtained by computing the average and standard values based on Eqs. (12) and (13) respectively.

$$average = \frac{1}{N_r} \cdot \sum_{i=1}^{N_r} BCS_i \quad (12)$$

$$standard = \sqrt{\frac{1}{N_r} \cdot \sum_{i=1}^{N_r} (BCS_i - average)^2} \quad (13)$$

Where the number of independent implementations is represented as N_r and the best solution for the candidate is denoted as BCS_i.

Every individual position of the pelican is comprised of quantitative values and the proposed method to optimize the hyper parameter is represented below steps as follows,

1. The position of the best pelican with randomly generated hyper-parameters is considered which is denoted as $C(0,5), \sigma(0,2),$ and $\varepsilon(0,1)$.
2. To select features, IPOA is utilized which is represented in the form of the p-bit binary string. The position of the best pelican is updated by a transfer sigmoid function which is represented in Eq. (14) as follows:

$$T(x) = \frac{1}{1+exp^{(-x)}} \quad (14)$$

3. Finally, the fitness value is evaluated to select the best pelican to optimize the parametric functions. The formula to evaluate the best parameter is represented in Eq. (15) as follows:

$$fitness = min\left[\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \left(y_{i,test} - \hat{y}_{i,test}\right)^2\right] \quad (15)$$

When the best pelican is not chosen to optimize the hyper parameters, the fore mentioned step 2 and 3 is continued to attain better optimization function.

3.6 Classification using Bi-LSTM

The Bi-LSTM has better capability to predict and categorize the events, and the Bi-directional time dependencies and features help to enhance the prediction accuracy when compared with unidirectional LSTM. The Bi-LSTM undergoes forward and backward operations to predict and categorize the video events. The unidirectional LSTM is comprised of memory cells which include three gates input gate, output gate and forget gate which is denoted as i_t, o_t and f_t respectively. The input information is evaluated by utilizing the input gate and it is regulated by the internal memory unit. The time data present in the internal memory unit is controlled using forget gate and the output obtained from the output gate is evaluated using the internal memory unit. The parameter x is considered as input for LSTM and the hidden state is denoted as h, these two parameters are responsible to evaluate the ability of the network's memory. The interconnection among directed graph nodes with a specified sequence is evaluated based on output obtained from the hidden state of the previous state and the input of the current moment. The principle of LSTM is based on the formulas presented in Eqs. (16-18) as follows:

$$\tilde{C}_t = tanh\,(W_c.[h_{t-1}, x_t] + b_c \quad (16)$$

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i \quad (17)$$

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f \quad (18)$$

The forget gate and the input gate which is utilized to regulate the information of cell state are represented in Eq. (19) as follows:

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (19)$$

The value for the output gate is evaluated using Eq. (20) and the hidden state is evaluated using Eq. (21) as follows:

$$O_t = \sigma((W_o.[h_{t-1}, x_t]) + b_o \quad (20)$$

$$h_t = O_t \times tanh\,(C_t) \quad (21)$$

Where the weight of four various types of matrices is represented as W_i, W_c, W_f, and W_o. The sigmoid function is denoted as σ and the outer vector product is denoted as by ×. The Bi-LSTM structure utilized forward and backward information of time series and helps to enhance the prediction and classification accuracy.

4. Results and analysis

In this research, the performance of Bi-LSTM after optimizing it with POA is evaluated using MATLAB 2020 software with system specifications such as Intel i7 processor and Windows 11 operating system. This section provides a detailed analysis to test the efficiency of the classifiers including Bi-LSTM with default hyper-parameters and optimized

Received: May 29, 2023. Revised: September 1, 2023.

Table 1. Performance evaluation of classifiers with default hyperparameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	86.99	87.80	85.65	86.15
MSVM	80.85	81.52	79.43	81.17
RF	79.53	80.13	79.66	80.89
Bi-LSTM	91.24	89.47	87.76	88.70

Table 2. Performance evaluation of classifiers based on optimized hyper-parameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	90.55	90.72	91.05	91.22
MSVM	89.86	88.69	87.16	89.73
RF	87.67	87.35	86.85	86.33
Bi-LSTM	93.91	93.45	92.14	93.70

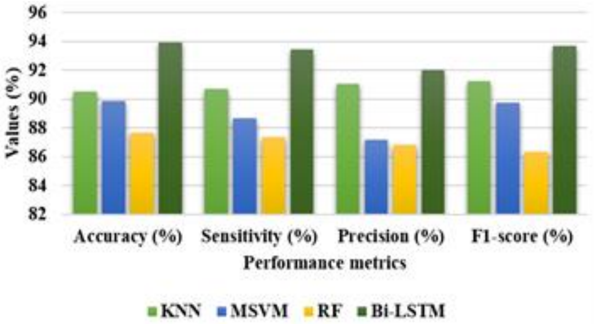

Figure. 2 Bi-LSTM performance for UCF101

hyperparameters for datasets such as UCF101, HMDB51 and CCV datasets. Here, the efficacy of the Bi-LSTM is evaluated based on performance metrics such as accuracy, precision, sensitivity and F-1 score. The accuracy and precision are considered significant performance metrics to evaluate the efficacy of the classifier in the process of detecting videos. The sensitivity is determined by the total number of positive and negative observations which is detected in a total number of observations. The value of the F-1 score is evaluated by considering sensitivity and precision. The formula to evaluate the aforementioned performance metrics is described in Eqs. (22-25) as follows:

$$Accuracy = \frac{TP+TN}{TN+TP+FN+FP} \times 100 \tag{22}$$

$$Precision = \frac{TP}{TP+FP} \times 100 \tag{23}$$

$$Sensitivity = \frac{TP}{TP+FN} \times 100 \tag{24}$$

$$F1 - score = \frac{2TP}{FP+2TP+FN} \times 100 \tag{25}$$

Where TP is a true positive, TN is represented as a true negative, FP is represented as a false positive and FN is represented as a false negative.

4.1 Quantitative analysis

In this subsection, the efficiency of classifiers including Bi-LSTM is evaluated with default hyperparameters and hyperparameters with optimization. The performance is evaluated for three datasets namely UCF101, HMDB51 and CCV based on the aforementioned evaluation metrics such as accuracy, sensitivity, precision, and recall. Table 1 describes the performance of various classifiers including Bi-LSTM with default hyperparameters and Table 2 describes the performance of classifiers based on optimized hyperparameters. The results from Tables 1 and 2 are evaluated for UCF101 dataset.

The results from Tables 1 and 2 show that the Bi-LSTM along with the proposed IPOA attained better performance in overall metrics. The Bi-LSTM has achieved a classification accuracy of 91.24% and 93.91% with default hyper-parameters and optimized hyper-parameters respectively. The better performance is due to the capability of Bi-LSTM to categorize multi-sequence videos whereas the LSTM is capable to classify a single input sequence. The graphical representation of the performance of Bi-LSTM based on optimized parameters is represented in Fig. 2.

Secondly, the performance of the Bi-LSTM with the proposed IPOA is evaluated with different classifiers such as MSVM, KNN and RF. The performance is computed based on accuracy, sensitivity, precision, and F-1 score for the HMDB51 dataset. Tables 3 and 4 represent the performance of Bi-LSTM with default hyperparameters and hyperparameters with optimization technique for the HMDB51 dataset respectively.

The results from Tables 3 and 4 show that the Bi-LSTM along with the proposed IPOA attained better performance in overall metrics. The Bi-LSTM has

Table 3. Performance evaluation of classifiers with default hyperparameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	84.56	85.80	83.56	84.51
MSVM	82.96	83.52	76.34	80.65
RF	80.81	81.13	77.87	78.76
Bi-LSTM	90.12	87.34	85.67	86.21

Table 4. Performance evaluation of classifiers based on optimized hyper-parameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	91.45	88.27	89.34	90.12
MSVM	84.86	83.45	86.78	87.37
RF	85.71	86.68	86.20	85.14
Bi-LSTM	94.19	90.45	91.04	92.74

Table 5. Performance evaluation of classifiers with default hyperparameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	83.45	83.88	84.56	81.15
MSVM	82.10	81.25	74.47	78.54
RF	80.22	79.31	75.86	77.23
Bi-LSTM	88.12	85.41	87.67	84.21

Table 6. Performance evaluation of classifiers based on optimized hyper-parameters

Classifiers	Accuracy (%)	Sensitivity (%)	Precision (%)	F1-score (%)
KNN	89.54	85.72	86.34	88.21
MSVM	87.34	81.54	85.89	84.27
RF	84.17	83.11	85.24	82.22
Bi-LSTM	90.19	88.45	89.04	91.74

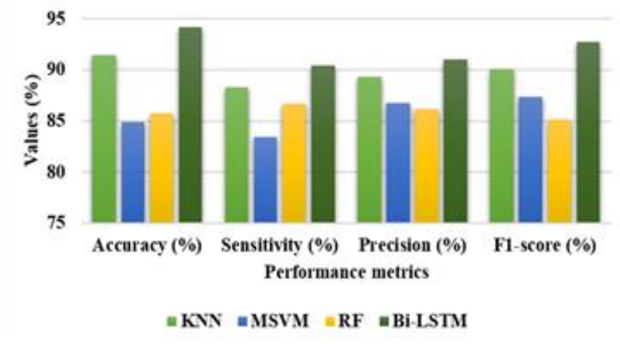

Figure. 3 Bi-LSTM performance for HMDB51

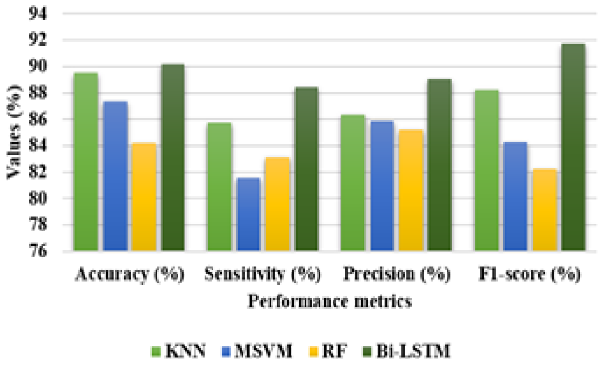

Figure. 4 Bi-LSTM performance for CCV dataset

achieved a classification accuracy of 90.12% and 94.19% with default hyperparameters and hyperparameters with optimization technique respectively. This proves that the Bi-LSTM is more efficient than KNN, MSVM and RF. The better performance is due to the capability of Bi-LSTM to categorize multi-sequence videos whereas the LSTM is capable to classify a single input sequence. The graphical representation of the performance of Bi-LSTM based on optimized hyper-parameters using IPOA for HMDB51 is represented in Fig. 3.

Finally, the performance of Bi-LSTM with the proposed IPOA for hyper-parameter optimization is related to various classifiers such as KNN, MSVM, and RF. The performance is evaluated based on accuracy, sensitivity, precision and F1 score for the CCV dataset. Tables 5 and 6 represent the overall performance of Bi-LSTM with default hyperparameters and hyperparameters with optimization for the CCV dataset respectively.

The results from Tables 5 and 6 show that the Bi-LSTM along with IPOA for hyper-parameter optimization achieved a better classification accuracy of 88.12% and 90.19% for default hyperparameters

Table 7. Performance analysis for different optimization algorithms for classification accuracy

Optimization algorithms	Dataset		
	UCF 101	HMDB51	CCV
PSO	82.34	88.11	82.86
GWO	81.97	90.88	88.12
WOA	85.54	85.09	85.05
POA	90.40	92.29	89.13
IPOA	93.91	94.19	90.19

and hyper-parameters with optimization techniques respectively. Moreover, the better result of Bi-LSTM is due to its capability in categorizing the multi-sequential videos where the LSTM considers only a single video sequence, this helps Bi-LSTM to achieve better performance than other classifiers. The graphical representation of the performance of Bi-LSTM based on optimized hyper-parameters using IPOA for the CCV dataset is represented in Fig. 4.

In Table 7, the performance of the proposed optimization algorithm is evaluated with the existing optimization techniques such as particle swarm optimization (PSO), grey wolf optimization (GWO), whale optimization algorithm (WOA) and pelican optimization algorithm (POA). The performance of the proposed and the existing optimization technqiues are evaluated by considering the classification accuracy as the common metric. The results from the Table 7 shows that IPOA have achieved better accuracy for all three datasets. For example, the classification accuracy of the IPOA for HMDB51 is 94.19% which is comparatively higher than the existing optimization technqiues such as PSO, GWO, WOA, POA with classification accuracy of 88.11%, 90.88%,85.09%, 92.29% and 94.19% respectively.

4.2 Comparative analysis

This section described the efficacy of the proposed method with existing methods utilized in the process of video event classification. The results are evaluated for three datasets such as UCF101, HMDB51 and CCV datasets. The results are evaluated by considering classification accuracy as a common performance metric. The proposed method is compared with improved residual convolutional neural network (CNN) [16], Bi-LSTM CNN [17], two-stream 3D CNN model [18] and gait event detection system [20]. Table 3 provides the comparative result of the proposed method with existing techniques based on classification accuracy.

The results from Table 8 show that the proposed Bi-LSTM with the proposed IPOA for hyperparameter optimization has attained better classification accuracy for all three datasets. The proposed method achieved a classification accuracy of 93.91%, 94.19%, and 90.19% for datasets such as UCF101, HMDB51 and CCV respectively. These experimental results are comparatively higher than the existing techniques. The better classification accuracy is due to the process of hyper-parameter optimization using IPOA and the utilized Bi-LSTM aids in recognizing multi video sequences with better accuracy.

Table 8. Comparative table

Methods	Datasets	Classification accuracy (%)
Improved residual CNN [16]	UCF101	91.68
	HMDB51	72.6
Bi-LSTM CNN [17]	UCF101	92.22
Two-stream 3D CNN model [18]	UCF101	92.56
	HMDB51	75.25
Gait Event Detection System [20]	UCF101	82.6
Bi-LSTM IPOA	UCF101	93.91
	HMDB51	94.19
	CCV	90.19

5. Conclusion

In this research study, an improved pelican optimization algorithm (IPOA) is proposed to optimize the hyper-parameters which aids in better classification accuracy using a Bi-LSTM classifier. The input data is obtained from three datasets such as UCF101, HMDB51 and CCV which consist of various types of sequential videos. The extracted output is comprised of high dimensional vectors which affect the classification performance of the whole model. So, improved GWO is utilized to extract the features and the hyper-parameters are optimized using the proposed IPOA. Due to hyper-parameter optimization, better classification accuracy is obtained using Bi-LSTM. The proposed method obtains better classification accuracy of 93.91%, 94.19%, and 90.19% for datasets such as UCF101, HMDB51 and CCV respectively. In the future, hybridization can be performed to optimize the hyper-parameters and enhance the classification accuracy.

Notation list

Parameter	Description
N	Normalized image
V_{max}	Maximum intensity of the video pixels

V_{min}	Minimum intensity of the video pixels
$[X_0, X_1, \ldots, X_{l-1}]$	Concatenation of the feature map
$X_l(.)$	Composite function
(H_l)	Concatenation operator
k^0	Layer of the input channel
FM	Feature Map
B	Best features
r_1 and r_2	Random values lies in the range [0,1]
d	Decision variables
X	Search space
l_i	Lower boundaries
u_i	Upper boundaries
N_r	Number of independent implementations
BCS_i	Best candidate solution
σ	Sigmoid function
i_t	Input gate
o_t	Output gate
f_t	Forget gate
x	Input for LSTM
$\times$	Outer vector product
h	Hidden state

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The paper conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, have been done by 1st author. The supervision and project administration, have been done by 2nd author.

References

[1] Q. Wu, A. Zhu, R. Cui, T. Wang, F. Hu, Y. Bao, and H. Snoussi, “Pose-Guided Inflated 3D ConvNet for action recognition in videos”, *Signal Processing: Image Communication*, Vol. 91, p. 116098, 2021.

[2] S. Bhaumik, P. Jana, and P. P. Mohanta, “Event and Activity Recognition in Video Surveillance for Cyber-Physical Systems”, *Emergence of Cyber Physical System and IoT in Smart Automation and Robotics: Computer Engineering in Automation, Advances in Science, Technology & Innovation*, pp. 51-68, 2021.

[3] T. Han, Y. Qi, and S. Zhu, “A Continuous Semantic Embedding Method for Video Compact Representation”, *Electronics*, Vol. 10, No. 24, p. 3106, 2021.

[4] X. Wu, Y. Park, A. Li, X. Huang, F. Xiao, and A. Usmani, “Smart detection of fire source in tunnel based on the numerical database and artificial intelligence”, *Fire Technology*, Vol. 57, No. 2, pp. 657-682, 2021.

[5] M. Ramesh and K. Mahesh, “Sports Video Classification Framework Using Enhanced Threshold Based Keyframe Selection Algorithm and Customized CNN on UCF101 and Sports1-M Dataset”, *Computational Intelligence and Neuroscience*, Vol. 2022, p. 3218431, 2022.

[6] L. Wu, Z. Yang, M. Jian, J. Shen, Y. Yang, and X. Lang, “Global motion estimation with iterative optimization-based independent univariate model for action recognition”, *Pattern Recognition*, Vol. 116, p. 107925, 2021.

[7] A. S. Brito, M. B. Vieira, S. M. Villela, H. Tacon, H. L. Chaves, H. A. Maia, D. T. Concha, and H. Pedrini, "Weighted voting of multi-stream convolutional neural networks for video-based action recognition using optical flow rhythms”, *Journal of Visual Communication and Image Representation*, Vol. 77, p. 103112, 2021.

[8] O. A. N. Rongved, M. Stige, S. A. Hicks, V. L. Thambawita, C. Midoglu, E. Zouganeli, D. Johansen, M. A. Riegler, and P. Halvorsen, “Automated event detection and classification in soccer: The potential of using multiple modalities”, *Machine Learning and Knowledge Extraction*, Vol. 3, No. 4, pp. 1030-1054, 2021.

[9] A. Hussain, T. Hussain, W. Ullah, and S. W. Baik, “Vision transformer and deep sequence learning for human activity recognition in surveillance videos”, *Computational Intelligence and Neuroscience*, Vol. 2022, p. 3454167, 2022.

[10] R. Vrskova, R. Hudec, P. Kamencay, and P. Sykora, “Human activity classification using the 3DCNN architecture”, *Applied Sciences*, Vol. 12, No. 2, p. 931, 2022.

[11] F. Serpush and M. Rezaei, “Complex human action recognition using a hierarchical feature reduction and deep learning-based method”, *SN Computer Science*, Vol. 2, p. 94, 2021.

[12] L. Zhang and X. Xiang, “Video event classification based on two-stage neural network”, *Multimedia Tools and Applications*, Vol. 79, No. 29, pp. 21471-21486, 2020.

[13] K. Kanagaraj and G. G. L. Priya, “A new 3D convolutional neural network (3D-CNN) framework for multimedia event detection”, *Signal, Image and Video Processing*, Vol. 15, No. 4, pp. 779-787, 2021.

[14] S. J. Shri, and S. Jothilakshmi, “Crowd Video Event classification using Convolutional Neural

Network", *Computer Communications*, Vol. 147, pp. 35-39, 2019.

[15] A.S. Keceli and A. Kaya, "Violent activity classification with transferred deep features and 3d-CNN", *Signal, Image and Video Processing*, Vol. 17, No. 1, pp. 139-146, 2023.

[16] B. Chen, F. Meng, H. Tang, and G. Tong, "Two-Level Attention Module Based on Spurious-3D Residual Networks for Human Action Recognition", *Sensors*, Vol. 23, No. 3, p. 1707, 2023.

[17] L. Zhang, C. P. Lim, and Y. Yu, "Intelligent human action recognition using an ensemble model of evolving deep networks with swarm-based optimization", *Knowledge-Based Systems*, Vol. 220, p. 106918, 2021.

[18] X. Xiong, W. Min, Q. Han, Q. Wang, and C. Zha, "Action Recognition Using Action Sequences Optimization and Two-Stream 3D Dilated Neural Network", *Computational Intelligence and Neuroscience*, Vol. 2022, p. 6608448, 2022.

[19] S. Alamuru and S. Jain, "Video event detection, classification and retrieval using ensemble feature selection", *Cluster Computing*, Vol. 24, No. 4, pp. 2995-3010, 2021.

[20] I. Akhter, A. Jalal, and K. Kim, "Adaptive pose estimation for gait event detection using context-aware model and hierarchical optimization", *Journal of Electrical Engineering and Technology*, Vol. 16, No. 5, pp. 2721-2729, 2021.

[21] P. D. Kusuma and A. L. Prasasti, "Guided Pelican Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 6, pp. 179-190, 2022, doi: 10.22266/ijies2022.1231.18.

[22] F. A. Zeidabadi and M. Dehghani, "POA: Puzzle Optimization Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, pp. 273-281, 2022, doi: 10.22266/ijies2022.0228.25.

[23] Link to download UCF101 dataset: https://www.crcv.ucf.edu/data/UCF101.php

[24] Link to download CCV dataset: https://www.ee.columbia.edu/ln/dvmm/CCV/

[25] Link to download HMDB51 dataset: https://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/

[26] K. Shankar, S. K. Lakshmanaprabu, A. Khanna, S. Tanwar, J. J. P. C. Rodrigues, and N. R. Roy, "Alzheimer detection using Group Grey Wolf Optimization based features with convolutional classifier", *Computers and Electrical Engineering*, Vol. 77, pp. 230-243, 2019.