Grey wolf optimization (GWO) with the convolution neural network (CNN)-based pattern recognition system

Aatif Jamshed, Bhawna Mallick and Rajendra Kumar Bharti

ABSTRACT

The dynamic video frame dataset's automated feature analysis addresses the complexity of intensity mapping with normal and abnormal classes. Iterative modelling is needed to learn the component of a video frame in several patterns for various video frame data types for threshold-based data clustering and feature analysis. GWO optimises the Convoluted Pattern of Wavelet Transform (CPWT) feature vectors employed in this paper's CNN feature analysis technique. A median filter reduces noise and smooths the video frame before normalising it. Edge information represents the video frame's bright spot boundary. Neural network based video frame classification clusters pixels using feature recurrent learning with minimal dataset training. The filtered video frame's features were evaluated using complex wavelet transformation feature extraction algorithms. These features demonstrate video frame spatial and textural classifications. CNN classifiers help analyse video frame instances and classify action labels. Categorization improves with the fewest training datasets. This strategy may be beneficial if compared to optimal practices.

Introduction

In the recent development process of the image processing system, most of the applications were focused on the various kinds of real-time video frame classification systems with enhanced prediction models of human action recognition. This improves the automation process which is to reduce the error rate in the classification system of video frame validation [1]. To classify or predict the type and human action in a video/video frame, there are several sources of instruments and devices to get the video in low-resolution and high-resolution factors. Video frames were extracted from the raw movie data, and then the acquired image was analyzed by segmenting and identifying the human activity in the video using image processing methods. The segmentation and classification process deals with the image pixel analysis based on certain rules and conditions [2]. This is to validate the pattern of the image where the region of interest (ROI) is present in the video frame. There are several methods to identify the location of ROI in the image which are based on the pattern extraction and clustering of the particles that are related to the required object [3]. In that, the correlation factor among the current frame and predefined features were classified to get the ROI.

There are several applications in video processing to analyze the source of video such as moving object tracking, vehicle tracking and recognition, gait recognition, etc. that are based on the image geometrical structure [4] and pattern-based image analysis method to classify the required object with high accuracy compared to other traditional feature analysis methods. The clustering model for ROI segmentation refers to the relevant pixel intensity value for the overall image matrix and also considers the histogram peaks of the image [5]. The segmentation and classification are integrated to form a better image analysis model. For enhanced classification performance, the Convolutional Neural Network (CNN) [6] can extract the features based on the block separation.

In the traditional model of image classification, the feature vector need to extract from the image matrix by using some feature extraction methods. In CNN, the convolution pattern of an image was considered as the feature vector and can be classified based on the neuron network formation at the training stage of the CNN classifier [7]. This
type of classification process doesn’t require any type of feature extraction method to represent the image feature set. This CNN model evaluates the pattern extraction model internally for the block-separated image.

From the previous discussion, the image pattern extraction method enhanced the recognition rate of an image processing application. In that, the CNN was also considered as the pattern-based image recognition system to recognize the image based on the convolution pattern of the image [8]. Instead of using raw image data as input, the performance of CNN was enhanced by using the texture pattern of images as the input of the classifier. Local Binary Pattern (LBP), Local Tetra Pattern (LTrP), and Local Ternary Pattern (LTrP) are only a few of the texture pattern extraction techniques that have been put into practice for the image/video categorization and human action prediction process [9]. In this proposed model of the human action recognition system, a novel image texture pattern extraction method was implemented to predict the difference of neighbourhood pixels in magnitude at different angles of the projection model. This can be achieved by using the Convoluted Pattern of Wavelet Transform (CPWT) texture extraction method with the combination of the wavelet transform. In this, the wavelet transform is for reducing the dimensionality of the image feature and the convolution pattern is the result of the convoluted image with pattern analysis. These combinations enhanced the pattern retrieval model and improve the classification performance of CNN. Furthermore, the feature can be optimized by using the Grey Wolf Optimization (GWO) method [10] which selects the best feature attribute among the overall feature sets. According to the selected attributes, the neural network in the CNN was formed in the training process of that classifier. The motivation of this proposed work is to extract a better pattern extraction model for video processing and reduce its complexity. In that, the frame pattern from wavelet transform provides a better feature model. Along with that GWO optimization improves feature learning by optimally selecting the feature attributes for classification. To achieve this, the Convoluted Patterns of Wavelet Transform (CPWT) and GWO optimization are proposed. The main motivation to implement the CNN classifier is that it can recognize the features of texture patterns better than other existing classification models. The novelty of this paper is to extract the features for the frame based on the texture pattern and optimal feature learning by using the combination of CPWT with GWO-based pattern analysis. The CPWT-GWO enhances the pattern learning model for CNN and improves the classification efficiency.

The main contribution of the proposed work is

1. To equalize the image intensity of the image and normalize the pattern, the Cellular Automata-based image filtering method.
2. A novel method of image pattern extraction by using the CPWT method.
3. To classify the category of human actions in video frames by using the CNN classifier.
4. To reduce the feature size and select the best attribute by using the GWO optimization method.

The paper has been segmented into the following sections: The survey of various methods of feature extraction, feature selection and classification algorithms is presented in section II. Then the overall proposed work and its algorithm procedure are described in section III. Section IV presents the analysis of different pattern extraction methods that are compared to the proposed model of the recognition system based on the CPWT pattern extraction and GWO optimization with the CNN model. The results are justified and concluded with future enhancement in section V.

Related work

This article provides an overview of several techniques for extracting and categorizing video frame features. This review focuses on the use of texture pattern extraction models and classification techniques in endoscopic and other video frame processing applications and covers the related ground.

In [11], the author proposed a dynamic feature learning system based on the skeleton structure of video sequences. In that, the spectra of body parts were extracted and classified by using the CNN method. In [12], the paper discussed the first-person action recognition model to classify the video appearance, shape and motion of the visual rhythm. This was extracted by referring to the textual features of the video dataset. From this type of texture analysis, [13] proposed a texture fusion-based facial expression recognition model. In that, the LBP and landmark-based Active Shape Model (ASM) were fused to form the texture plus geometrical feature analysis model which is classified as the Support Vector Machine (SVM) method. In [14], the author proposed the action recognition model from RGB with the Depth video dataset. To extract the RGB and depth image/video data, the Kinect sensor-based video system was used. To classify human action, a deep learning method was implemented with extreme learning machines. Later, to enhance the prediction performance for the RGB-D image dataset, [15] proposed a multi-directional projected depth motion mapping technique that was implemented and tested for the
action recognition system. This was further improved by embedding the skeleton key points of the human body [16,17]. In [18], the texture pattern was extracted to map the magnitude and parameter representation. The depth image sequence became an important feature of the action prediction system. By considering that, [19] proposed an R-pattern transformation method combined with Zernike Moments to recognize the human abnormal action for the depth image sequence of video data. Similarly, [20] proposed a fusion-based machine learning model based on the SVM classifier and the Nave Bayes classification model to recognize the action for RGB and the depth image. In [21], the skeleton structure-based action prediction by using the hierarchical spatial feature learning combined with temporal stack learning network architecture is shown. The human action recognition was further enhanced by the segmentation approach based on the statistical weight of the image pattern in [22]. This was optimally selected by the rank correlation method.

In the image classification process, most of the applications were developed by using the convolutional neural network to improve the accuracy of prediction. According to that, [23] proposed a salient feature analysis based on the 3D CNN model with the LSTM method. This can reduce the size of the overall feature by considering the salient features of the video. In [24], the author proposed a novel model of Deep CNN-based action prediction model for still images. In this, the CNN was fused with two or three concatenated models. To enhance the feature learning of CNN, [25] proposed an efficient image pooling model of an action recognition system. This image pooling method reduces the dataset of video frames instead of analyzing the whole video sequences. Also, [26] proposed a hint-enhanced deep learning method for action recognition. In this paper, the investigation is based on the potential of CNN for image-based action recognition. This type of CNN-based action recognition system was implemented in the 3D skeleton structure of human video based on the scale-invariant mapping and the multi-scale dilated CNN technique in [27]. Similarly, [28] proposed two-stream-based action recognition using CNN. For the LSTM-based CNN, [29] proposed a quaternary spatial temporal model that was developed for the RGB video action recognition system. Compared to the traditional CNN model, the QST-CNN enhances the feature learning concept of CNN. In [30], the paper proposed the review of different techniques of the CNN model for the action recognition-based application. For the RGB-Depth image classification, the CNN extracts the textural feature to analyze the rotational invariant vector. In [31], the author proposed depth motion maps-based Local Ternary Pattern-based texture pattern extraction method to combine the spatial feature and textural feature of the image sequence. For the cloud-based application, [32] proposed a super-pixel transformation-based feature retrieval model to improve the efficiency and increase the speed of performance compared to the traditional model of CNN. In this, the human action recognition is processed by the four-stream deep convolutional neural network method. To reduce the feature size for the learning process of the CNN classifier, the image feature was optimized by using the Artificial Bee Colony (ABC) optimization algorithm [33]. This can select the best feature attributes among the overall feature sets. Similarly, in [34], the author implemented the optimal feature selection by using the combination Grey Wolf Optimization (GWO) algorithm with the CNN classifier. In [35], the author proposed a hierarchical spatial–temporal-dependent feature analysis model with CNN to enhance the classification performance of CNN with the LSTM model.

In [36], the paper surveyed different fusion techniques of a depth image with RGB to recognize human action. The multi-class SVM and other classification types of neural networks were presented for action recognition. This was then improved by using the hybrid method of a heuristic algorithm with a deep learning model in [37]. In this, the ABC optimization and the PSO are combined to form the hybrid optimization technique for feature selection. This improves the classification performance. A survey of vision-based human action recognition is presented in [38]. In this survey, the problems faced in vision capturing were focused such as camera motion, dynamic background and other impacts that failed in the previous methods of action recognition. From these analyses, [39] presented action recognition by using the LSTM method. In this, the LSTM is improved by a two-stream attention model. This improves the learning structure of the classifier better than the other traditional model. In [40], the author proposed a two-fold transformation model with the Gabor Wavelet Transform (GWT) and Ridgelet Transform (RT). In this, human action recognition uses a decisive pose.

From these reviews of existing work for the video frame feature analysis, most of the authors proposed texture pattern-based feature extraction methods and also justified better enhancement over another method to improve the classification performance. The survey collections consist of action prediction and other video frame processing applications to evaluate the performance of pattern extraction methods in various domains. From this survey, it is clear that in the pattern extraction method, the multi-angular-based pixel validation achieved better
accuracy than those methods that are referring a smaller number of boundary validation. In these methods, the spatial representation of the video frame helps to improve the recognition rate with a clear depth of pixel intensity. This form of spatial video frame representation and multi-angular pixel validation can be achieved by the proposed method of Convoluted Pattern of Wavelet Transform (CPWT) compared with Local Binary Pattern (LBP). The detailed description with algorithm steps and the result validation of the proposed method are discussed in the following sections.

Proposed work

A novel model of image pattern recognition model was proposed in this paper with the CNN classification. In this proposed feature analysis, the feature vector from the Convoluted Pattern of Wavelet Transform (CPWT) for the testing video frame in a video was optimally selected by using the Grey Wolf Optimization (GWO) algorithm. This combination performs a multi-angular pattern analysis model to improve the performance of classification. In this work, the texture pattern from the CPWT was considered as the input for CNN in both the training and testing feature sets. This type of feature pattern improves the network formation in the CNN to identify the relevancy between the pattern from training and the testing feature. In this, the GWO acts as the dimensionality reduction by referring to the histogram peaks of the image pattern.

The important key points in the proposed pattern analysis model can be listed as follows:

1. The pre-processing can be processed by using the Cellular Automata-based filtering method to reduce the noise in the image pixel and it enhanced the edge information to extract the pattern.
2. The texture pattern of the video frame can be extracted by using the convolution pattern integrated with the wavelet transform of the CPWT method.
3. The pattern features can be optimally selected as the best texture for classification by using the GWO optimization algorithm.
4. Then these texture features of a video frame can be classified by using the CNN method to enhance the prediction performance.

The overall block diagram of the proposed feature analysis model is shown in Figure 1 with line flow for the application of the video retrieval system. In that, the pattern of the image can be extracted by verifying the difference in pixel intensity between the neighbouring pixels. Here, the mask of the image can be chosen based on the number of neighbouring pixels that are selected from the mask of that image. Then from that encoded image, wavelet transform is used to get the full texture of CPWT. In that, the GWO optimization is used to select the best feature of that image by referring to the histogram peaks of that pattern. From the histogram peaks, the best region was selected to represent the feature of a video frame. This was processed for the frames in a video of the source. Finally, the average rate of classification presents the recognized result class to indicate the action of the video.

The strength of the proposed type of classification and prediction model can be listed as follows.

- Texture-based feature analysis improves the representation of the input data with the proper edge preservation method.
- The optimal feature selection and classification selects the best feature of the overall data and reduces the size of the training dataset to save it as the database.
- Texture classification enhances the CNN compared to the traditional image input for the classification model.

The functions that are included in the proposed model can be listed as follows.

(a) Pre-processing (CA),
Texture Pattern Extraction (CPWT), Feature Optimization (GWO), Texture Classification (CNN).

Pre-processing

In image processing, image noise filtering and image equalization techniques were most commonly used in the pre-processing stage. This is to enhance the pixel quality by normalizing the pixel intensity to make the equalization of the histogram in an image. Several types of traditional filtering algorithms were implemented to remove the noise by applying a smoothing effect to the image. This can be processed by using image transformation methods. Some of these methods were approaching the concept of neighbourhood pixel validation. The methods, such as median filter, mean filter and other types of statistical properties, improve the noise removal rate better than other methods. In relation to that, the Cellular Automata (CA) also refers to the neighbourhood pixels to filter the image. These types of filters used some standard size mask to apply over the image for noise identification and normalization.

Figure 2 shows the sample matrix with indexing labelled masks that are generally used in the filtering model. In that, the mask size was selected based on odd values which can be as like a $3 \times 3$ and $5 \times 5$ matrix. While choosing the mask size, the smaller size needs to consider for the better Peak Signal To Noise Ratio (PSNR) value and lower Mean Square Error (MSE). By considering this, the proposed filtering technique implements a $3 \times 3$ mask size. In that, the detected noisy pixel from the image can be normalized by applying the Laplacian transform over the separate mask of the video frame. The detailed steps that are involved in the CA filtering with Laplacian transform are stated in Algorithm 1.

Figure 2. Schematic structure of CA mask.

Algorithm 1: CA filtering

| Input: Testing Video input, $V$ |
| Output: Filtered video frame, $V_f$ |

Laplacian transforms for video frame by Equation (1).

\[
\text{for } m = 2 \text{ to } R-1 \text{ do} \\
\text{for } n = 2 \text{ to } C-1 \text{ do} \\
\text{// 'R' and 'C' are the row and column of video frame size.} \\
I_b = I_r(m-1:n+1, n-1:n+1) \text{// Separating image to cells.} \\
I_c = I_r(m, n) \text{// $I_b$ - Boundary Pixels, $I_c$ - Center of mask.} \\
\text{Estimate magnitude of the difference } I_p \text{ by Equation (3).} \\
\text{If } I_b \approx I_c, \text{ then } t = I_p \\
\text{Else } t = I_c \\
\text{End if} \\
\text{if } (m, n) = t \text{// Filtered video frame output} \\
\text{End 'm' loop} \\
\text{End 'n' loop} \\
\]

The overall block diagram of the proposed CA with the Laplacian filtering model is shown in Figure 2.

The Laplacian transform can be estimated for the video frame $I_{in}$ which is represented as $I_{M}$. This can be calculated by Equation (1). In this, the sum of the convoluted matrix from $M$ with $I_{in}$ forms the transformed result of the image matrix. In this, the mask of the matrix was represented as the random value that is referred from the Laplacian distribution.

\[
I_l = \sum (I_{in} \ast M) \quad (1)
\]

where, $M$ – Mask matrix of the Laplacian distribution-based filter.

Here the error pixel can be estimated by Equation (2). The error pixels are collected by the rule that the pixels satisfy the condition. Others are considered

Figure 3. Block diagram of CA-Laplacian.
For normal image pixels,
\[ E_{xy} = \begin{cases} \frac{P_{mn}}{\max} & \text{if}(N_{min} \geq P_{mn} \geq N_{max}) \\ 0 & \text{Otherwise} \end{cases} \quad (2) \]

where, \( P_{mn} \) – Noisy pixel; \( N \) – Number of boundaries.

**Figure 3** The predicted noisy pixel and its position can be normalized by estimating the magnitude of the difference between the neighbouring pixels. This was represented as \( l_b \) and can be evaluated by Equation (3). From this equation, the error pixels are corrected by estimating the average of pixels in the cell matrix and replacing the error pixel with this \( l_b. \) The final matrix formed as the filtered frame of the video.

\[ l_b = \frac{1}{l} \sum_{k=1}^{l-1} (l_k) \]

where \( l \) – Length of boundaries in the video frame mask.

### Texture pattern extraction (CPWT)

A detailed description of the proposed texture pattern model of CPWT is presented in this section. Here, the combination of wavelet transform and the proposed model of the CPWT algorithm represents the texture of each video frame from the pre-processed video input. Algorithm 2 describes the steps in the CPWT pattern extraction method.

**Algorithm 2: CPWT algorithm for pattern extraction**

**Input:** Filtered Video frame.

**Output:** Video frame Pattern output \( I'_f. \)

Initialize Convolution Mask, \( G_M \)

1. In this, the \( M \) – matrix is in the size of \( 5 \times 5 \) for the video frame pattern.
2. For \( m = 3 \) to \( M - 2 \) // Loop for row index 3 to M-2 size
3. For \( n = 3 \) to \( N - 2) // Loop for column index 3 to N-2 size
4. \( l_k = I_k(m - 2m + 2, n - 2n + 2) // Video frame into a 5 x 5 matrix of cells based on the \( m \) and \( n \) index.
5. The cell separated patch can be convolute by Equation (4).
6. The image gradient for the convoluted frame can be estimate by Equation (5).
7. For various projection angles \( ( -45^0, 0^0, +45^0, 90^0) \), the magnitude was estimated by Equation (6).
8. The maximum pixel progression value from the function for each pixel can be calculated by Equation (7).
9. The iteration of \( k \) value was varied to get the first and second maximum of the function as by iterating \( k = (1, 2) \).
10. Encode the CPWT particle by Equation (8),

\[ E_{xy} = \frac{P_{mn}}{\max} \text{if}(N_{min} \geq P_{mn} \geq N_{max}) \text{Otherwise} \]

\[ I'_f(m, n) = \text{CPWT}(x, y) \]

From this, the gradient of the image can be estimated by the Gabor texture extraction method with the filter coefficients that are referred from (5). \(|G_{xy}| \) and \(|\alpha(x, y)\) represent the magnitude and gradient of the convoluted image matrix, respectively. These can be calculated by using Equation (5).

\[ |G_{xy}| = \sqrt{G_x^2 + G_y^2}, \quad \alpha(x, y) = \tan^{-1} \left( \frac{G_x}{G_y} \right) \]

where \( x \) and \( y \) are the mask cell size

\[ G_x = \frac{\partial G}{\partial x} \quad \text{and} \quad G_y = \frac{\partial G}{\partial y} \]

After getting the gradient and magnitude of the video frame, this can be quantized in different projection angles \( (-45^0, 0^0, +45^0, +90^0) \). This form of normalization can be estimated by using Equation (6). The function \( f_2 \) and the magnitude of the gradient feature are arranged to form the feature matrix that is estimated at different directions of angles.

\[ l_{ux}(x, y) = \sum_{k=N1}^{N2} \sum_{r=N2}^{N2} |G_{xy}(x, y)| \times f_2(\alpha_l, \alpha_u, \alpha(x, y)) \]

where \( f_2(p, q, r) = \begin{cases} 1 & \text{if } p \leq q < r \\ 0 & \text{else} \end{cases} \]

\[ \alpha_l = \{ -45^0, 0^0, +45^0, 90^0 \}, \quad \alpha_u = \alpha_l - 45^0 \]

The maximum pixel progression for each exported pixel of the pattern can be estimated by Equation (7).

\[ \gamma_k(x, y) = \max \left( \frac{l_{ux}(x, y)}{a_l} \right) \]

Finally, the CPWT pattern can be encoded from the binary representation of magnitude difference can be calculated by using Equation (8). The binary-to-decimal conversion estimates the resultant pattern of the frames to represent the texture pattern for each frame in the video dataset.

\[ CPWT(x, y) = \sum_{i=0}^{p} 2^i \times f_3(l_{ux}(x, t), l_y(x, t), l_y(x, t)) \]

where \( l_{ux}(x, t) = C_t(x + t, y + t) \text{, } \forall t = -1:1 \)

\[ f_3(p, q, r) = \begin{cases} 1, & \text{if } p(r \& q OR p)\& p \text{ or } p \text{ else} \end{cases} \]

\[ \gamma_k = 1 \rightarrow m = 2, n = 0 \]

\[ \gamma_k = 2 \rightarrow m = 2, n = 2 \]

The pattern extraction by the three different functions was related to the window size that is used in the cell-separated filtering technique. An example of a \( 5 \times 5 \) size image mask with its indexing is shown in Figure 2(b). From the window mask, the pixel magnitude was...
estimated in different projection angles as in the directions of \(-45^\circ, 0^\circ, +45^\circ, +90^\circ\).

Finally, the wavelet transforms for the extracted pattern can be represented in (9).

$$W(n) = \sum_{k=-\infty}^{\infty} X(k)h(2n - k)$$  \hspace{1cm} (9)

The histogram of the wavelet pattern was represented as the feature vector from the CPWT texture pattern.

**Feature optimization (GWO)**

The Grey Wolf Optimization (GWO) algorithm solves the problem like the way of hunting the prey by grey wolves. Basically, there are four different categories of grey wolves Alpha (\(\alpha\)), Beta (\(\beta\)), Delta (\(\delta\)) and Omega (\(\omega\)). In that, the alpha type is considered as the head of that group, the beta type acts as the decision maker, the delta type acts as the food provider and the omega is act as hunting the prey. Likewise, the GWO arranges the searching and optimization structure to select the best attributes among the overall feature set. The overall concept of GWO is referred from [24].

At first, the wolf particles are initialized with the size of feature attributes which can be represented as \(\chi\).

\(\chi = \{1, 2, \ldots, n\} \)

Then, the coefficient vectors \(\alpha'\), \(\beta'\), and \(\delta'\) were initialized as the random value for indicating the parameters of the current position.

At the startup, the fitness value of each search agent can be calculated for the \(\alpha'\), \(\beta'\), and \(\delta'\). Now the position of these search agents can be estimated by the random value of vector \(\alpha'\) and based on the velocity of particle movement.

Then this was iteratively updated in a loop until it reaches the maximum number of the iteration count. The search agents update the position and find the prey near that location and whether it is near the radius of searching by updating the position and velocity for each iteration. Once, if they found the prey within that radius limit, the omega group starts to hunt the best one and this will be considered the best solution. Then it searches for the nearest prey within the radius by estimating the fitness value.

From the paperwork, the best features can be selected based on validating the \(\alpha'\), \(\beta'\), and \(\delta'\) by updating the position and velocity at each iteration to find the best selection of attributes from histogram peaks.

**Texture classification (CNN)**

The CNN classifier was introduced to reduce the classification error rate due to different types of feature extraction techniques that are less robust than the CNN method. The CNN and its detailed algorithm procedure were described in [25]. In this, at the training stage of the classifier, the layers are to form the network between neurons that are based on the relevancy features in the dataset. During the testing case, this estimates the distance between the training set and testing feature vector to retrieve the class label of the testing video frame.

In the proposed work, the optimized feature was selected by the GWO and these are classified by estimating the probability of the hypothesis between training and testing data. Three major layers in the CNN can be listed as follows:

1. Convolutional layer,
2. Max-Pooling layer, and
3. Fully-Connected layer.

In the training stage of CNN, these layers were mainly involved in the two major categories of propagations, such as

1. Forward propagation and
2. Backward propagation.

In these two propagations, one is for the evaluation of feature relevancy and the other is for validating the distance whether it is within the range or not. In that, the forward propagation contains the mathematical model to represent the network arrangement based on the feature matrix in the input and hidden layer of NN. In this, 5 layers are used in the CNN to train the features and predict the relevancy among features. This was considered for the training case of the dataset for the input patterns of the video frame. Then during the testing process, the feature vector was validated from the convoluted pattern and predicted the class by identifying the minimum distance between the training feature set and the testing vector. The fully-connected layer is present at the output stage of CNN that represents the classified result of it.

The convolution of video frame \(H\) can be evaluated by Equation (10). \('f'\) is to extract the higher coefficient value of the frame matrix.

$$H = X * f$$  \hspace{1cm} (10)

where \(X\) – Input video frame, \('f'\) – Filter mask for convolution.

In the CNN, the convoluted video frames were rearranged into several pre-defined blocks to make them into small patches. The size of patch separation is depending on the feature-length that is to be classified. From these split blocks of the video frame, the feature vector can be extracted by estimating the maximum value of pixels in each block which are collectively arranged as the feature vector. This is pooled into the stream of data and passed into the input layer of the neural network stage.

In this, the selection of forward and backward propagation can be selected by referring to the
layers that are propagating through it. From that, the results from CNN are arranged in the fully connected layer. In this, the convolution of the video frame from CNN can be defined as in Equation (11).

\[ y_{ij} = \sigma(x_{ij}) \]  

(11)

where \( y_{ij} \) – Convolutional layer output by non-linearity of data, \( \sigma(x_{ij}) \) – Convolution function.

The pre-nonlinearity of the data that can be computed for \( (x_{ij}) \) can be written as

\[ x_{ij} = \sum_{a=0}^{m-1} \sum_{b=0}^{n-1} \alpha_{ab} y_{i+a}(j+b) \]  

(12)

where \( \alpha \) – Filter matrix.

The gradient error function \( \frac{\partial E}{\partial y_{ij}} \) for each neuron output represents the convolution layer of backpropagation. This can be represented in Equation (13).

\[ \frac{\partial E}{\partial \alpha_{ab}} = \sum_{i=0}^{N-m} \sum_{j=0}^{N-m} \frac{\partial E}{\partial y_{ij}} y_{i+a}(j+b) \]  

(13)

The basic block diagram of the CNN with both forward and backward propagation arrangements is shown in Figure 4. The architecture diagram of the CNN model is shown in Figure 5.

In this, the pooling layer is to evaluate the maximum value from each segmented block of the video frame to form a pre-feature matrix representation. This matrix is arranged as the sequential data to represent the feature vector for the input of a convoluted video frame. This feature vector was processed in the fully-connected layer. This layer acts as the traditional process of neural network connectivity formation.

From the output layer of the neural network, the classified result of the video frame was presented as the index of matched vector which is related to the action of humans from the video data. The applicability domain of the proposed classification model can be represented in the net file that is stored separately in the training process of the classifier. This can be updated when there was any change in the video dataset and update in the feature attributes. This training model was loaded during the testing process that is to compare and predict the relevant data. Since the simulation of this process was developed in the MATLAB tool environment, the training model was saved as the ‘mat’ file extension.

Result analysis

The result analysis section evaluates the performance of the proposed model of action recognition for the input video frame by comparing the statistical parameters of the proposed work with other state-of-the-art methods. For this, the UCF Sports video dataset and HMDB51 dataset are used to implement the proposed algorithm and test the result by finding the maximum key points of the image pattern. This proposed work is implemented and tested in the PYTHON version of V3.6 with the required libraries. The validation of performance measures is calculated by finding the error difference between the predicted result and the ground truth of the dataset. In this dataset, there are 51 discrete actions of classes are there. Among that, five actions are selected for the training and testing of human action from the video frames. This it contains 100 video datasets that are used for analysis. Of that, 70% are considered as
training and 30% are testing. There are three methods of existing systems were referred to compare the performance of the proposed work which can be listed as:

1. Deep bidirectional long short-term memory (DBiLSTM) [41].
2. Deep Convolution Symmetric Neural Network with PCANET [42].
3. Weakly-supervised action localization (WSAL) [43].
4. Deep Convoluted Neural Network (DCNN) [44].

In that, the DBiLSTM implements the bidirectional feature representation of the frame to improve the LSTM model of the classifier. Similarly, the PCANET integrates the Eigenvectors with the PCA components to optimize the feature data of the Convolution Symmetric Neural Network. This arranges the symmetrical architecture of Neural network formation according to feature matches. Similarly, the WSAL was performed by a partially supervised classification model to cluster the classes of actions in the video dataset. Also, the proposed model of classification technique was compared with the Deep CNN which integrates the CNN with the Deep learning of the classification model.

To validate the proposed model, the results of the predicted output from the classifier are collected from the overall dataset and compared with the ground truth of the dataset. Here, the ground truth was considered as the actual data and the classified labels are considered as the predicted data. Comparing these actual and predicted data, the confusion matrix can be formed to represent the true positive and negative results and false positive and negative results of the classification result. From that, the statistical parameters are calculated to represent the performance of the proposed model. Figure 6 shows the sample frames for the testing dataset. Figure 7 shows the Salient feature for weight lifting video frames from the CNN layer. The results can be compared with the existing methods by the statistical parameters, such as Sensitivity, Specificity, Precision, Recall, Jaccard, Dice Overlap, F1-Score, Matthews correlation coefficient (MCC), Error rate, Kappa Coefficient and Accuracy which are all represented in Equations (14)–(24).

\[
\text{Sensitivity} = \frac{TP}{TP + FN} \quad (14)
\]
\[
\text{Specificity} = \frac{TN}{TN + FP} \quad (15)
\]
\[
\text{Jaccard Similarity} = \frac{TP}{TP + FN + FP} \quad (16)
\]
\[
\text{Dice Overlap} = \frac{2TP}{FP + 2TP + FN} \quad (17)
\]
\[
\text{Precision} = \frac{TP}{TP + FP} \quad (18)
\]
\[
\text{Recall} = \frac{TP}{TP + FN} \quad (19)
\]
\[
\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (20)
\]
\[
\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (21)
\]
\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (22)
\]
\[
\text{Error Rate} = 1 - \text{Accuracy} \quad (23)
\]
\[
\text{Kappa Coeff} = \frac{P_o - P_e}{1 - P_e} \quad (24)
\]

where TP, FP, TN, and FN define the True Positive, False Positive, True Negative, and False Negative, respectively. In the kappa coefficient, the relative observed agreement $P_o$ and
The hypothetical probability of chance agreement $P_o$ can be calculated by Equations (25) and (26).

$$P_o = \frac{TP + TN}{TP + TN + FP + FN}$$ (25)

$$P_e = \frac{(TP + FP) \times (TP + FN) + (TN + FP) \times (TN + FN)}{(TP + TN + FP + FN)^2}$$ (26)

Figures 8 and 9 show the performance measure and the similarity rate for the clustering and classification of human action that compares the proposed work with the existing classification method of the LSTM learning method. These graphs show the bar plot for Sensitivity, Specificity, Positive Predictive Value (PPV) and Accuracy by comparing the recognized result with ground truth of the dataset. These graphs represent that the performance level of the proposed algorithm was increased around $\sim98\%$. This shows that the error rate of the proposed model is in the value of $\sim1\%$ compared to other existing classification methods.

Figure 10 shows the comparison result of the Accuracy and Kappa Coefficient for the ensemble pattern extraction method with the existing method of [41] tested in the UCF sports video dataset. Figure 11 shows the Receiver Operating Curve (ROC) comparison graph referred from [42]. The ROC curve is calculated from the confusion matrix and it is plotted for the False Positive Rate (FPR) and True Positive Rate (TPR). The FPR can be calculated by subtracting the Specificity from 1 and the True Positive Rate indicates the Sensitivity. This ROC reaches high sensitivity at minimum FPR which shows that the false rate of proposed work is around $\sim0.2$. The failure case in the proposed model can be represented in terms of parameters of the error rate and the False predictive rate. The number of failures or misclassification samples is collected to find the ratio of the overall samples to the dataset to estimate the accuracy range of the proposed classification model.

Figure 7. Sample video frames of the UCF sports dataset: (a) Video frames, (b) Salient features.

Figure 8. Performance measures for UCF dataset.

Figure 9. A similarity measure for UCF dataset.

Figure 10. Accuracy chart for UCF dataset.
Figure 12 shows the confusion matrix of the classified action for the overall video dataset. The confusion matrix is used to estimate the accuracy of the proposed work and to calculate the other parameters. In this, the diagonal of the matrix represents the correctly classified result and the remaining are the misclassified results.

Tables 1–4 represent the comparison table of proposed work with existing systems of [41] for the parameter of Dice Similarity, PPV, sensitivity and accuracy, respectively. Also, Table 5 shows the comparison result of Quantitative results for different architectures implemented in the UCF sports video dataset. The comparison for the proposed work is with the WSAL method in [43]. This shows that the proposed model achieved a better recognition rate due to the texture pattern analysis from video input.

The Dice similarity and sensitivity measurement for the methods of [A] Deep Neural Network, [B] CNN, [C] CNN with fully connected CRF, [D] WRN-PPNet, and [E] the proposed method of classification is shown in Figure 13.
From the overall results, the performance of the proposed work is justified with different methods of classification and feature representation. Also, the complexity of the proposed work is discussed in the result discussion section.

Table 6–8 show the comparison of the proposed work with several existing models of CNN based on the performance measures such as Accuracy (%), Sensitivity and Specificity in (%) and Testing time (Sec). This was also represented in Figure 14–16. This was tested and analyzed for the HMDB51 dataset. In these table results, the accuracy of the proposed work is improved by approximately 6.4% better than other CNN models in [44]. Also, the time taken for the testing gets reduced which represents the reduction of time complexity in the proposed model of GWO with CNN classification.

Table 6. Comparison table of accuracy (%) for the HMDB51 dataset.

| Methods      | Accuracy (%) | Error rate (%) |
|--------------|--------------|----------------|
| Proposed     | 89.7         | 10.3           |
| ELM          | 81.4         | 18.6           |
| Softmax      | 74.8         | 25.2           |
| MSVM         | 65.9         | 34.1           |
| KNN          | 66.3         | 33.9           |
| Ensemble Tree| 63.4         | 36.6           |

Table 7. Comparison table of sensitivity (%) and specificity (%) for the HMDB51 dataset.

| Methods      | Sensitivity (%) | Specificity (%) |
|--------------|-----------------|-----------------|
| Proposed     | 88.71           | 89.75           |
| ELM          | 80.82           | 82.14           |
| Softmax      | 74.19           | 74.83           |
| MSVM         | 65.77           | 66.62           |
| KNN          | 65.55           | 66.2            |
| Ensemble Tree| 62.91           | 63.93           |

Table 8. Comparison table of training time and testing time (Sec) for the HMDB51 dataset.

| Methods      | Training time (sec) | Testing time (sec) |
|--------------|---------------------|--------------------|
| Proposed     | 411.243             | 176.247            |
| ELM          | 493.514             | 211.506            |
| Softmax      | 562.557             | 241.096            |
| MSVM         | 920.694             | 394.583            |
| KNN          | 725.704             | 311.016            |
| Ensemble Tree| 950.367             | 407.300            |

Discussion

This section discusses the reports of proposed work results for the application of action prediction for the video source input. The performance of the optimization process is depending on the complexity of the system. The complexity indicates the time and space complexity of the algorithm that is consumed by the GWO method. The time complexity is depending on the number of iterations required to solve the problem and the time taken for one iteration.

In this model, the space complexity can be represented by the amount of feature space that is used to save as the feature database. In general, the CNN requires the raw image to classify the image type. In
the proposed work, instead of taking the raw image, the dimensionality-reduced texture pattern was used for the classification step. By considering this, the space complexity is reduced compared to the traditional model of CNN classification. For the proposed model, the time complexity can be estimated and represented in the form of notation $O\left( q \times \frac{\ln(p)}{h} \right)$. ‘$q$’ indicates the total number of cycles taken for searching for the best solution and the logarithmic of time ‘$p$’ for predicting the nearest match with the length of feature attributes ‘$h$’. From the overall work result, it reduces the complexity of the model compared to other classification methods.

The result analysis justifies that the proposed model of texture pattern classification system achieved ~98% of recognition rate compared to the other traditional pattern analysis method. The limitation that is in the proposed work is that it consumes time for the texture pattern extraction due to multi-directionality estimation for the frame matrix. This increases time consumption compared to other texture pattern extraction techniques.

**Conclusion and future enhancement**

This paper mainly focused on the identification of object movement in a video frame and to classify its action label in that, the proposed feature extraction ensembles the spatial model of the image pattern by using the image convolution method and the pattern model of the feature extraction technique by the wavelet transform of the video frame. This type of feature extraction enhanced the pattern analysis model in the action prediction for the video dataset. The CNN also makes the classification performance in feature analysis better than the other state-of-the-art methods. This was also enhanced by using the Grey Wolf Optimization method (GWO) by optimally selecting the best attributes for neural network architecture arrangement.

In the future, this type of optimal feature selection and classification can be implemented in the other image processing application to reduce the classification time complexity and also improve the performance furthermore. For the clustering and segmentation process, the iteration count can also be reduced by the optimal selection of cluster labels which may further improve the performance rate.

**Disclosure statement**

No potential conflict of interest was reported by the author(s).

**Notes on contributors**

**Aatif Jamshed** is working as Assistant Professor (Senior Scale) at the ABES Engineering College, Ghaziabad (U.P). He is pursuing a PhD from Veer Madho Singh Bhandari, Uttarakhand Technical University Dehradun; A State Govt. University. He is a member of IEEE, IJACEE, IAENG, IACSIT, CSTA etc. He has 13 years of experience in academics. He has a specialization in progressive databases. He is the reviewer of many International Journals. He has experience in real-life projects from a leading IT company in India with proficiency in Python.

**Dr. Bhawna Mallick** is working as Dean of Academics in MIET Meerut and has served as Professor and Head of the Department at Galgotia College of Engineering and Technology. She has also served as a senior consultant at Maverick Quality Advisory Services Pvt. Ltd Ghaziabad, India, she is having 23+ years of experience. Degree(s) PhD, M. Tech and B.E (Computer Technology). She has published more than 30 Research papers in reputed International Refereed Journals/ Conferences. She has organized 2 International Conferences and many workshops & Seminars.

**Dr. Rajendra Kumar Bharti** is working as Associate Professor at Bipin Tripathi Kumaon Institute of Technology, Dwarahat, Uttarakhand University. He is a member of many reputed societies. He has a specialization in Data Compression and Network Security. He is the reviewer of many International Journals. He has an experience in real-life projects.

**ORCID**

Aatif Jamshed [http://orcid.org/0000-0003-3152-6147](http://orcid.org/0000-0003-3152-6147)

**References**

[1] Zhu F, Shao L, Xie J, et al. From handcrafted to learned representations for human action recognition: a survey. Image Vis Comput. 2016;55:42–52. doi:10.1016/j.imavis.2016.06.007.

[2] Ijina EP, Mohan CK. Human action recognition based on motion capture information using fuzzy convolution neural networks. In: Eighth International Conference on Advances in Pattern Recognition, ICAPR 2015, Kolkata, India, January 4–7, 2015. IEEE; 2015.

[3] Dawn DD, Shaikh SH. A comprehensive survey of human action recognition with spatio-temporal interest point (STIP) detector. Vis Comput. 2016;32(3):289–306. doi:10.1007/s00371-015-1066-2.

[4] Vemulapalli R, Arrate F, Chellappa R. R3DG features: relative 3D geometry-based skeletal representations for human action recognition. Comput Vis Image Underst. 2016;152:155–166. doi:10.1016/j.cviu.2016.04.005.

[5] Qazi HA, et al. Human action recognition using SIFT and HOG method. In: International Conference on Information and Communication Technologies (ICICT), Karachi, Pakistan. IEEE; 2017.

[6] Ma M, Marturi N, Li Y, et al. Region-sequence based six-stream CNN features for general and fine-grained human action recognition in videos. Pattern Recognit. 2018;76:506–521. doi:10.1016/j.patcog.2017.11.026.

[7] Tu Z, Xie W, Qin Q, et al. Multi-stream CNN: learning representations based on human-related regions for action recognition. Pattern Recognit. 2018;79:32–43. doi:10.1016/j.patcog.2018.01.020.

[8] Bulbul MF, et al. Improving human action recognition using hierarchical features and multiple classifier ensembles. Comput J. 2019;64(1):1633–1655.
[9] Murala S, Jonathan Wu QM. Spherical symmetric 3D local ternary patterns for natural, texture and biomedical image indexing and retrieval. Neurocomputing. 2015;149:1502–1514. doi:10.1016/j.neucom.2014.08.042.

[10] Sam BB, Fred AL. An efficient grey wolf optimization algorithm based extended kalman filtering technique for various image modalities restoration process. Multimed Tools App. 2018;77(23):30205–30232. doi:10.1007/s11042-018-6088-0.

[11] Hou Y, Li Z, Wang P, et al. Skeleton optical spectra-based action recognition using convolutional neural networks. IEEE Trans Circuits Syst Video Technol. 2016;28(3):807–811.

[12] Moreira TP, Menotti D, Pedrini H. First-person action recognition through visual rhythm texture description. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, March 5–9, 2017. IEEE; 2017.

[13] Kumar N, Bhargava D. A scheme of features fusion for facial expression analysis: a facial action recognition. J Stat Manag Syst. 2017;20(4):693–701. doi:10.1080/09720510.2017.1395189.

[14] Ijjina EP, Chalavadi KM. Human action recognition in RGB-D videos using motion sequence information and deep learning. Pattern Recognit. 2017;72:504–516. doi:10.1016/j.patcog.2017.07.013.

[15] Satyamurthi S, Tian J, Chua MCH. Action recognition using multi-directional projected depth motion maps. J Ambient Intell Humaniz Comput. 2018;1:1–7.

[16] Ji X, Cheng J, Feng W, et al. Skeleton embedded motion body partition for human action recognition using depth sequences. Signal Processing. 2018;143:56–68. doi:10.1016/j.sigpro.2017.08.016.

[17] Wang H, Wang L. Beyond joints: learning representations from primitive geometries for skeleton-based action recognition and detection. IEEE Trans Image Process. 2018;27(9):4382–4394. doi:10.1109/TIP.2018.2837386.

[18] Arivazhagan S, Sheibah RN, Harini R, et al. Human action recognition from RGB-D data using complete local binary pattern. Cogn Syst Res. 2019;58:94–104. doi:10.1016/j.cogsys.2019.05.002.

[19] Dhiman C, Vishwakarma DK. A robust framework for abnormal human action recognition using R-transform and ternize moments in depth videos. IEEE Sensors J. 2019;19(13):5195–5203. doi:10.1109/JSEN.2019.2903645.

[20] Avola D, Bernardi M, Foresti GL. Fusing depth and colour information for human action recognition. Multimed Tools App. 2019;78(5):5919–5939. doi:10.1007/s11042-018-6875-7.

[21] Si C, Jing Y, Wang W, et al. Skeleton-based action recognition with hierarchical spatial reasoning and temporal stack learning network. Pattern Recognit. 2020;107:107511. doi:10.1016/j.patcog.2020.107511.

[22] Sharif M, Khan MA, Zahid F, et al. Human action recognition: a framework of statistical weighted segmentation and rank correlation-based selection. Pattern Anal App. 2020;23(1):281–294. doi:10.1007/s10400-019-00789-0.

[23] Wang X, Gao L, Song J, et al. Beyond frame-level CNN: saliency-aware 3-D CNN with LSTM for video action recognition. IEEE Signal Process Lett. 2016;24(4):510–514. doi:10.1109/LSP.2016.2611485.

[24] Lavinia Y, Vo HH, Verma A. Fusion based deep CNN for improved large-scale image action recognition. In: IEEE International Symposium on Multimedia (ISM), San Jose, CA, USA. IEEE; 2016.

[25] Banerjee B, Murino V. Efficient pooling of image based CNN features for action recognition in videos. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA. IEEE; 2017.

[26] Qi T, Xu Y, Quan Y, et al. Image-based action recognition using hint-enhanced deep neural networks. Neurocomputing. 2017;267:475–488. doi:10.1016/j.neucom.2017.06.041.

[27] Li B, He M, Dai Y, et al. 3D skeleton based action recognition by video-domain translation-scale invariant mapping and multi-scale dilated CNN. Multimed Tools App. 2018;77(17):22901–22921. doi:10.1007/s11042-018-5642-0.

[28] Zhang B, Wang L, Wang Z, et al. Real-time action recognition with deeply transferred motion vector cnns. IEEE Trans Image Process. 2018;27(5):2326–2339. doi:10.1109/TIP.2018.2791180.

[29] Meng B, Liu X, Wang X. Human action recognition based on quaternion spatial-temporal convolutional neural network and LSTM in RGB videos. Multimed Tools App. 2018;77(20):26901–26918. doi:10.1007/s11042-018-5893-9.

[30] Yao G, Lei T, Zhong J. A review of convolutional-neural-network-based action recognition. Pattern Recognit Lett. 2019;118:14–22. doi:10.1016/j.patrec.2018.05.018.

[31] Li Z, Zheng Z, Lin F, et al. Action recognition from depth sequence using depth motion maps-based local ternary patterns and CNN. Multimed Tools App. 2019;78(14):19587–19601. doi:10.1007/s11042-019-7356-3.

[32] Rajput AS, Raman B, Imran J. Privacy-preserving human action recognition as a remote cloud service using RGB-D sensors and deep CNN. Expert Syst Appl. 2020;152:113349.

[33] Ozcan T, Basturk A. Performance improvement of pre-trained convolutional neural networks for action recognition. Comput J. 2021;64(11):1715–1730.

[34] Kumaran N, Vadivel A, Saravana Kumar S. Recognition of human actions using CNN-GWO: a novel modeling of CNN for enhancement of classification performance. Multimed Tools App. 2019;78(14):19587–19601. doi:10.1007/s11042-017-5591-z.

[35] Zhu A, Wu Q, Cui R, et al. Exploring a rich spatial–temporal dependent relational model for skeleton-based action recognition by bidirectional LSTM-CNN. Neurocomputing. 2020;414:90–100. doi:10.1016/j.neucom.2020.07.068.

[36] Chen C, Safi R, Kehatarnavaz N. A survey of depth and inertial sensor fusion for human action recognition. Multimed Tools App. 2017;76(3):4405–4425. doi:10.1007/s11042-015-3177-1.

[37] Ozcan T, Basturk A. Human action recognition with deep learning and structural optimization using a hybrid heuristic algorithm. Cluster Comput. 2020;23(4):2847–2860.

[38] Jegham I, Ben Khalifa A, Alouani I, et al. Vision-based human action recognition: an overview and real world challenges. Forensic Sci Int: Digital Invest. 2020;32:200901. doi:10.1016/j.fsidi.2020.200901.

[39] Dai C, Liu X, Lai J. Human action recognition using two-stream attention based LSTM networks. Appl Soft Comput. 2020;86:105820. doi:10.1016/j.asoc.2019.105820.
[40] Vishwakarma DK. A two-fold transformation model for human action recognition using decisive pose. Cogn Syst Res. 2020;61:1–13. doi:10.1016/j.cogsys.2019.12.004.

[41] Sahoo SP, et al. HAR-depth: a novel framework for human action recognition using sequential learning and depth estimated history images. IEEE Trans Emerg Topics Comput Intell. 2020;5(5):813–825.

[42] Abdelbaky A, Aly S. Two-stream spatiotemporal feature fusion for human action recognition. Vis Comput. 2021;37(7):1821–1835.

[43] Ullah S, et al. Weakly-supervised action localization based on seed superpixels. Multimed Tools Appl. 2020;80(4):6203–6220.

[44] Khan MA, et al. A resource conscious human action recognition framework using 26-layered deep convolutional neural network. Multimed Tools Appl. 2020;80(28):35827–35849.