Multimedia Tools and Applications
https://doi.org/10.1007/s11042-020-08806-9

Human action recognition using fusion of multiview and deep features: an application to video surveillance

Muhammad Attique Khan[1] · Kashif Javed[2] · Sajid Ali Khan[3] · Tanzila Saba[4] · Usman Habib[5] · Junaid Ali Khan[1] · Aaqif Afzaal Abbasi[3]

Received: 10 October 2019 / Revised: 27 January 2020 / Accepted: 28 February 2020
Published online: 14 March 2020

Abstract
Human Action Recognition (HAR) has become one of the most active research area in the domain of artificial intelligence, due to various applications such as video surveillance. The wide range of variations among human actions in daily life makes the recognition process more difficult. In this article, a new fully automated scheme is proposed for Human action recognition by fusion of deep neural network (DNN) and multiview features. The DNN features are initially extracted by employing a pre-trained CNN model name VGG19. Subsequently, multiview features are computed from horizontal and vertical gradients, along with vertical directional features. Afterwards, all features are combined in order to select the best features. The best features are selected by employing three parameters i.e. relative entropy, mutual information, and strong correlation coefficient (SCC). Furthermore, these parameters are used for selection of best subset of features through a higher probability based threshold function. The final selected features are provided to Naive Bayes classifier for final recognition. The proposed scheme is tested on five datasets name HMDB51, UCF Sports, YouTube, IXMAS, and KTH and the achieved accuracy were 93.7%, 98%, 99.4%, 95.2%, and 97%, respectively. Lastly, the proposed method in this article is compared with existing techniques. The resuls shows that the proposed scheme outperforms the state of the art methods.

Keywords Human action recognition · Multiview features · Deep features · Features fusion · Recognition

✉ Sajid Ali Khan
sajidalibn@gmail.com

Extended author information available on the last page of the article

1 Introduction

Since last few dacades, HAR has become one of the most active research field in the area of pattern recognition and computer vision, due to its significant role in video understanding. HAR is involved in various real-world applications such as video surveillance [4], human-computer interaction, patient monitoring system, pedestrian detection [28], and robotics, etc. [5]. Action recognition is the method of assigning action labels such as jumping, playing, punching, walking and running, etc. in video frames, that enables the system to efficiently and automatically recognize various actions performed by a human [43]. In recent years, many HAR techniques are applied that include wearable sensor-based, video-based and wireless sensor network-based HAR etc. [21, 26, 45]. But video-based HAR methods gain attention because of its high recognition rate and easy organization. Moreover, it is extensively used in various industrial applications [9, 44].

The correct recognition of human action is still a challenging task In videos, because of various reasons such as having many inter and intraclass variations, environmental, lightning and angle variations [34], etc. To handle these challenges hand-crafted feature extraction methods such as histogram of oriented gradients (HOG) and histrogram of optical flow (HOF) are used in previous studies with images taken from 2D cameras [30]. In these methods, human actions are generally recognized on the basis of the outlook and motion of human body parts in video frames. These mechanisms lack the process of action recognition using 2D video data, due to absence of a 3D structure from the action sequences. Hence, HAR on RGB/RGB-D video frames using one modality is not adequate to handle challengues of solving real world scenario's [11]. Due to rapid advancement in recent years, the development of in-depth sensing camera technologies greatly helped in solving these challenges. They overcome the challenges by giving a detailed information about the 3D structure of moving part of human body and changes in posture positions [20, 39]. The position of a joint angle can be utilized to compute the distance among all joints, hence forms a distance-vector of every video frame.

The conventional way of human examination, segmentation and action classification begins with the extraction of human silhouettes from noisy and shadowy background areas, tracking the motion of human body parts insight, thus recognizes the action performed by the subject [8]. In order to extract features and classification of video sequences, traditional handcrafted features are applied. It can be observed from work of last few days that the use of convolutional neural network (CNN) improves the performance of recognition results. In computer vision community, CNN gain much attention due to improved performance for many applications like surveillance [12], medical, object classification [22], agriculture, biometrics [25], and few more [19]. Through CNN, it is easy to handle large data along with the best accuracy [22]. Many CNN based methods are presented in literature which follows the concept of transfer learning instead of trained a new model from scratch. That relies on the pre-trained CNN models like VGG [31], AlexNet [14], ResNet [6], and few more [7, 33]. But in many complex scenarios in HAR, only CNN features are not well performed; therefore it is essential to add few handcrafted features along with them for better features representation. The choice of pre-trained model is a key step because, using this, a well-known CNN features are required for prediction. Therefore, the choice of model selection is always dependent

on its prior performance. So major aim of this study is to develop a new automated system which can be implemented in the real time environment with minimum computational time. Furthermore, we follows few hot applications of computer vision in which best features are selected through latest techniques [29].

2 Related work

Many action recognition studies are introduced in the literature which have focused on classical and CNN features [3, 10, 27]. In this era, CNN gives an attractive performance in the area of machine learning. The most recent, Ullah et al. [35] introduced an optimized and efficient CNN based deep auto-encoder (DAE) method for Human Action Recognition (HAR). In the proposed method, input data is taken from real-world non-static surveillance surroundings, and a pre-trained CNN model is applied to extract deep features. To observe the action variations in surveillance sequences optimized differential algebraic equtions (DAE) is used, furthermore, quadratic SVM is applied for human action classification. In the final step parameters of the training model is updated by adding the iterative fine-tuning model. The introduced method is experimented using publically available datasets such as HMDB51, YouTube action dataset, UCF50, and UCF101 dataset. The achieved result show that the introduced method works efficiently in terms of time and computational cost in contrast to existing methods. Yang et al. [40] presented an asymmetric unidirectional deep 3D-CNN method to recognize human actions. In this approach, micro nets are applied to enhance the feature learning ability of an asymmetric 3D convolution. In the pre-processing phase, multi-source enhanced input is introduced that fuses required features from RGB and flow frame. The proposed architecture is tested on UCF-101 and HMDB-51 human action dataset. The achieved result show improvement in the performance as compared to existing 3D-CNN methods. Wang et al. [36] presented a temporal pooling based method to aggregate the frame-level features for human action recognition. The temporal convolutional procedure is introduced for frame-level representation for the extraction of dynamic information and also maintains the submissive value of model parameters. The presented method experimented on three datasets i.e. UCF101, Hollywood2 and HMDB51 to show the effectiveness of the method in contrast to existing methods. Wu et al. [38] presented a deep learning-based MPCANet method for human action classification. In this method, tensor interaction is included to enhance the recognition rate. It consists of projection dictionaries, projection encoder layer and pooling layer. The introduced method is experimented on UCF11, medical image dataset, and UCF sports action datasets to show the efficacy of the approach. Liu et al. [17] presented a hierarchical clustering multi-task learning (HC-MTL) approach, for joint human action grouping and recognition. In this approach, the objective function is formulated into two underlying variables for joint optimization i.e. grouping data and model parameters. Moreover, it is then divided into two sub-tasks, that is multi-task learning and task relatedness detection. Hence the presented approach can achieve optimal action model and group detection by changing it iteratively. The presented approach is tested using six real action based datasets, two constrained datasets and two multiview datasets. The achieved result shows the effectiveness of presented approach in contrast to existing methods. Zare et al. [41] presented CNN based video spatiotemporal map (VSTM) for HAR. It is a dense illustration of

video clips that combines its spatial and temporal features. Temporal features from VSTM are extracted by CNN. In this approach features are extracted using two steps. In the first step, three scale wavelet transform is employed to each frame and VSTM is created by vertical concatenation of each feature vector of the frame, and it shows the temporal estimation of video. Consequently, in second step, the convolutional layer of the CNN model extracts temporal features of VSTM and the outcome of the last layer is directed towards the FC layer. Finally the softmax layer classifies the type of human action. The presented approach is verified by using publically available datasets that include Weizmann, KTH, and UCF sports dataset. The achieved results show the efficiency of the introduced method in contrast to previous methods.

The above listed techniques work through different type of features such as CNN, motion, and handcrafted. In the CNN, FC layer-based features are computed, and directly provided to softmax classifier for recognition. On the other hand, motion features are computed from video frames and classified on the basis of their movement. In the handcrafted features, multiple features are combined like shape, motion, and point, which are further used for classification through supervised learning methods.

3 Challenges and contributions

Most of the recent human action recognition (HAR) techniques are focusing on the single view of human representation using deep learning. The multi-view action recognition is a challenging and difficult task, due to many different factors such as illumination affects human styles (walk, jogging, listing phones, punch, etc), and quality of selected videos. In the proposed work, action recognition is performed by the fusion of multiview and deep feature extraction. The multiview features are difficult to extract, therefore, it is required to compute information of given frames under horizontal gradient, and vertical gradient directions. Moreover, high-level features are also required to obtain better system accuracy. Therefore, in this work, deep learning and the best multiview features are combined for action recognition. Major contributions in this work are:

(i) Multiview features are extracted through gradient information of both x-axis and y-axis. Later, the features are combined along with horizontal and vertical features for better information.
(ii) Deep neural network (DNN) based high-level features are calculated. The transfer learning is performed on original pre-trained models (VGG19), by employing entropy max activation function.
(iii) Fused both multiview and DNN features through the serial-based approach.
(iv) Select the best features by employing entropy, mutual information, and strong correlation. Through these operations, three distinct vectors are obtained which are fused by mean parallel fusion. Finally, a threshold function is defined using a maximum probability value.
(v) A detailed analysis is conducted with original DNN, multiview, fused, and best-selected features on five well-known datasets. Also, a comprehensive comparison is conducted with recent techniques.

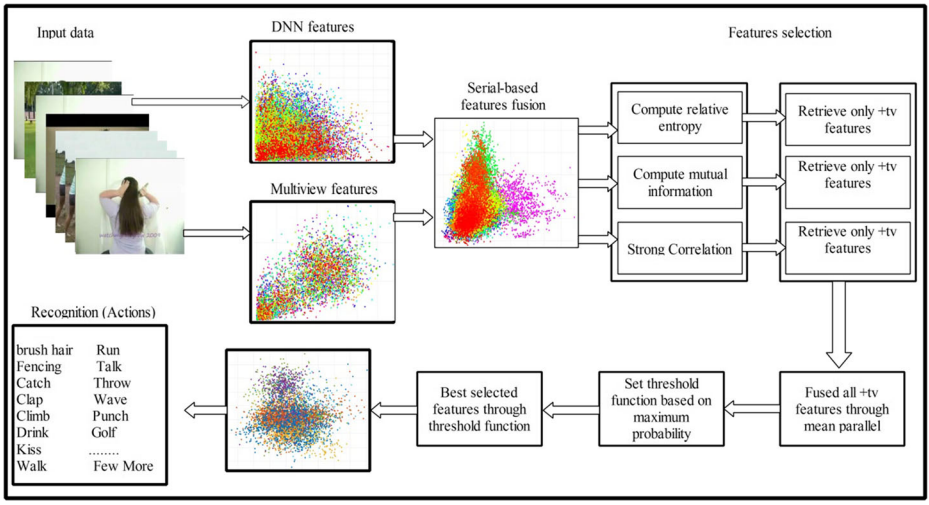

Fig. 1 Proposed methodology of human action recognition using multi-view and DNN features selection

4 Proposed methodology

The proposed human action recognition (HAR) method under Multiview scenarios is presented in Fig. 1. A simple but effective architecture is designed for action recognition by employing multi-view and high-level features. Both types of features are fused in serial approach, and selection of best among them is done through three parameters i.e. relative entropy, mutual information, and strong correlation. Later, all these parameters are combined by employing a mean parallel fusion approach, and help in designing a high probability based threshold function to select the best features. These features are finally provided to Naïve Bayes classifiers for final recognition. The details of each step are highlighted in Fig. 1.

4.1 DNN-multiview features

Deep Neural Network (DNN) has put an enormous effort in the area of machine learning (ML) against a large number of datasets. It is successfully applied in many machine learning (ML) applications including medical, surveillance, and a many more. CNN is an alternative form of DNN that produces both low and high-level features. In this work, a fused framework is designed for HAR. Furthermore in our research work the DNN model, and calculated multiview features are combined through a serial-based approach. Through DNN, the high-level features are computed, whereas, the multiview features are utilized for better recognition of human activities under different camera positions. The mathematical description of both DNN and multi-view features is discussed in the sub-sequent section.

4.2 Modified DNN features

In this work, a VGG19 pre-trained model [31] is utilized which consists of L number of FC layers where $L = 1, 2, 3$. The P^Lunits are in Lth FC layers for $L^{(1)} = 4096$, $L^{(2)} = 4096$,

and $L^{(3)} = 1000$, respectively. For the given selected datasets denoted by $\boldsymbol{\Delta}$ and training samples are denoted by $X_i^j \in \boldsymbol{\Delta}$. Each sample X_i^j is also $\mathbb{R}$. By following this, the output of the first layer is:

$$\boldsymbol{\lambda}^{(\mathbf{1})} = \boldsymbol{f}\left(m^{(1)} X_i^j + \beta^{(1)}\right) \in \mathbb{R}^{(1)} \tag{1}$$

Where, $f(.)$ denotes the ReLU activation function, β^1 is a bias vector, and $m^{(1)}$ denotes the weights matrix of the first layer, defined as:

$$m^{(1)} \in \mathbb{R}^{L^{(1)} \times k} \tag{2}$$

The output of the first layer is utilized as an input of the next layer which is defined through the following mathematical expression:

$$\boldsymbol{\lambda}^{(\mathbf{2})} = \boldsymbol{f}\left(m^{(2)} \boldsymbol{\lambda}^{(\mathbf{1})} + \beta^{(2)}\right) \in \mathbb{R}^{(2)} \tag{3}$$

$$\boldsymbol{\lambda}^{(\mathbf{3})} = \boldsymbol{f}\left(m^{(3)} \boldsymbol{\lambda}^{(\mathbf{2})} + \beta^{(3)}\right) \in \mathbb{R}^{(3)} \tag{4}$$

Where, $m^{(2)} \in \mathbb{R}^{L^{(2)} \times L^{(1)}}$ and $m^{(3)} \in \mathbb{R}^{L^{(3)} \times L^{(2)}}$. Similarly, $\boldsymbol{\lambda}^{(N)}$denotes the last fully connected layer which utilized in this work for high-level features extraction. Mathematically, FC layer is expressed as:

$$\boldsymbol{\lambda}_{\boldsymbol{g}}\left(X_i^j\right) = \boldsymbol{\lambda}^{(\boldsymbol{L})} = \boldsymbol{f}\left(m^{(L)} \boldsymbol{\lambda}^{(\boldsymbol{L-1})} + \beta^{(L)}\right) \in \mathbb{R}^{(L)} \tag{5}$$

A transfer learning-based feature mapped on the destination datasets. In this scenario, the source data is imagenet dataset whereas the destination is selected action recognition datasets. Cross entropy is applied as a loss function on FC8–1000, mathematically defined as:

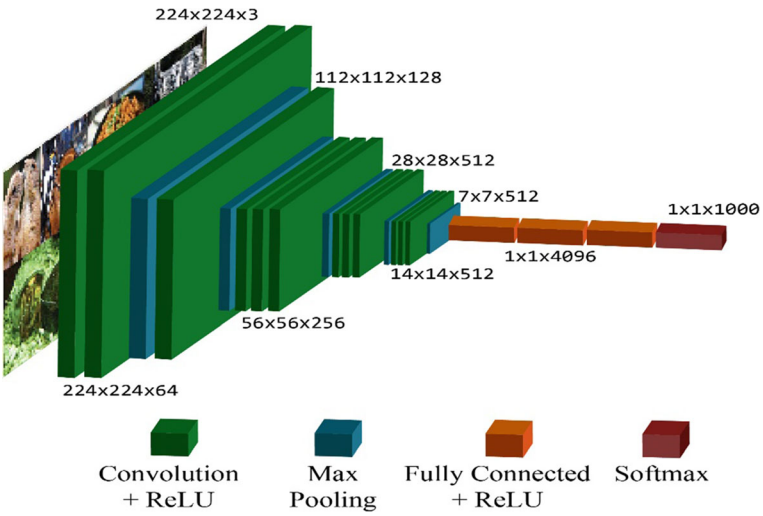

Fig. 2 Layers description in VGG19 whereas the size of the input is 224 × 224 (*RGB*)

$$C^{(E)} = -\sum_{cl=1}^{M} B_{(0b,cl)} \log\left(p_{(ob,cl)}\right) \tag{6}$$

Where, $C^{(E)}$ denotes the cross-entropy function, $\boldsymbol{B}$ indicates the number of classes cl, wheras, observationsob, and p is a predicted probability. This function returns a deep feature vector of dimension $N \times d1$, denoted by $\breve{V}_d$ where $d1 = 1000$ as a graphical description of VGG19 based on its layers architecture as shown in Fig. 2. In Fig. 2, it is shown that the VGG19 includes a total of 5 pools (max pool layers), 5 convolutional and 3 FC layers. This architecture has a total of 144 M parameters which are initially used for training. During the activation on the FC8 layer, few parameters are defined like momentum is 0.9, the number of epochs is 164, mini-batch size is 64, and the learning rate is 0.01, respectively.

4.3 Multiview features

As we have the selected datasets, denoted by $\boldsymbol{\Delta}$**.** Suppose we have an output Multiview feature vector $\boldsymbol{V_i} \in (x_i, y_i)$ computed from the original video frames of selected datasets. The posterior mean of $\boldsymbol{V_i}$ is denoted by $\xi_{i,k} = \boldsymbol{E}(\xi_k | x_i, y_i)$. This mean is utilized to compute the latent vector of $\boldsymbol{V_i}$ which is formulated as follows:

$$\xi_k = \left[\boldsymbol{W}_x^{kT}, \boldsymbol{W}_y^{kT}\right] \varphi_k^{-1} \sum_i \lambda_{i,k} (\boldsymbol{V_i} - \boldsymbol{E}_k) \tag{7}$$

Where, $\boldsymbol{W}_x$ and $\boldsymbol{W}_x$ are weight matrix of dimension $n \times d$and $m \times d$, respectively. $\boldsymbol{E}$represents a mean parameter and φ is a local standard deviation of $\boldsymbol{V_i}$. Then compute the gradient information of each frame in both x and y directions, denoted by $g_x$and g_y. Mathematically, the g_x is formulated as follows:

$$g_x = \left\{ \frac{\partial \Delta(\theta)}{\partial \boldsymbol{E}_x^k}, \frac{\partial \Delta(\theta)}{\partial \varphi_x^k} \right\}_{k=1,2,3,\ldots,N} \tag{8}$$

$$\frac{\partial \Delta(\theta)}{\partial \boldsymbol{E}_x^k} = \frac{-2 \sum_i \lambda_{(i,k)} x_{(i,k)}}{\varphi_x^k} \tag{9}$$

$$\frac{\partial \Delta(\theta)}{\partial \varphi_x^k} = \frac{2\left(\psi_k \varphi_x^k - Dg\left(\sum_i \lambda_{(i,k)} x_{(i,k)} \boldsymbol{x_{(i,k)}}\right)\right)}{\varphi_x^k} \tag{10}$$

Where, $\psi_k = \sum \lambda_{(i,k)}$and $\varphi_x^k = \sqrt{Dg(\psi)}$. The ψ is a diagonal value of the vector $\boldsymbol{V_i}$. Similarly, the gradient g_y^i is formulated as follows:

$$g_y = \left\{ \frac{\partial \Delta(\theta)}{\partial \boldsymbol{E}_y^k}, \frac{\partial \Delta(\theta)}{\partial \varphi_y^k} \right\}_{k=1,2,3,\ldots,N} \tag{11}$$

$$\frac{\partial \Delta(\theta)}{\partial \boldsymbol{E}_y^k} = \frac{-2\sum_i \lambda_{(i,k)} y_{(i,k)}}{\varphi_y^k} \tag{12}$$

$$\frac{\partial \Delta(\theta)}{\partial \varphi_y^k} = \frac{2\left(\psi_k \varphi_y^k - Dg\left(\sum_i \lambda_{(i,k)} \boldsymbol{y}_{(\boldsymbol{i},\boldsymbol{k})}\right)\right)}{\varphi_y^k} \tag{13}$$

Where $\varphi_y^k = \sqrt{Dg(\psi)}$. After that, compute the vertical features of each frame due to a change in human directions under different scenarios. These features are computed through the following formulation by employing both rows and columns values.

$$V(r) = V(r) + (j \times \phi_{\tau c}) \tag{14}$$

$$V(r_x) = V(r_x) + V(r) \tag{15}$$

$$V(\mathbf{C}) = V(\boldsymbol{C}) + (b \times \phi_{\tau c}) \tag{16}$$

$$V(\boldsymbol{C_y}) = V(\boldsymbol{C}_y) + V(\boldsymbol{C}) \tag{17}$$

Where, $V(r_x)$ denotes the rows pixel of each frame and $V(\boldsymbol{C_y})$ represents the column pixels. The variable $\phi_{\tau c}$ used as a cropping parameter in the whole process. Then, the values are utilized through the following formulations for final feature extraction.

$$V_1(r) = \frac{V(r_x)}{N} \tag{18}$$

$$V_2(\boldsymbol{C}) = \frac{V(\boldsymbol{C}_y)}{N} \tag{19}$$

This process is completed for all pixels of the given frames for selected datasets, thus, obtain a vector of dimension $N \times K$. Finally, we combine all Multiview features through the following concatenation process:

$$\boldsymbol{V_{mi}} = \left\{g_x, g_y, V_1(r), V_2(\boldsymbol{C})\right\} \tag{20}$$

Where, the dimension of $\boldsymbol{V}_{mi}$ is $N \times d2$where $d2 = (n + m + k + l)$.

4.4 Features fusion and selection

As we have two feature vectors $\breve{V}_d$ and $\boldsymbol{V_{mi}}$ of dimensions $N \times d1$ and $N \times d2$. Let we have a fused vector $\boldsymbol{V}_{\overset{\frown}{\boldsymbol{d}}}$of dimension $N \times \overbrace{d}$. Then, the features are fused by employing a simple concatenation through the following formulation:

$$\boldsymbol{V}_{\overbrace{\boldsymbol{d}}} = \begin{pmatrix} \breve{V}_d \\ \mathrm{V}_{mi} \end{pmatrix} \tag{21}$$

$$\boldsymbol{V}_{\overbrace{\boldsymbol{d}}} = \begin{pmatrix} N \times d1 \\ N \times d2 \end{pmatrix} = N \times \overbrace{d} \tag{22}$$

The major challenge of existing HAR techniques, is the high dimensionality of extracted feature points. In the video frame, the background pixels generate irrelevant features that misleads the classification accuracy. Moreover, many features are shrill, less instructive, and redundant. The problem of redundant features has occurred after the fusion process. Therefore, it is essential to propose a new feature selection technique that not only leads the overall system accuracy but also improves the system efficiency. In this work, the fusion of multiview and DNN features increases the human action information under the complex scenarios, but it also introduces a limitation when a few redundant features are combined in one matrix. To resolve this kind of problem, we suggest a feature selection scheme that selects only active features for final recognition. The proposed selection scheme includes save positive (+tv) features, followed by designing the maximum probability-based function and at the end selection of the best ones for final classification.

4.4.1 Save and combine positive features

As we have fused feature vector of dimension $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$ of dimension $N \times \overbrace{d}$ which includes several negative features. Therefore, the major aim of this step is to remove these negative features which are included in $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. To resolve this problem, we compute three parameters separately such as relative entropy, mutual information, and strong correlation coefficient (SCC) from vector $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. Mathematically, the relative entropy is defined as:

$$E_r\left(f_i || C\right) = \sum_{r \in N} p(r) \frac{\log p(r)}{C(r)} \tag{23}$$

Where, $C(r)$ denotes the column values of the given matrix which are utilized as a relative entropy features extraction, $p(r)$ denotes represent the probability value of row features, and $f_i \in \boldsymbol{V}$. This expression returns an entropy vector of dimension $N \times \overbrace{d1}$. After that compute the mutual information vector by employing fused vector $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. Here we utilized a symbol f_i which is defined as $f_i \in \boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. Mathematically, the MI is formulated as:

$$MI(f_i; C) = \widetilde{D}\Big((f, C) \| p(f_i) p(y)\Big) \tag{24}$$

$$= \sum_{f_i,C} p(f_i, C)\log\frac{p(f_i, C)}{p(f_i), p(C)} \tag{25}$$

$$= \sum_{f_i,C} p(f_i, C)\log p(f_i, C) - \boldsymbol{G1} - \boldsymbol{G2} \tag{26}$$

$$\boldsymbol{G1} = \sum_{f_i,C} p(f_i, C)\log p(f_i) \tag{27}$$

$$\boldsymbol{G2} = \sum_{f_i,C} p(f_i, C)\log p(C) \tag{28}$$

$$= -\vartheta(f_i, C) + \vartheta(f_i) + \vartheta(C) \tag{29}$$

$$MI(f_i; C) = \vartheta(f_i) - \vartheta(f_i/C) \tag{30}$$

$$MI(f_i; C) = \vartheta(f_i) - \vartheta(C/f_i) \tag{31}$$

The $MI(f_i; C)$ returns an MI vector of dimension $N \times \overbrace{d2}$ and it contains only positive features. Finally, we compute an SCC based feature vector by employing the fused vector $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. The strongly correlated vector is defined as:

$$\rho\left(f_i, f_j\right) = \sum_{i=1}^{N} \frac{(S1 \times S2)}{N-1} \tag{32}$$

$S1 = \sqrt{\frac{(f_i - \overline{f_i})}{\sigma(f_i)}}$, $S2 = \sqrt{\frac{(f_j - \overline{f_j})}{\sigma(f_j)}}$ (33).Where $S1$ and $S2$ are standardized versions of row (f_i) and column (f_j) features of vector $\boldsymbol{V}_{\overbrace{\boldsymbol{d}}}$. The above expression returns a strongly correlated vector of dimension $N \times \overbrace{d3}$, where the values in vector $\rho(f_i, f_j)$ are near to 1 due to the most strong and positive correlated features. The all these vectors return only positive features and if few of them are negative then we simply neglect then by threshold function. After that, combine these all features by employing a parallel fusion scheme. For parallel fusion approach, initially equal the length of all three extracted vectors E_r, MI, and ρ where the dimension of each vector is $N \times \overbrace{d1}$, $N \times \overbrace{d2}$, and $N \times \overbrace{d3}$, respectively. The higher dimension vector is chosen and computes its mean value which is utilized as a padding value in smaller length vectors. Finally, we design a threshold function that returns only those features which are greater in value at the current index. Mathematically, the threshold function is expressed as follows:

$$\boldsymbol{V_{fp}} = \begin{cases} V_{si} & for \quad \underset{E_r, MI, \rho \; follows \; Mx}{\operatorname{argmax}} (E_r(i), MI(j), \rho(k)) \\ Neglect \end{cases} \tag{34}$$

In the above function, the three vectors E_r, MI, ρ are fused based on the index values and from each vector, the index features are compared and the maximum value is selected for a new parallel fused vector. This process is continued until the length of the fused parallel vector $\boldsymbol{V_{fp}}$. In the above expression, the symbol V_{si} denotes maximum index features from all three vectors E_r, MI, ρ, respectively. The new obtained parallel fused vector of length $N \times \overbrace{d4}$. Finally, we compute the probability value of each feature of the vector $\boldsymbol{V_{fp}}$and based on the maximum probability value feature, a fitness function is design for final selection.

$$\boldsymbol{Th}(\boldsymbol{V}) = \begin{cases} V_s(i) \;\; for \;\; V_{si}(r_i) \geq p(\boldsymbol{V_{fp}}) \\ Remove \;\; Others \end{cases} \tag{35}$$

In this case, we have 5 different probabilities values for all 5 selected datasets. The probability value of KTH dataset is 0.27, UCF sports is 0.33, IXMAS is 0.41, HMDB51 is 0.19, and YouTube is 0.14, respectively. Finally, the final selected vector is provided to the Naïve Bayes classifier for final recognition [23]. Mathematically, Naïve Bayes is defined as:

$$P(Y|V_s) = \frac{P(V_s|Y)P(Y)}{P(V_s)} \tag{36}$$

Where, Y denotes class labels such as golf, walk, run, wave etc. V_s denotes extracted fused features represented as $V_s \in V_s(1)$, $V_s(2)$, …, $V_s(i)$. So, the above expression is modified as:

$$P(Y|V_s(1), V_s(2), \ldots, V_s(i)) = \frac{P\left(V_s(1)|Y \; P(V_s(2)|Y), \ldots, P\left(V_s(i)|Y\right)\right.}{P(V_s(1)), P(V_s(2)), \ldots, P(V_s(i))} \tag{37}$$

$$P(Y|V_s(1), V_s(2), \ldots, V_s(i)) = \propto P(Y) \prod_{i=1}^{n} P(V_s(i)|Y) \tag{38}$$

$$\mathrm{Y} = \underset{Y}{\operatorname{argmax}} P(Y) \prod_{i=1}^{n} P(V_s(i)|Y) \tag{39}$$

A few samples of labelled recognition results are shown in Fig. 3.

5 Results and analysis

In this section, first, we describe the datasets and then introduced performance parameters that are used for evaluation. Later, the details of the trained model is presented for evaluation, followed by the test results.

Five publically available datasets are utilized in this work for evaluation of the proposed recognition algorithm. The selected datasets are- HMDB51 [15], KTH [16], UCF Sports [24],

Fig. 3 Proposed labelled results using best features along with Naïve Bayes classifier

YouTube [16], and IXMAS [37]. Three performance parameters are used i.e. accuracy, FNR, and computation time for the evaluation of recognition process. Different types of experiments are conducted for analysis of results: a) modified DNN features based accuracy; b) multi-view features based accuracy; c) fusion of multi-view and DNN features; and d) proposed selected features.

5.1 Implementation detail

Initially, the selected datasets are split randomly in a ratio 50:50. Then we perform the proposed recognition approach and select the best features. To train the Naïve Bayes classifier using proposed selected features, a 10 fold cross-validation is employed. The overall implementation of the proposed recognition system is conducted in MATLAB 2018b using a CoreI7 Desktop Computer with 16 GB RAM and 8 GB Graphics cards.

5.2 Results

This section discusses the different results of the proposed method. The proposed method is applied on several datasets and results for each dataset is discussed.

5.2.1 HMDB51 dataset results

The HMDB51 is the largest datasets of action recognition which includes 51 numbers of actions like a hairbrush, catch fencing, etc. A total of 7000 videos of different varieties are included which are collected from movies and YouTube. The sample frame is shown in Fig. 4.

The results are presented in Table 1 for all four experiments. It can be seen in Table 1, that the proposed recognition approach have performed quite well with the Naïve Bayes method and attained an accuracy of 93.7% along with FNR is 6.3% and overall testing time is 88.4 s. The accuracy performance is also validated through Fig. 5 (d). The fusion of deep and multi-view features also provides better accuracy of 91.3% along with FNR is 8.7%, but the testing time is 700.5 (sec), which is too high. This accuracy is also validated in Fig. 5 (c). Moreover, the individual accuracy of both deep and multi-view features is computed and presented in this table. The best individual accuracy is 89% and 76.4% respectively, which is validated from Fig. 5 (a) and (b). Moreover, the individual computation time is 314.6 (sec) and 456.9 (sec),

Fig. 4 Sample frames of HMDB51 action dataset [15]

Table 1 Proposed Recognition accuracy of the proposed method using HMDB51 dataset

Method	Features Type				Evaluation Metrics		
	DNN	MV	Fused	Selected	Accuracy (%)	FNR (%)	Time (sec)
MSVM	✓				76.5	23.5	311.4
		✓			64.2	35.8	402.5
			✓		78.9	21.1	622.8
				✓	82.4	17.6	211.5
Ensemble	✓				78.9	21.1	317.9
		✓			67.8	32.2	416.7
			✓		81.3	18.7	689.9
				✓	82.9	17.1	297.4
Naïve Bayes	✓				89.0	11.0	314.6
		✓			76.4	23.6	456.9
			✓		**91.3**	**8.7**	700.5
				✓	**93.7**	**6.3**	188.4

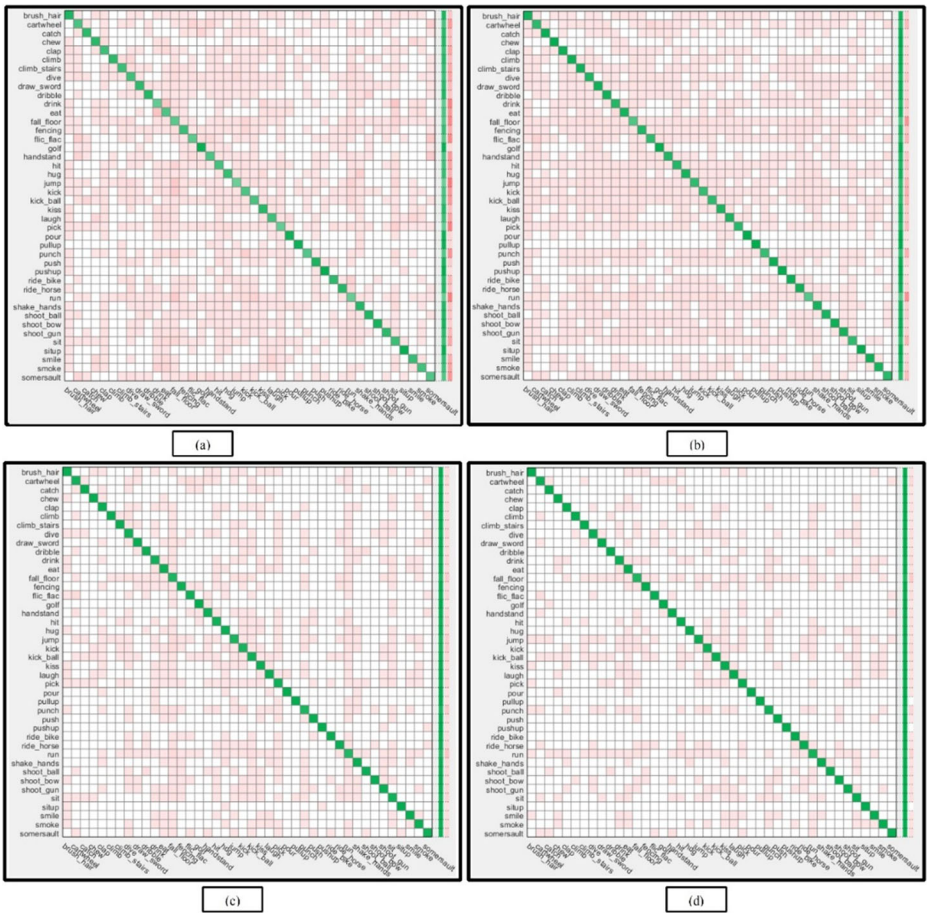

Fig. 5 Confusion matrixes of HMDB51 dataset for Naïve Bayes classifier on different feature types. (**a**) DNN features, (**b**) Multiview features; (**c**) proposed fused features; (**d**) Proposed selected features

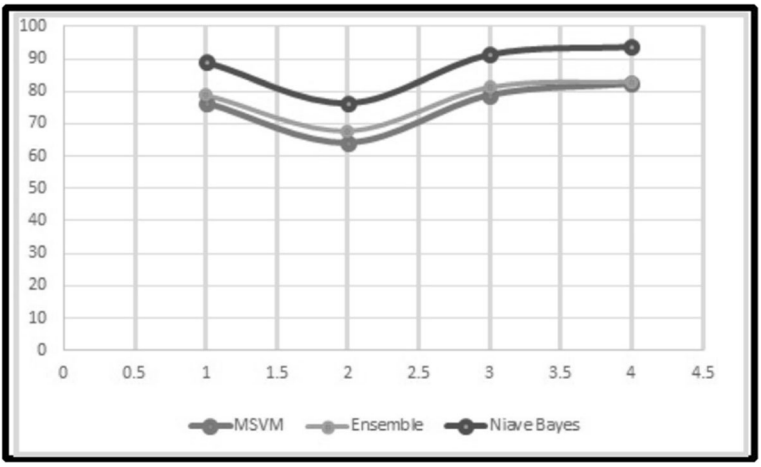

Fig. 6 Accuracy comparison of different methods based on four selected experiments. The representation of accuracy is described from left to right. 1. DNN features, 2. MV features, 3. Fused features, 4. Proposed selected

respectively. The accuracy of Naïve Bayes classifier is also compared with MSVM and Ensemble methods, as presented in Table 1. However, the Naïve Bayes well performed for all experiments, as visually plotted in Fig. 6.

5.2.2 KTH dataset results

KTH activity dataset was created in 2004. This dataset is captured under both indoor and outdoor environments. All videos are captured using static cameras. 6 action classes are performed in this dataset such as boxing, handclapping, and few more. Each action is performed by 25 individuals and a total of 600 videos captured, as sample frames are demonstrated in Fig. 7.

The results are presented in Table 2 for all selected experiments. In this table, the proposed recognition approach well performed on the Naïve Bayes method, maximum accuracy of 97.0% along with FNR is 3%. The overall testing time of Naïve Bayes classifier is also computed and executed in 33.1 s. The accuracy performance of this classifier is validated in Fig. 8 (d). The accuracy of fused deep and multi-view features is also provided and attained an

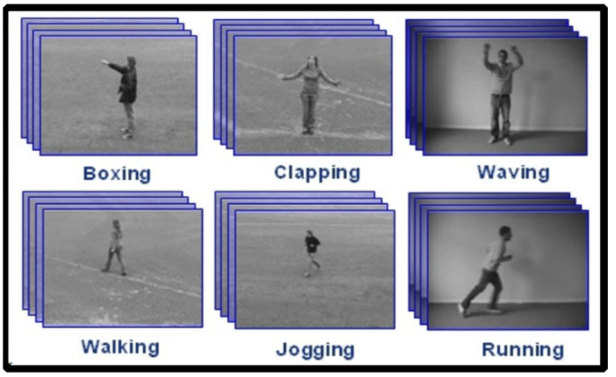

Fig. 7 Sample frames of the KTH dataset [16]

Table 2 Proposed Recognition accuracy of the proposed method using KTH dataset

Method	Features Type				Evaluation Metrics		
	DNN	MV	Fused	Selected	Accuracy (%)	FNR (%)	Time (sec)
MSVM	✓				93.1	6.9	76.2
		✓			58.0	42	79.7
			✓		94.0	6.0	89.2
				✓	95.9	4.1	36.3
Ensemble	✓				89.6	10.4	89.4
		✓			68.1	31.9	81.2
			✓		94.4	5.6	93.5
				✓	96.6	3.4	47.4
Naïve Bayes	✓				95.1	4.9	72.5
		✓			79.8	20.2	68.9
			✓		**95.7**	**4.3**	81.3
				✓	**97.0**	**3.0**	33.1

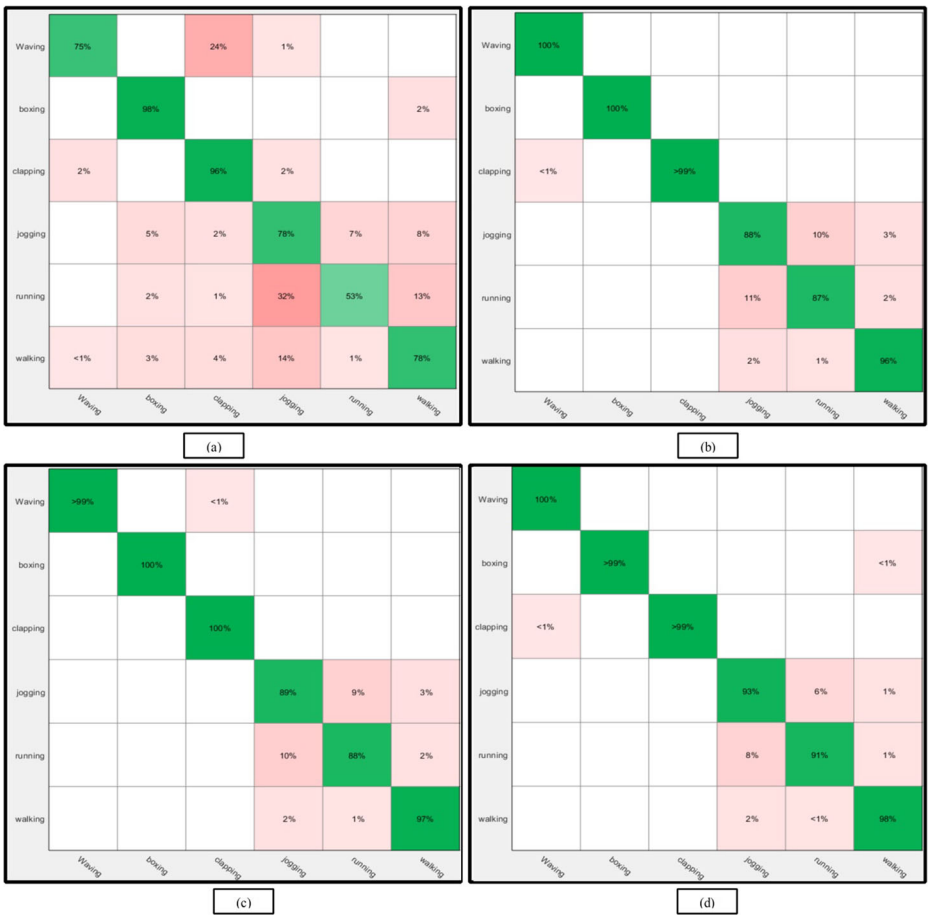

Fig. 8 Confusion matrixes of the KTH dataset for Naïve Bayes classifier on different feature types. (**a**) DNN features, (**b**) Multiview features; (**c**) proposed fused features; (**d**) Proposed selected features

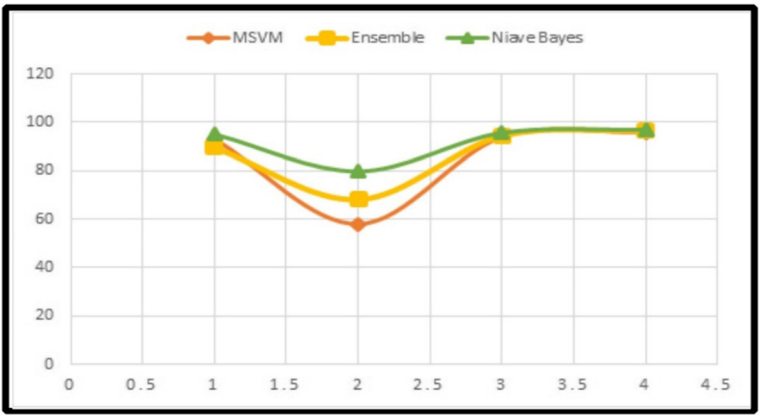

Fig. 9 Accuracy comparison of different methods based on four selected experiments using the KTH dataset. The representation of accuracy is described from left to right. 1. DNN features, 2. MV features, 3. Fused features, 4. Proposed selected

accuracy of 95.7% which is validated through Fig. 8 (c). However, the execution time is increased up to 8.3 s. Moreover, the individual accuracy of both deep and multi-view features is also computed and provided in Table 2. The best individual accuracy is 95.1% and 79.8%, respectively which can be validated through Fig. 8(a) and (b). The computation time for each feature set on Naïve Bayes classifier is 72.5% and 68.9%, respectively. The accuracy of the proposed system is also computed on MSVM and Ensemble methods, as presented in Table 2 and shows that Naïve Bayes well performed for all of them, as visually accuracy of each of them is plotted in Fig. 9.

5.2.3 UCF Sportsdataset results

This dataset includes a total of 150 video sequences of resolution 720×480. A total of 13 actions is performed in this dataset. The videos of this dataset are collected from different sports channels such as BBC and ESPN. A few sample images are demonstrated in Fig. 10.

The recognition results are presented in Table 3 and attained a maximum accuracy of 98% on proposed selected features with FNR is 2.0%. The performance of the Naïve Bayes

Fig. 10 Sample frames of UCF Sports dataset [24]

Table 3 Proposed Recognition accuracy of the proposed method using UCf Sports dataset

Method	Features Type				Evaluation Metrics		
	DNN	MV	Fused	Selected	Accuracy (%)	FNR (%)	Time (sec)
MSVM	✓				68.5	31.9	121.6
		✓			59.3	40.7	198.6
			✓		94.4	5.1	251.5
				✓	97.1	2.9	104.3
Ensemble	✓				90.6	9.4	127.9
		✓			79.5	20.5	182.4
			✓		94.6	5.4	294.9
				✓	97.2	2.8	107.5
Naïve Bayes	✓				92.5	7.5	111.5
		✓			89.0	11	188.9
			✓		**96.3**	**3.7**	240.5
				✓	**98.0**	**2.0**	92.48

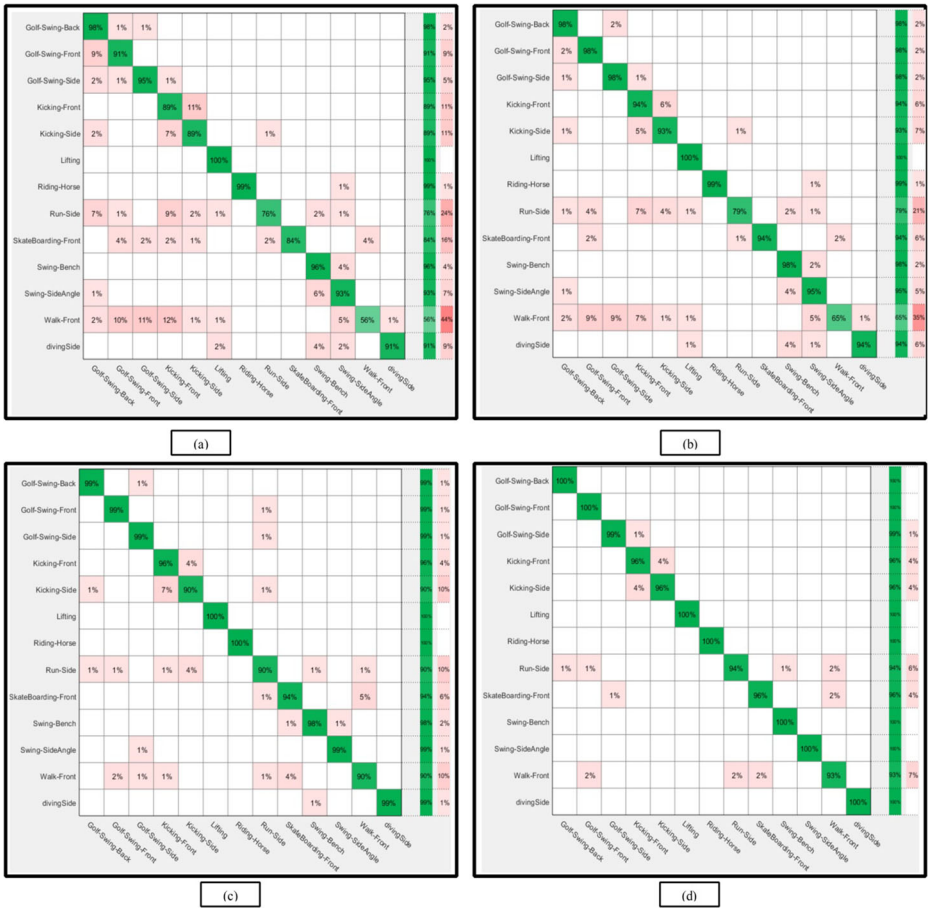

Fig. 11 Confusion matrixes of UCf Sports dataset for Naïve Bayes classifier on different feature types. (**a**) DNN features, (**b**) Multiview features; (**c**) proposed fused features; (**d**) Proposed selected features

classifier on selected features is also validated through Fig. 11 (d). The individual features performance is also computed for a better analysis of the proposed results. The best individual accuracy of DNN and MV is 92.5% and 89%, respectively for Naïve Bayes classifier, also validated through Fig. 11 (a) and (b). The overall testing time of Naïve Bayes classifier is 111.5 s, 188.9 s, and 92.48 s. The fusion of both DNN and MV features accuracy is reached to 96.3%, validated through Fig. 11 (c) which is better as compared to individual DNN and MV features, respectively but the execution time is exceeded up to 240.5 s. Due to high computation time, the features selection technique is implemented and minimizes the execution time up to 92.48 s. In the end, the proposed accuracy on Naïve Bayes classifier is compared with two other classification methods, named MSVM and Ensemble methods, as results presented in Table 3. From the results, it is clearly shown that the Naïve Bayes classifier outperforms on proposed selected features, as accuracy is plotted in Fig. 12.

5.2.4 YouTube action dataset results

UCF YouTube action dataset consists of a total of 11 action classes including horse driving, biking, cycling, and few more. Due to a huge variation in camera movement, change in viewpoint, pose, and complex background, it is a challenging dataset. This dataset is completed by 25 groups of individuals. A few sample frames are demonstrated in Fig. 13.

The recognition results on this dataset are presented in Table 4 and attained a maximum accuracy of 99.4% on proposed selected features using Naïve Bayes classifier with FNR is 0.6%. The performance of this classifier can be validated through Fig. 14 (d). For a better analysis of propose selected features accuracy, the individual and fused features are also employed for recognition performance. The best individual accuracy of DNN features is 98.1% on the Naïve Bayes classifier, validated through Fig. 14 (b) whereas, for MV features, the best accuracy is 81.7% on Ensemble classifier. Later, fused both DNN and MV features and attain accuracy of 98.4%, validated through Fig. 14 (e). But the computation time of the fusion process is increased up to 19.9 s for Naïve Bayes whereas the individual testing time is 104.9 s and 112.6 s, respectively. This time is minimized through the proposed selection

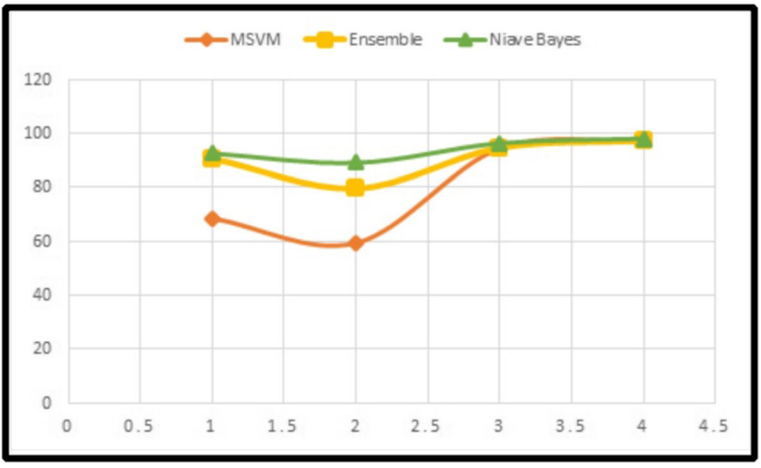

Fig. 12 Accuracy comparison of different methods based on four selected experiments using the UCF Sports dataset. The representation of accuracy is described from left to right. 1. DNN features, 2. MV features, 3. Fused features, and 4. Proposed selected

Fig. 13 Sample frames of YouTube action dataset [16]

process and minimized up to 59.4 s. In the end, the proposed accuracy on Naïve Bayes classifier is compared with two other classification methods, named MSVM and Ensemble methods, as results presented in Table 4. From the results, it is clearly shown that the Naïve Bayes classifier outperforms on proposed selected features, as accuracy is plotted in Fig. 15.

5.2.5 IXMAS action dataset

IXMAS action dataset consists of 11 action classes like check watch, pickup, and many more. A total of 1148 video sequences are performed by 5 male and 5 female actors. All actors can freely change their orientation during the acquisition of video sequences. The 5 standard cameras are used for the acquisition process. A few sample frames are shown in Fig. 16.

The results of this dataset are given in Table 5 for all selected experiments. In this table, the proposed recognition approach well performed on the Naïve Bayes method, attained

Table 4 Proposed Recognition accuracy of the proposed method using YouTube dataset

Method	Features Type				Evaluation Metrics		
	DNN	MV	Fused	Selected	Accuracy (%)	FNR (%)	Time (sec)
MSVM	✓				94.2	5.8	122.4
		✓			76.9	23.1	137.1
			✓		95.8	4.2	158.9
				✓	98.7	1.3	69.4
Ensemble	✓				90.9	9.1	107.5
		✓			81.7	18.3	119.4
			✓		93.9	6.7	148.9
				✓	95.6	4.4	76.5
Naïve Bayes	✓				98.1	1.9	104.9
		✓			62.3	37.7	112.6
			✓		**98.4**	**1.6**	129.9
				✓	**99.4**	**0.6**	59.4

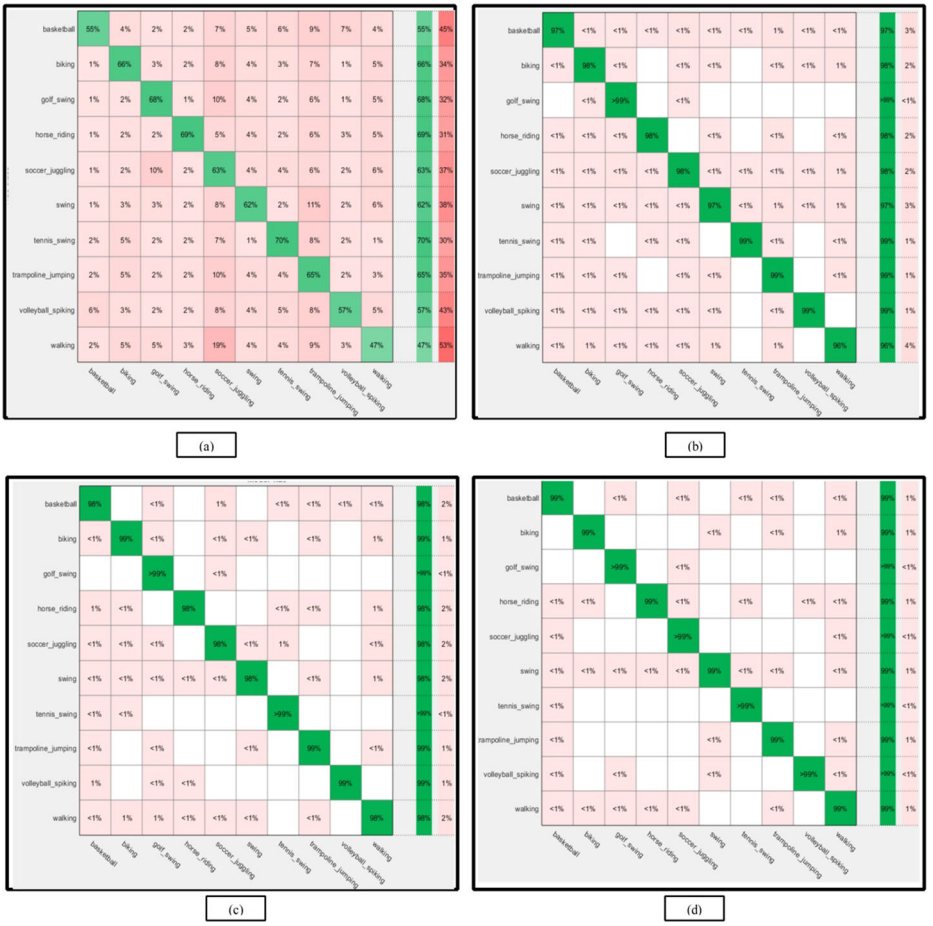

Fig. 14 Confusion matrixes of the YouTube dataset for Naïve Bayes classifier on different feature types. (**a**) DNN features, (**b**) Multiview features; (**c**) proposed fused features; (**d**) Proposed selected features

maximum accuracy of 95.2% along with FNR is 4.8%, validated through Fig. 17 (d). The overall testing time of this classifier is 69.84 s on selected features. For analysis of proposed selected features accuracy, individual features based results are computed and attained an accuracy of 77.8% and 64% for Naïve Bayes, validated from Fig. 17(a) and (b). Then, fused both feature sets and reached accuracy up to 81.1%, validated through Fig. 17 (e) but the execution time is almost double which is minimized through best-selected features. In the end, the proposed system is also validated on MSVM and Ensemble methods, results presented in Table 5 and show that Naïve Bayes overall outperforms. The accuracy comparison between all three classifiers is also plotted in Fig. 18.

5.3 Analysis and comparison

The proposed action recognition scheme is evaluated on five datasets through various experiments. Initially, accuracy is computed on each dataset by employing individual DNN and

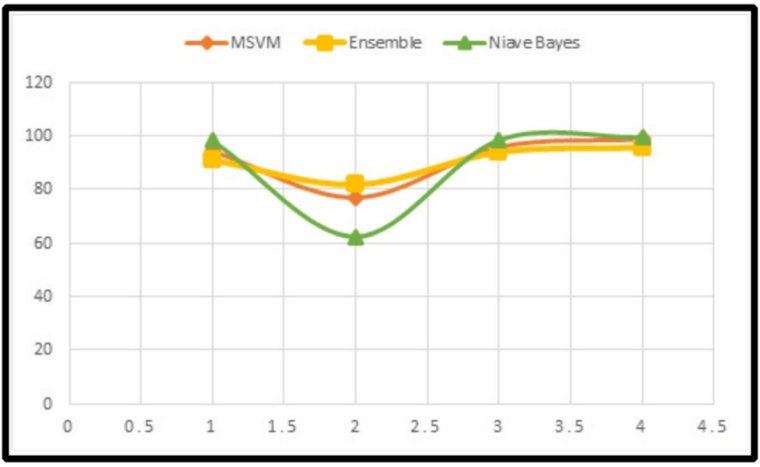

Fig. 15 Accuracy comparison of different methods based on four selected experiments using the YouTube action dataset. The representation of accuracy is described from left to right. 1. DNN features, 2. MV features, 3. Fused features, and 4. Proposed selected

Multiview features and compute accuracy as given in Table 1-5. The results show that DNN performs well as compare to Multiview but the computational time of Multiview features is a little bit low as compare to DNN. Later, fused both DNN and Multiview features and accuracy are increased up to 10% due to better information but the computation time is very high and almost double. To handle this issue, we proposed a selection technique that selects the best features which not only increase the accuracy rate but also minimized the overall computational time. A detailed statistical analysis is also performed for proposed method in which 1000 iterations are performed to check the consistency in the results, as given in Table 6. In this table, it is observed that maximum 3 to 4% change is occurred after 1000 iterations which shows the consistency of implemented method.

Fig. 16 Sample frames of IXMAS action dataset [37]

Table 5 Proposed Recognition accuracy of the proposed method using IXMAS Action dataset

Method	Features Type				Evaluation Metrics		
	DNN	MV	Fused	Selected	Accuracy (%)	FNR (%)	Time (sec)
MSVM	✓				71.1	28.9	107.96
		✓			66.5	33.5	156.6
			✓		76.9	23.1	211.9
				✓	87.5	12.5	89.6
Ensemble	✓				69.5	30.5	116.04
		✓			67.2	32.8	145.5
			✓		73.6	26.4	201.5
				✓	84.3	15.7	74.8
Naïve Bayes	✓				77.8	22.2	109.54
		✓			64.0	36.0	113.6
			✓		**81.1**	**18.9**	194.2
				✓	**95.2**	**4.8**	69.84

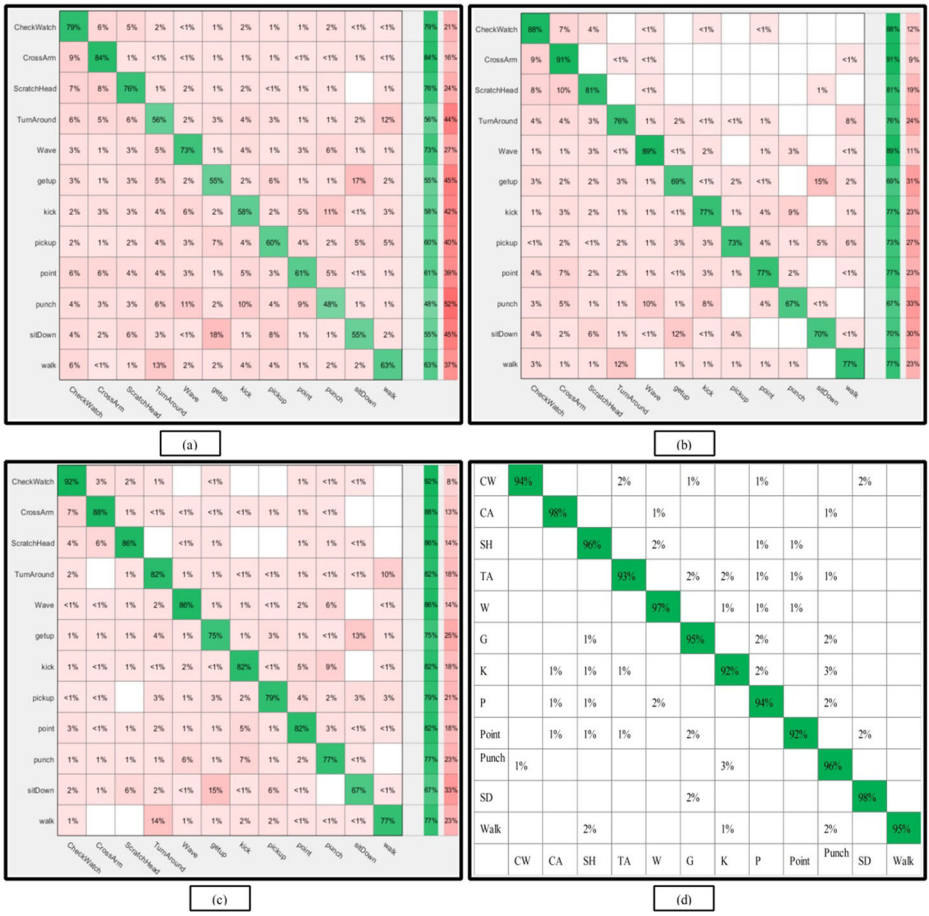

Fig. 17 Confusion matrixes of the IXMAS Action dataset for Naïve Bayes classifier on different feature types. (**a**) DNN features, (**b**) Multiview features; (**c**) proposed fused features; (**d**) Proposed selected features

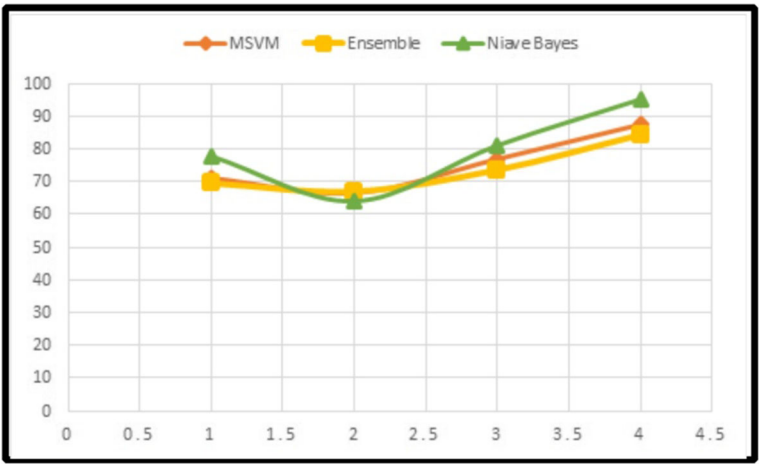

Fig. 18 Accuracy comparison of different methods based on four selected experiments using the IXMAS dataset. The representation of accuracy is described from left to right. 1. DNN features, 2. MV features, 3. Fused features, and 4. Proposed selected

At the last, an extensive comparison with existing techniques is conducted; give in Table 7 on the same datasets based on accuracy value. The most recent, best reported accuracy of HMDB51 dataset is 70.3% [13] whereas other reported accuracies are 69% [18], 65.4% [40], and 64.1% [36], respectively. In [2], authors presented an automated method for action recognition and attained accuracy of 86.40% using the UCF Sports dataset whereas other noted accuracies on the same dataset are 73.93% [38] and 82.14% [41]. Using the KTH dataset, authors in [17] obtained an accuracy of 94.3%. The other noted accuracies are 93.63% [41] and 86.70% [1], respectively. In [17], authors achieved an accuracy of 89.7% using YouTube dataset whereas for IXMAS dataset, previously best accuracy is 94.7% [17] and 70.8% [42]. In this work, the proposed fusion and selection method outperforms on selected datasets and achieved accuracy 93.7% using HMDB51, 95.2% for IXMAS, 98% using UCF Sports, 99.4% using YouTube, and 97% for KTH dataset. The results show that proposed method gives better results as compared to existing techniques.

Table 6 Analysis of proposed method using Naïve Bayes classifier after 1000 iterations

Dataset	Features		Accuracy (%)			Standard Dev
	Fused	Selected	Min	Avg	Max	
	✓		89.7	90.5	91.3	0.653
		✓	**90.9**	**92.3**	**93.7**	1.143
	✓		93.2	94.4	95.7	1.021
		✓	**96.1**	**96.5**	**97.0**	0.367
	✓		90.7	93.5	96.3	2.287
		✓	**95.9**	**96.9**	**98.0**	0.857
	✓		95.6	97.0	98.4	1.143
		✓	**97.1**	**98.2**	**99.4**	0.936
	✓		77.5	79.3	81.1	1.470
		✓	**93.0**	**94.1**	**95.2**	0.898

Table 7 Proposed method results comparison with existing techniques on the same datasets

Method	Year	Dataset	Accuracy (%)
[13]	2019	HMDB51	70.3
[18]	2019	HMDB51	69.0
[40]	2019	HMDB51	65.4
[36]	2019	HMDB51	64.1
[2]	2019	UCF Sports	86.40
[38]	2017	UCF Sports	73.93
[41]	2019	UCF Sports	82.14
[41]	2019	KTH	93.63
[1]	2016	KTH	86.70
[17]	2016	KTH	94.3
[17]	2016	YouTube	89.7
[17]	2016	IXMAS	94.7
[42]	2018	IXMAS	70.8
Proposed		**HMDB51**	**93.7**
		UCF Sports	**98.0**
		KTH	**97.0**
		YouTube	**99.4**
		IXMAS	**95.2**

6 Conclusion

Human action recognition (HAR) under multi viewpoints is a major challenge for correct recognition tasks. In this work, we proposed a new scheme for HAR under multiview scenarios. In this scheme, we fused multiview and DNN features and later select the best ones based on a probability function. As compared to previous methods, we focused on multiview features along with DNN based high-level features. The fusion of both features gives better accuracy using Naive Bayes classifier but due to an increase in the number of features, the overall time is increased. Therefore, we design a selection algorithm that focused on both accuracy and computational time. The experiments are conducted on five famous datasets and the proposed algorithm outperforms for all. From the results, we conclude that the multiview features are important when the actions are recognized under the various viewpoints. Moreover, the DNN based features are not affected by low noisy data and achieved significant accuracy from raw images. We also conclude that the selection of best features minimizes the number of predictors but accuracy is not diminished. In the future, we are planning to use a few advanced datsests like UCF101 [32], further use it to train a new CNN model from scratch. Moreover, the optimization of newly implemented model would be performed through Whale optimization algorithm.

References

1. Ahad MAR, Islam MN, Jahan I (2016) Action recognition based on binary patterns of action-history and histogram of oriented gradient. Journal on Multimodal User Interfaces 10:335–344
2. Aly S, Sayed A (2019) Human action recognition using bag of global and local Zernike moment features. Multimed Tools Appl:1–31
3. Arshad H, Khan MA, Sharif M, Yasmin M, Javed MY (2019) Multi-level features fusion and selection for human gait recognition: an optimized framework of Bayesian model and binomial distribution. Int J Mach Learn Cybern:1–18

4. Aurangzeb K, Haider I, Khan MA, Saba T, Javed K, Iqbal T, Rehman A, Ali H, Sarfraz MS (2019) Human behavior analysis based on multi-types features fusion and Von Nauman entropy based features reduction. Journal of Medical Imaging and Health Informatics 9:662–669
5. Dai C, Liu X, Lai J (2020) Human action recognition using two-stream attention based LSTM networks. Appl Soft Comput 86:105820
6. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 770–778
7. F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "Densenet: Implementing efficient convnet descriptor pyramids," *arXiv preprint arXiv:1404.1869,* 2014.
8. Jalal A, Kamal S, Azurdia-Meza CA (2019) Depth maps-based human segmentation and action recognition using full-body plus body color cues via recognizer engine. Journal of Electrical Engineering & Technology 14:455–461
9. A. Kamel, B. Sheng, P. Yang, P. Li, R. Shen, and D. D. Feng, "deep convolutional neural networks for human action recognition using depth maps and postures," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* 2018.
10. Khan SA (2019) Facial expression recognition in unconstrained environment. Shaheed Zulfikar Ali Bhutto Institute of Sciences & Technology, Karachi
11. Khan SA, Hussain S, Xiaoming S, Yang S (2018) An effective framework for driver fatigue recognition based on intelligent facial expressions analysis. IEEE Access 6:67459–67468
12. Khan M, Akram T, Sharif M, Muhammad N, Javed M, Naqvi S (2019) An improved strategy for human action recognition; experiencing a cascaded design. IET Image Process
13. M. A. Khan, M. I. Lali, M. Sharif, K. Javed, K. Aurangzeb, S. I. Haider, *et al.*, "An optimized method for segmentation and classification of apple diseases based on strong correlation and genetic algorithm based feature selection," *IEEE Access,* 2019.
14. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, pp 1097–1105
15. Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) HMDB: a large video database for human motion recognition. In: *2011 International Conference on Computer Vision*, pp 2556–2563
16. J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wild," 2009.
17. Liu AA, Su YT, Nie WZ, Kankanhalli M (2016) Hierarchical clustering multi-task learning for joint human action grouping and recognition. IEEE Trans Pattern Anal Mach Intell 39:102–114
18. Ma CY, Chen MH, Kira Z, AlRegib G (2019) Ts-lstm and temporal-inception: exploiting spatiotemporal dynamics for activity recognition. Signal Process Image Commun 71:76–87
19. G. I. Parisi, "Human Action Recognition and Assessment via Deep Neural Network Self-Organization," *arXiv preprint arXiv:2001.05837,* 2020.
20. Pham HH, Khoudour L, Crouzil A, Zegers P, Velastin SA (2018) Exploiting deep residual networks for human action recognition from skeletal data. Comput Vis Image Underst 170:51–66
21. Rahimi S, Aghagolzadeh A, Ezoji M (2019) "human action recognition based on the Grassmann multi-graph embedding," *Signal*. Image and Video Processing 13:271–279
22. Rashid M, Khan MA, Sharif M, Raza M, Sarfraz MM, Afza F (2019) Object detection and classification: a joint selection and fusion strategy of deep convolutional neural network and SIFT point features. Multimed Tools Appl 78:15751–15777
23. Rish I (2001) An empirical study of the naive Bayes classifier. In: *IJCAI 2001 workshop on empirical methods in artificial intelligence*, pp 41–46
24. Rodriguez MD, Ahmed J, Shah M (2008) Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition. In: *CVPR*, p 6
25. Sharif M, Khan MA, Faisal M, Yasmin M, Fernandes SL (2018) A framework for offline signature verification system: best features selection approach. Pattern Recogn Lett
26. Sharif A, Khan MA, Javed K, Gulfam H, Iqbal T, Saba T et al (2019) Intelligent human action recognition: a framework of optimal features selection based on Euclidean distance and strong correlation. Journal of Control Engineering and Applied Informatics 21:3–11
27. Sharif M, Khan MA, Zahid F, Shah JH, Akram T (2019) Human action recognition: a framework of statistical weighted segmentation and rank correlation-based selection. Pattern Anal Applic:1–14
28. Sharif M, Akram T, Raza M, Saba T, Rehman A (2020) Hand-crafted and deep convolutional neural network features fusion and selection strategy: an application to intelligent human action recognition. Appl Soft Comput 87:105986
29. Sharif M, Attique M, Tahir MZ, Yasmim M, Saba T, Tanik UJ (2020) A Machine Learning Method with Threshold Based Parallel Feature Fusion and Feature Selection for Automated Gait Recognition. *Journal of Organizational and End User Computing (JOEUC)* 32:67–92

30. Siddiqui S, Khan MA, Bashir K, Sharif M, Azam F, Javed MY (2018) Human action recognition: a construction of codebook by discriminative features selection approach. International Journal of Applied Pattern Recognition 5:206–228
31. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.
32. K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402,* 2012.
33. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
34. Tu Z, Xie W, Qin Q, Poppe R, Veltkamp RC, Li B et al (2018) Multi-stream CNN: learning representations based on human-related regions for action recognition. Pattern Recogn 79:32–43
35. Ullah A, Muhammad K, Haq IU, Baik SW (2019) Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments. Futur Gener Comput Syst 96:386–397
36. Wang P, Liu L, Shen C, Shen HT (2019) Order-aware convolutional pooling for video based action recognition. Pattern Recogn 91:357–365
37. Weinland D, Ronfard R, Boyer E (2006) Free viewpoint action recognition using motion history volumes. Comput Vis Image Underst 104:249–257
38. Wu J, Qiu S, Zeng R, Kong Y, Senhadji L, Shu H (2017) Multilinear principal component analysis network for tensor object classification. IEEE Access 5:3322–3331
39. Yang S, Yang J, Li F, Fan G, Li D (2019) Human Action Recognition Based on Fusion Features. In: *The International Conference on Cyber Security Intelligence and Analytics*, pp 569–579
40. Yang H, Yuan C, Li B, Du Y, Xing J, Hu W et al (2019) Asymmetric 3d convolutional neural networks for action recognition. Pattern Recogn 85:1–12
41. Zare A, Moghaddam HA, Sharifi A (2019) Video spatiotemporal mapping for human action recognition by convolutional neural network. Pattern Anal Applic:1–15
42. Zhang J, Shum HP, Han J, Shao L (2018) Action recognition from arbitrary views using transferable dictionary learning. IEEE Trans Image Process 27:4709–4723
43. Zhang P, Lan C, Xing J, Zeng W, Xue J, Zheng N (2019) View adaptive neural networks for high performance skeleton-based human action recognition. IEEE Trans Pattern Anal Mach Intell
44. Zhang HB, Zhang YX, Zhong B, Lei Q, Yang L, Du JX et al (2019) A comprehensive survey of vision-based human action recognition methods. Sensors 19:1005
45. Zhao R, Xu W, Su H, Ji Q (2019) Bayesian Hierarchical Dynamic Model for Human Action Recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 7733–7742

Affiliations

Muhammad Attique Khan[1] · Kashif Javed[2] · Sajid Ali Khan[3] · Tanzila Saba[4] · Usman Habib[5] · Junaid Ali Khan[1] · Aaqif Afzaal Abbasi[3]

[1] Department of Computer Science, HITEC University Museum Road, Taxila, Pakistan

[2] Department of Robotics, SMME NUST, Islamabad, Pakistan

[3] Department of Software Engineering, Foundation University Islamabad, Islamabad, Pakistan

[4] College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

[5] FAST-National University of Computer & Emerging Sciences (NUCES), Chiniot-Faisalabad Campus, Faisalabad, Pakistan