Computers, Materials & Continua
DOI:10.32604/cmc.2021.016864
Article

Tech Science Press

Video Analytics Framework for Human Action Recognition

Muhammad Attique Khan[1], Majed Alhaisoni[2], Ammar Armghan[3], Fayadh Alenezi[3], Usman Tariq[4], Yunyoung Nam[5,*] and Tallha Akram[6]

[1]Department of Computer Science, HITEC University Taxila, Taxila, 47080, Pakistan
[2]College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia
[3]Department of Electrical Engineering, College of Engineering, Jouf University, Sakaka, Saudi Arabia
[4]College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Khraj, Saudi Arabia
[5]Department of Computer Science and Engineering, Soonchunhyang University, Asan, Korea
[6]Department of Computer Science, COMSATS University Islamabad, Wah Campus, 47040, Pakistan
[*]Corresponding Author: Yunyoung Nam. Email: ynam@sch.ac.kr
Received: 14 January 2021; Accepted: 19 February 2021

Abstract: Human action recognition (HAR) is an essential but challenging task for observing human movements. This problem encompasses the observations of variations in human movement and activity identification by machine learning algorithms. This article addresses the challenges in activity recognition by implementing and experimenting an intelligent segmentation, features reduction and selection framework. A novel approach has been introduced for the fusion of segmented frames and multi-level features of interests are extracted. An entropy-skewness based features reduction technique has been implemented and the reduced features are converted into a codebook by serial based fusion. A custom made genetic algorithm is implemented on the constructed features codebook in order to select the strong and well-known features. The features are exploited by a multi-class SVM for action identification. Comprehensive experimental results are undertaken on four action datasets, namely, Weizmann, KTH, Muhavi, and WVU multi-view. We achieved the recognition rate of 96.80%, 100%, 100%, and 100% respectively. Analysis reveals that the proposed action recognition approach is efficient and well accurate as compare to existing approaches.

Keywords: Video analytics; action recognition; features classification; entropy; data analytic

1 Introduction

Action recognition based on human movements has drawn considerable interest due to its emerging applications in video analytics [1,2]. An emerging trend of video labeling for various actions within certain sports such as football, swimming, paragliding, and even in typical daily life movements [3] such as for forensic analysis require recognition that can be made at certain levels of abstraction [4]. Interactive applications [5] already involve human computer interaction in which substantial amount of work has been done covering a broad range of topics [6]. In the literature, most of the works have addressed very specific problems in action recognition. These problems are human-body movement, facial expression, image labeling, and perception of human object iteration [7]. Some authors have also focused on introducing feature selection algorithms for distance-based similarity measures and SVM [8]. Many techniques have been recently introduced for HAR, which may be categorized into graph based, trajectory based, codebook based, feature extraction based [9], to name a few [10]. Wu et al. [11] presented a HAR method with graph based visual saliency and space-time nearest points. Gao et al. [12] introduced a hypergraph-based method to compute the distance between two objects at a multiview scenario. In these methods, vertices and edges of the objects are defined in cluster view. Edges join multiple points and weights are assigned to each edge based on their relationships between any two views in the group. Yi et al. [13] introduced trajectory based HAR. This method solves the problem of motion information between distinct motion regions. The method makes use of the trajectory based covariance features and performs better as compared to Histogram of oriented gradients and its variants. Althloothi et al. [14] presented HAR based technique based on motion and shape features, extracted using spherical harmonics.

In [15] introduced a new feature referred to as local surface geometric feature (LSGF). The LSGF features for human body posture and expression are extracted to be utilized further in the covariance matrix and feature vectorization. Chen et al. [16] presented depth motion maps (DMM). This method consists of four major steps such as depth map generation, features extraction by utilizing DMM, features reduction and recognition. The PCA is used for the dimensionality reduction and provides improved efficiency in recognition. The few other methods are 16-layers CNN [17], fusion of features [18], weighted segmentation based approach [19], fusion of deep and handcrafted [19], and name a few more [20].

However, most of the recent contributions based on features selection have not addressed frame enhancement, which we believe is a crucial step in making the foreground object more visible. For instance, the optical flow algorithm, proposed in [21] fails to segment the foreground object due to low-resolution videos and variation in motion speed. Similarly, various feature selection and extraction techniques, such as [22], do not consider optimization of the local and global features, which usually lead to lower classification accuracy [23]. We believe that a sound feature enhancement technique coupled with efficient features optimization mechanism would result in increased classification accuracy. In order to achieve greater classification accuracy, a novel framework has been proposed that implements segmented frames fusion and Entropy-Skewness (ES) based features reduction. In what follows we enumerate the primary contributions of the proposed work, which also describes our research methodology in order:

- Construction of an enhanced HSI color space, utilizing a hybrid color transformation technique, which incorporates refinement of RGB channels, bottom-hat filtering, and NTSC transformation.
- An implementation of novel maximum regions based segmentation technique in which pixels' fusion has been performed, using the proposed saliency mapped frame.
- Extraction of a hybrid set of features and their dimension reduction by using the entropy skewness control method.
- Construction of a feature codebook having a size of 1×470, using serial based features fusion. This is followed by an implementation of a genetic algorithm for prominent features selection. The selected features have a dimension of 1×210.

Finally, an extensive experimentation and comparison has been performed between the proposed and existing methods by implementing two use cases.

2 Proposed Framework

The proposed architecture consists of five major steps: a) Frame enhancement using new series of steps; b) introduction to maximum regions based segmentation technique with an integration of frames fusion with novel saliency map; c) extraction of texture, local, and global features using SFTA, LBP, and HOG; d) a novel features reduction technique is implemented based on Entropy-Skewness (ES) control method and then serial based feature fusion is performed for the construction of features codebook having size 1 × 470; e) implementation of a custom made genetic algorithm for the selection of most optimal features prior to multi-class SVM for final classification. Fig. 1 show the detail of proposed method.

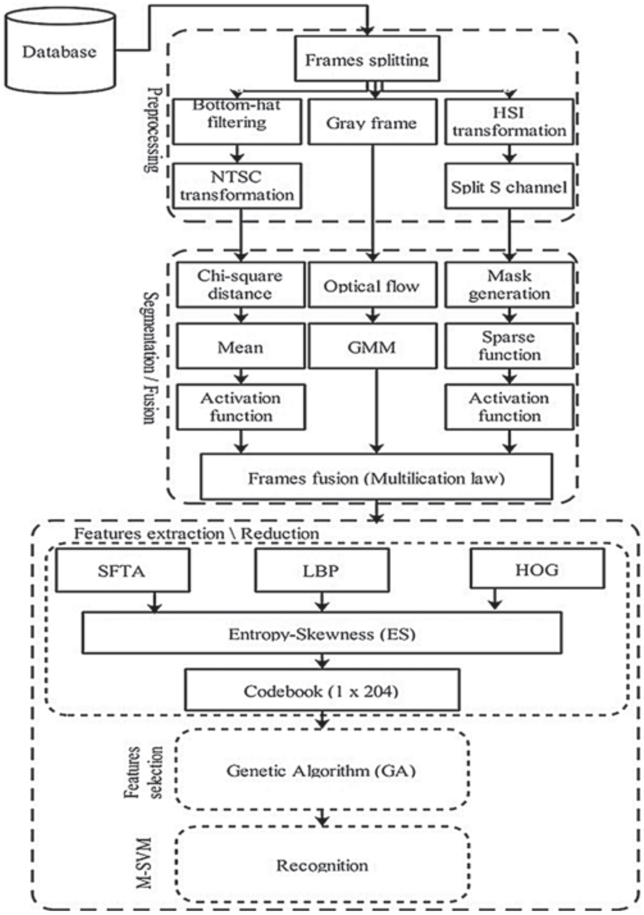


Figure 1: System architecture of the proposed action recognition approach

2.1 Preprocessing

Foreground visibility is a major issue in this area which is addressed in this section. Frame enhancement is an important preprocessing step for an accurate segmentation of foreground objects because we are dealing with raw input video data [24]. This data contains many distorted, noisy, and dull kind of images (weak edges and fused boundaries). To get improved images and quality information, we need to enhance these frames to get our desired results, and this is the main motivation behind our frames' enhancement approach. Also, in few recent studies, an optical flow algorithm failed to segment the foreground object due to low-resolution videos. To handle this kind of problem, a new technique is implemented named as (BHNT SC–S) which incorporates two fundamental steps as bottom-hat filtering and color space transformations. The complete process is performed in parallel. Firstly, a conventional RGB frame is enhanced with bottom-hat filtering, which is subsequently utilized in the segmentation phase.

The fusion relation between the bottom-hat filter and NTSC transformation frame is given as; Let $\varphi^{I}(x,y)$ represents an original RGB frame having dimensions 256 × 256 × 3. The bottom hat-filtering technique is implemented on $\varphi^{I}(x,y)$ to enhance the brightness of the foreground object and reduce the background contrast with respect to black pixels. The bottom-hat filtering technique effectively works on tiny objects on a scattered background as follows:

$$\widetilde{\varphi bot} = \varphi^{I} \cdot St - \varphi^{I} \tag{1}$$

where $\widetilde{\varphi bot}$ represents the bottom-hat frame, St represents the structuring element, which is initialized as 9 and $\cdot$ is the closing operation. Then NTSC transformation is performed by utilizing $\widetilde{\varphi bot}(x,y)$ to make the foreground object more visible. The NTSC transformation is performed as follows:

$$\varphi^{R} = \frac{\varphi^{R}}{\sum_{l=1}^{3}\varphi^{j}}, \quad \varphi^{G} = \frac{\varphi^{G}}{\sum_{l=1}^{3}\varphi^{j}}, \quad \varphi^{B} = \frac{\varphi^{B}}{\sum_{l=1}^{3}\varphi^{j}} \tag{2}$$

where $l = [123]$, and represents an index for three channels of red, green, and blue, respectively. φ^{R}, φ^{G}, φ^{B} are modified red, green, and blue channel, respectively. The green channel is utilized for the gaussian mixture model (GMM) segmentation.

$$\varphi^{\widetilde{YIQ}} = \begin{bmatrix} \tilde{\varphi}^{Y} \\ \tilde{\varphi}^{I} \\ \tilde{\varphi}^{Q} \end{bmatrix} = \begin{bmatrix} \varphi^{Y^*} \times \varphi^{I^*} \times \varphi^{Q^*} \end{bmatrix} \begin{bmatrix} \widetilde{\varphi^{R}} \\ \widetilde{\varphi^{G}} \\ \widetilde{\varphi^{B}} \end{bmatrix} \tag{3}$$

where $\varphi^{\widetilde{YIQ}}$ is the NTSC frame. The enhanced NTSC frame is improved with Gaussian function and is further utilized for novel saliency segmentation. Finally, HSI transformation is performed on $\varphi^{I}(x,y)$ for maximal region segmentation as shown in Fig. 1. The visibility results are tested on each channel, however, the saturation channel produced better results. Hence, we select saturation channel for maximal region segmentation. The saturation channel is defined as $\widetilde{\varphi^{S}} = 1 - \frac{3}{\sum_{j=1}^{3}\varphi^{j}} \times \alpha$, where $\alpha 1 = \min(\widetilde{\varphi^{R}}, \widetilde{\varphi^{G}}) \cdot \widetilde{\varphi^{S}}$ channel is set as input for maximal region segmentation. The results of the preprocessing step are shown in Fig. 2. In this figure, it is showing that the original frames are initially processed in the green channel and then followed the bottom hat filtering and NTSC transformation. After that frames are reconstructed and get a saturation frame for further process.

2.2 Frames Segmentation

In this section, we segment the foreground objects for identification of their activities. The optical flow algorithm has been used for identification of motion regions in the frame. We then construct a novel saliency method, which is fused with a new maximal region segmentation technique. The optical flow algorithm is executed in parallel with the novel saliency method as shown in Fig. 1. The purpose of frames fusion is to obtain maximum accuracy and reduce the error rate.

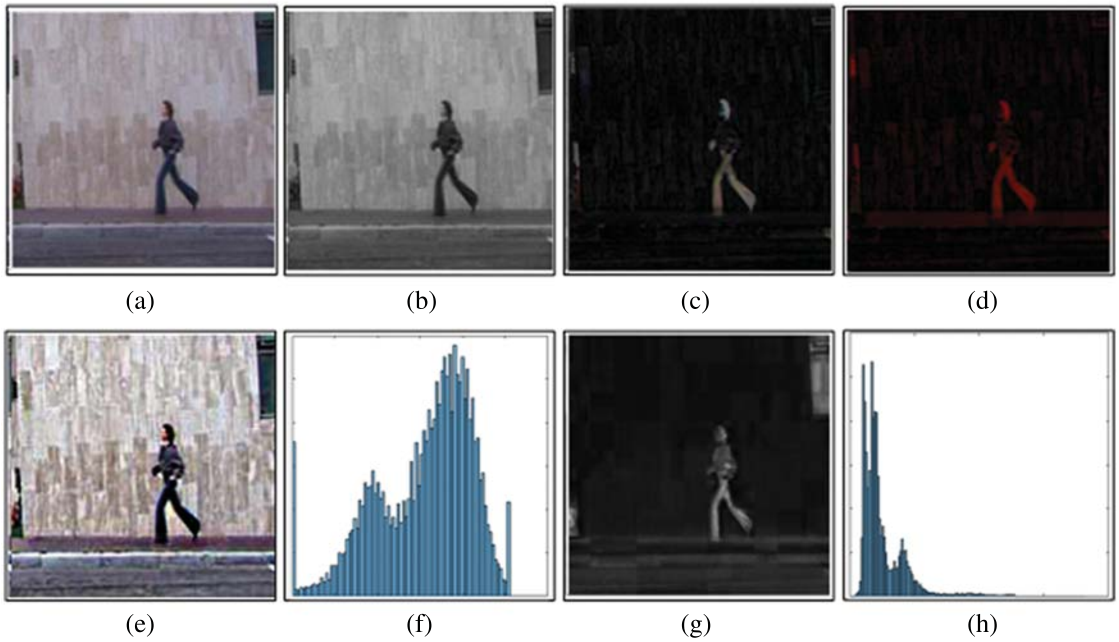


Figure 2: Effects of preprocessing step. (a) Original frame; (b) green channel; (c) bottom hat filter frame; (d) NTSC frame; (e) enhanced frame after reconstruction; (f) combined histogram of reconstructed frame (RGB); (g) saturation frame; (h) histogram of saturation frame

Saliency map: Let ψ represents optical flow function having three parameters for horizontal, vertical, and time ion (h, v, t) and $\varphi^{\widetilde{YIQ}}$ represents a 3-dimensional enhanced frame. ψ executes in parallel with $\varphi^{\widetilde{YIQ}}$ to give motion information of a foreground object in the current frame. A chi-square distance function is performed on the resultant frame to calculate distance between the motion pixels. The motion pixels with minimum distance are considered as a salient object and pixels with maximum distance represent the background. The chi-square distance is calculated as follows:

$$Xd^2 = \sum_{i=1}^{N} \frac{(\varphi i - E(\varphi i))}{\sigma^2} \tag{4}$$

$$T = \begin{cases} Salient & if\ \ Xd^2 = minimum \\ Background & if\ \ Xd^2 = maximum \end{cases} \tag{5}$$

where, T represents a selection of a salient object and the background. If $Xd^2 = minimum$, the chi-square distance between pixels is minimum which in turn labels it as a salient object, otherwise it is considered as background. Then color features of a salient object are extracted, which are effective for saliency estimation. RGB and LAB color spaces are used for features extraction and mean, range and entropy are calculated for each channel. The cumulative mean and standard deviation are calculated for the color frame. The mean value is used as a threshold value for frame binarization and the centered value of the frame is computed by σ as follows:

$$center = \frac{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{M}(q_j - p_j)^2 + (q_i - p_i)^2}}{2\sigma^2} \tag{6}$$

The center value is subtracted from the color image and a new mapped frame is obtained as follows: $\varphi(map) = \varphi_i(color) - Center$

We then perform an activation function to remove noise in the salient frame and make the object more visible.

$$F(R) = \begin{cases} \max\limits_{j\epsilon N} & if \ \dfrac{1}{|H|}\sum\limits_{j\epsilon N}\varphi(map) \geq 1 - \mu \\ \min\limits_{j\epsilon N} & if \ \dfrac{1}{|H|}\sum\limits_{j\epsilon N}\varphi(map) \leq \mu \\ \varphi(map) & otherwise \end{cases} \tag{7}$$

where H denotes the number of neighbor pixels and μ is mean of the mapped frame. The noise removal function F(R) is performed on the mapped frame $\varphi(map)$ to get a new improved salient frame. The improved salient frame is defined as:

$$Im(sal) = F(R)[\varphi(map)] \tag{8}$$

where, $Im(sal)$ represents the improved salient frame. The graphical sample results are shown in Fig. 3.

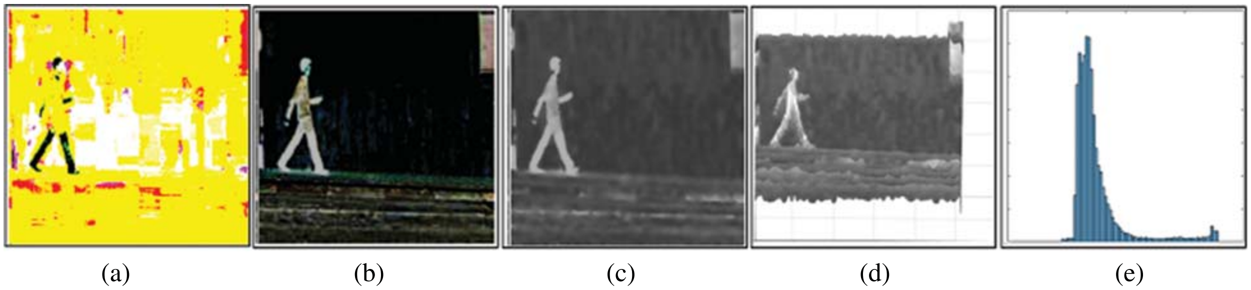


Figure 3: Sample results of a mapped frame (a) color frame; (b) mapped frame; (c) Noise removal frame; (d) 3-D constructed frame; (e) histogram of the noise removal frame

Finally, we set a threshold function to obtain the binary image as follows:

$$Fin(sal) = \begin{cases} 1 & if \ Im(Sal) > \mu \\ 0 & if \ Im(Sal) < \mu \end{cases} \tag{9}$$

where, μ denotes the cumulative mean value, which is computed from the color frame. Some morphological operations are performed to remove extra regions from the segmented frame. The saliency-based method is described in Algorithm 1. The results are shown in Fig. 4.

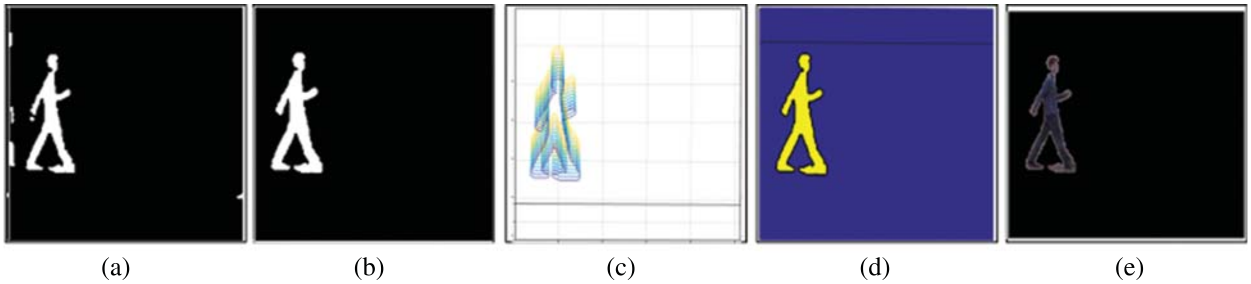


Figure 4: Final saliency results. (a) Initial salient frame; (b) morphological operations frame; (c) 3-D constructed frame; (d) contour frame; (e) mapped on original frame

Segmentation Based on Maximum Region Extraction: The maximum extraction region has two primary steps. Firstly, a mask of input saturation channel $\tilde{\varphi}^S$ is generated and secondly, a threshold value is obtained automatically using object magnitude for most significant regions in the masked frame. Later, few morphological operators are utilized to remove unused regions.

Algorithm 1: Saliency Estimation

```
Input: Motion vector $\psi$ and $\widetilde{\varphi^{YIQ}}$ frame.
Output: Saliency frame $Fin(sal)$.
$i = 1$
$N =$ Number of pixels
Step 1: For $i = 1 : N$
        {
Step 2: Calculate Chi-square distance $Xd^2$ by using Eq. (4).
        Set threshold for minimum distance.
Step 3: Extract color features.
Step 4: $\mu$ and $\sigma$ is calculated for center value.
Step 5: Initial map is constructed by Eq. (7).
        Perform activation function by Eq. (8).
Step 6: Final saliency map is constructed by Eqs. (9) and (10).
        }
Step 7: END
```

Mask generation: In mask generation of the saturation frame $\widetilde{\varphi^S}$, we create a Zero matrix of size 256 × 256 and set condition up to 1 as follows:

$$\varphi\,(mask) = \sum_{i=1}^{p}\sum_{j=1}^{q}(p-i)\,(q-j) = 1 \tag{10}$$

where p and q denote the number of pixels in one frame. We then set a dynamic threshold value and store the pixel value of the extracted object in the *Zero matrix*. The threshold is set as follows:

$$\varphi^{v}(\mathbb{T}) = \begin{cases} 1 & if \ \widetilde{\varphi^{S}} \times \varphi(mask) \geq T_1 \\ 0 & if \ \widetilde{\varphi^{S}} \times \varphi(mask) < T_1 \end{cases} \tag{11}$$

where $\varphi^{v}(\mathbb{T})$ is the threshold frame, T_1 is the threshold value, which is automatically selected depending on the object magnitude. Further closing and filling operations are performed to make the segmented image more accurate. Algorithm 2 explains the segmentation process based on the maximum region extraction. The sample results are shown in Fig. 6d.

Frames Fusion: Frames fusion corresponds to the process of combining comparative information of two frames into a single frame. The fused frame is more accurate with respect to parent frames and contains comprehensive information compared to any single segmented frame. In this article, we implemented a novel frame fusion technique based on similar pixel values as shown in Fig. 5. The proposed fusion technique is simple but more effective as compared to above listed approaches. The fusion process follows the additive law of probability which overcomes the problem of over smoothness/over sharpness and provides the statistically balanced segmented frames. Moreover, enhancement procedure strengthens the weak edges that lead to an appropriate segmentation having clear boundaries.

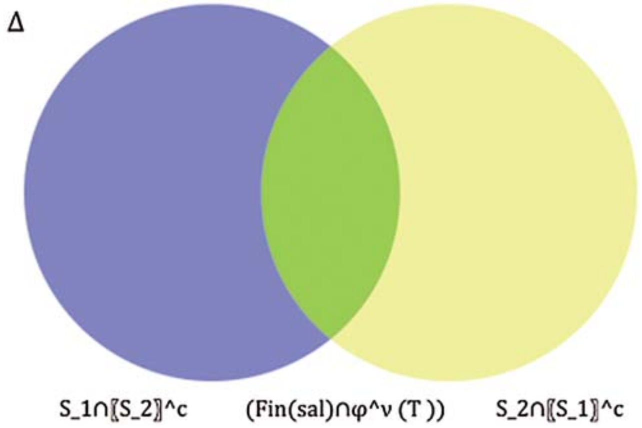


Figure 5: Representation of the fusion of two frames based on their similar pixels values

Let Δ denotes the all pixel's values of both segmented frames. Let s_1 denotes the pixel values of saliency frame $Fin(sal)$, s_2 denotes the pixel values of maximal region segmentation frame $\varphi^{v}(T)$, and s3 denotes the common points of *fin* and $\varphi^{v}(\mathbb{T})$. The frames fusion is computed as follows:

$$\begin{aligned} \xi\left(Fin(sal) \cap \varphi^{v}(\mathbb{T})\right) &= \xi\left(s_1 \cap s_2^{c}\right) - \xi(s_1) \\ &= \frac{s_3}{n} \times \frac{s_1}{s_1}, \ = \frac{s_1}{n} \times \frac{s_3}{s_1}, \ = Fin(sal) \times \xi(s_2 \mid s_1) \end{aligned} \tag{12}$$

$$\varphi(fused) = Fin(sal) \times \frac{\xi(s_2 \cap s_1)}{\xi(s_1)} \tag{13}$$

where ξ denotes the number of occurrences of frame pixels, $\cap$ denotes the common pixels between two segmented frames, c denotes the complement of segmented frames and $\varphi\,(fused)$ is fused segmented frame. The detailed segmentation and fusion results are shown in Fig. 6, which demonstrates the value of the fusion method.

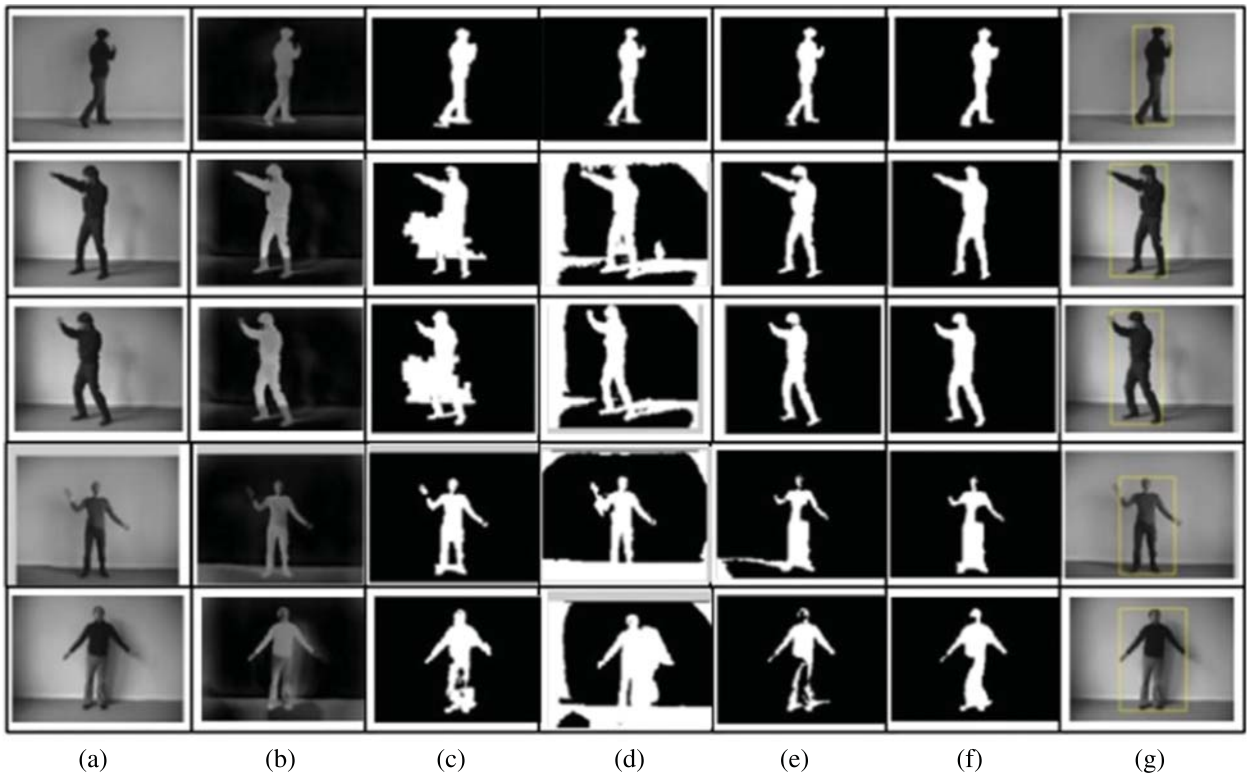


Figure 6: Segmentation and ROI detection results using KTH dataset. (a) Original frame; (b) initial mapped frame; (c) saliency frame; (d) maximal region frame; (e) refined frame of (d); (f) fused frame; (g) ROI detection frame

Algorithm 2: Segmentation based on maximum region extraction

```
Input: $\tilde{\varphi}^S$ frame.
Output: Threshold frame $\varphi^v(\mathbb{T})$
$Z \leftarrow Totalframes$
Step 1: $For\ i = 1 : Z$
        {
Step 2: Calculate mask frame $\varphi(mask)$ using Eq. (11)
Step 3: Calculate $\widetilde{\varphi^v(\mathbb{T})}$ using Eq. (12) prior to morphological operations.
Step 4: Perform morphological operations such as closing and filling.
        }
Step 5: END
```

2.3 Feature Descriptors

Feature extraction is very important for representation of an object [25]. In this section, we are dealing with raw video data, which possibly contains faces, texture coatings, background objects, etc., with a variety of artificial makeup. To deal with these assortments we need to use a combination of features. Shape, SFTA, and LBP descriptors are extracted. The grayscale and the proposed fused segmented frames $\varphi\,(fused)$ are used in the feature extraction phase. The SFTA features are extracted in three steps. Firstly, the fused segmented frame $\varphi\,(fused)$ is used to make the set of binary frames. Secondly, the fractal features are calculated by using 8 neighborhood pixels.

Finally, we calculate the mean (μ) and size (pixels) of the segmented frame. By using 8 neighborhood pixels, a 1 × 45 dimensional feature vector (FV) is obtained. For LBP features extraction, binary code is calculated for each pixel in the frame and compares it whether the intensity value of the pixel is greater or less than the current pixel intensity. Then a histogram is computed to count the number of occurrences of each binary code. The LBP features are defined as: $\varsigma\,(LBP) = \sum_{i=0}^{n} 2^m s(g_p - g_c)$ where $n = 7$, m runs over 8 neighbors of the central pixels g_c and $s(u)$ is: $s(u) = \begin{cases} 1 & u \geq 0 \\ 0 & otherwise \end{cases}$. The final LBP FV has dimensioned 1 × 59 for each frame, which is later utilized for fusion. Finally, the HOG features are extracted from fused segmented frame and obtained a vector of size 1 × 3780. Later on, the proposed features reduction technique, entropy skewness, is implemented on these features.

Features Reduction using Entropy Skewness: A large number of features negatively hits the accuracy and increases computational time of the system [26,27]. The PCA is used in literature for dimensionality rebate/reduction. In this article, we compare our proposed features reduction technique with PCA in terms of five performance measures. The workflows of ES methods are shown in Fig. 7. The same size feature is used to analyze the information on the same dimensional frames to obtain the high similarity index before subjecting to the classification phase. For the proposed method, entropy and skewness value is calculated for all three types of extracted features. The entropy value for one frame features is calculated as follows:

$$E\,(\varsigma) = -\sum_{i=1}^{F_t} P_{u_i} \log b P_{u_i} \tag{14}$$

where F_t denotes the total number of extracted features for one frame, P denotes the probability of occurrences of features and $b = 10$. Similarly, the mean and standard deviation are calculated for each frame feature for skewness value. The skewness value is computed by mean and SD, that are defined as: $\mu = \sum_{i=1}^{F_t} \frac{u_i}{F_t}$ and $\sigma = \sqrt{\left(\frac{u_i}{F_t}\right)^2 - \left(\frac{u_i}{F_t}\right)}$ Hence,

$$S\,(\varsigma) = \frac{t}{(t-1)\,(t-2)} \sum \left(\frac{u_i - \mu}{\sigma}\right) \tag{15}$$

where μ, σ, and S denote the mean, standard deviation, and skewness of extracted features, respectively. Then we add both entropy and skewness values as:

$$\Upsilon\,(+) = \sum_{i=1}^{F_i} (S\,(\varsigma_i) + E(\varsigma_i)) \tag{16}$$

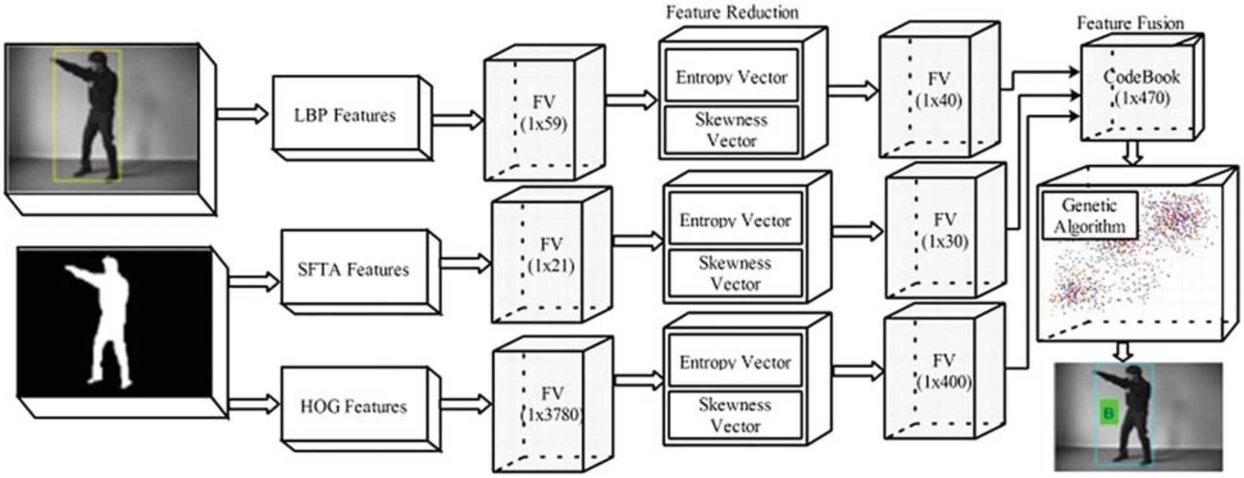


Figure 7: Workflow of the proposed entropy skewness based reduction and selection

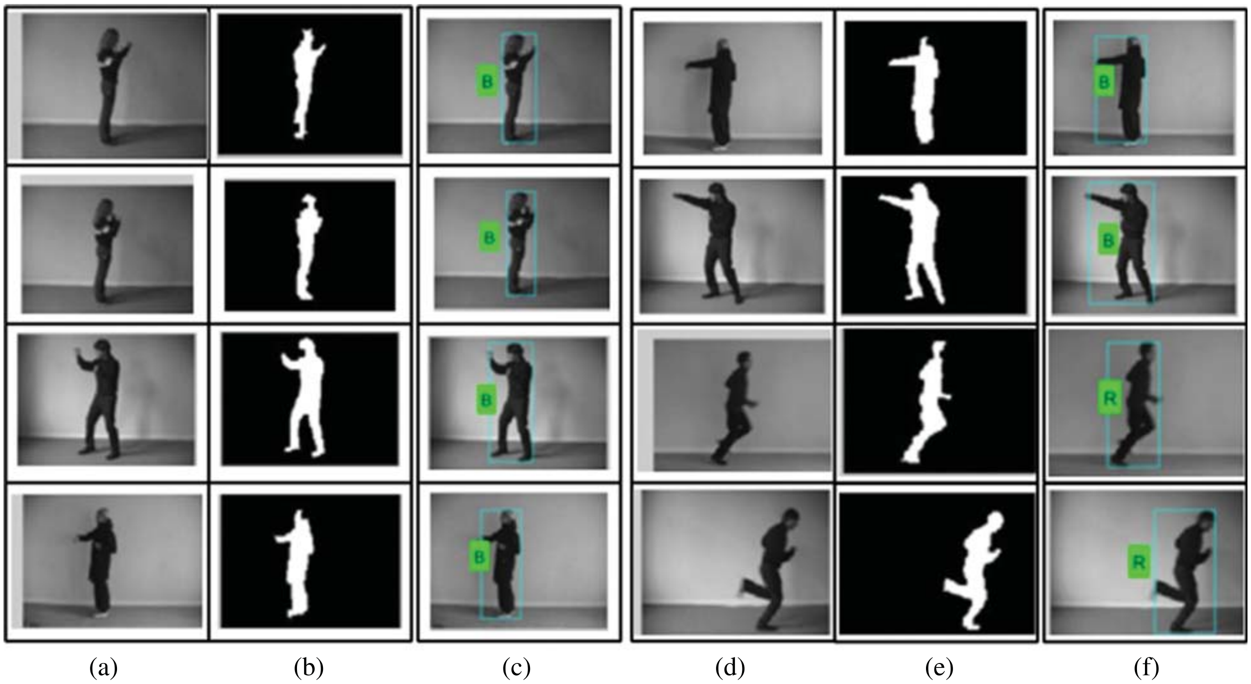
Finally, 40, 30, and 400 features are selected from LBP, SFTA, and HOG respectively based on their mean value. The features are reduced to a value which is less than the mean value of $\Upsilon(+)$. Then the remaining features are fused by serial-based fusion to build a codebook having dimensions of 1×470. The serial-based fusion is simple but more effective. Let $\varsigma(HOG)$, $\varsigma(LBP)$ and $\varsigma(SFTA)$ be three extracted feature vectors having dimensions 1×400, 1×40, and 1×30, respectively. Then these features are added as:

$$\varphi(CB) = \sum_{n=1}^{NF_t} \{\varsigma(HOG), \varsigma(LBP), \varsigma(SFTA)\} \tag{17}$$

Finally, we get the codebook of size $\varphi(CB) = \{1 \times 470\}$. The constructed feature codebook is optimized by a custom-made genetic algorithm (GA) and selects the best features for action recognition.

Features Selection: The feature selection is performed on the fused vector $\varphi(CB)$ in order to identify most relevant and uncorrelated feature data. For best features selection, we opted genetic algorithm which has the tendency to handle larger space problems even when the objective function is stochastic. In our proposed work, the input to the genetic algorithm is extracted codebook $\varphi(CB)$ of size 1×470 whereas; the optimized features are the output, given to the classifier. Mainly, the GA is comprised of the standard steps of population initialization, fitness calculation, crossover, mutation and finally selection. Amongst several existing crossover techniques, we opted for uniform crossover technique having crossover rate of 0.7. The $\varphi c = CrosOver(y_1, y_2)$, where $y_1 = \alpha \times x_1 + (1-\alpha) \times x_2$ and $y_1 = \alpha \times x_2 + (1-\alpha) \times x_1$ is crossover. The x_1, x_2 are selected parents. In the mutation, a uniform approach is applied of rate 0.1. For selection, we adopted the roulette wheel and defined as: $P = \frac{p_1}{\sum(p_1)}$ where $p_1 = \exp\left(-\beta_1 \times \frac{S_p}{W_c}\right)$, S_p is sorted population W_c is the last number of population. The β_1 is selected for parent pressure, which is set to be 8. For our proposed case, the fitness function is defined to be a mean of chromosome as $\widetilde{\varphi_{Ft}} = mean(Cm)$. In our case, this function guarantees the optimized solution. The newly generated feature sets are used in the classification phase. In The classification phase, Multi-Class SVM is employed for final features classification. The labeled results are showing in Fig. 8.

```
**Algorithm 3:** Features selection
**Input:** Codebook of size $1 \times 470$.
**Output:** Optimized feature vector of size $1 \times 210$.
**Step 1**: Initialize GA parameters:
$M_1 \leftarrow 1000$
$N_p \leftarrow 10$
$C_p \leftarrow 0.7$
$M_R \leftarrow 0.1;$
$\beta_1 \leftarrow 8$
$t \leftarrow 0$
$P_{Op0} \leftarrow$ Initialize with population size $\Phi^{N_p}$
Evaluate $P_{Op0}$
**Step 2:** *For* $t < M_1$
**Step 3:** *Parents* $(X_{par}) \leftarrow$ Select $X_{par}$ for $P_{Opt}$.
**Step 4:** *Offspring* $(X_{off}\ ) \leftarrow$ Crossover $(X_{off}\ (C_{prob}))$
**Step 5:** Mutation $(X_{off},\ M_{prob})$
**Step 6:** Evaluate $X_{off}$
**Step 7:** $P_{Opt+1} \leftarrow$ actual population through replacement
**Step 8:** $P_{Opt}$ & $X'_{off}$
$t = t + 1$
**Step 9: END**
```



(a) (b) (c) (d) (e) (f)

Figure 8: Proposed action recognition results using Weizmann dataset: (a) Original frame; (b) segmented fused frame; (c) recognized frame; (d) original frame; (e) segmented fused frame; (f) recognized frame

3 Experimental Setup and Results

The computational complexity of the proposed framework is linearly dependent on the input. For each pixel, $N^2r^2q^2(n_1^2+n_2^2$, where N^2, q^2, and r^2 are represents mass of input, search window, and patch respectively. This statement connects the total steps and operations perform in this work. The sum $n_1^2+n_2^2$ represents total required operations during the fusion step.

3.1 Selected Datasets

Weizmann dataset: Weizmann dataset [28] is considered a flexible and comprehensive action recognition dataset. This dataset has been built in an indoor environment and contains a total of 90 videos. There are 10 classes of different actions which are described in Tab. 1. Every action is performed by 9 actors in each class.

KTH dataset: The KTH action dataset [28] includes a total of 599 videos of 6 action classes, which are described in Tab. 1. Each action class is completed by 25 actors in four distinct situations like outdoors, scale variation, in outdoors, outdoors with distinct clothes and lighting variations in indoors.

Muhavi dataset: The Muhavi action dataset [28] involves a total of 17 actions and each action is completed by 14 persons. Eight cameras are located on different views for the recording of human actions. A total of 10 actions are considering in this work for classification, depicted in Tab. 1.

WVU Multi-view dataset: The WVU multi-view action dataset [28] consists of total of 780 action videos. This dataset consists of 12 human actions and every action is performed by 2 persons. Eight different view cameras are located for human action recording. Tab. 1 depicts the selected action for classification.

Table 1: Description of action classes of the selected datasets. The L denotes class label

Weizman		Muhavi		WVU action		KTH	
Action	**L**	**Action**	**L**	**Action**	**L**	**Action**	**L**
Bending	B	Climp ladder	L	Clapping	C	Boxing	B
Jack	K	Crawlon knees	N	Jogging	G	Clapping	C
Jumping	J	Drunk walk	U	Jack	K	Hand wave	H
One hand wave	O	Graftee walk	G	Kick	I	Jogging	G
Two hand waves	T	Jump over fence	J	Node head	N	Running	R
Jumping place	P	Jump over gap	O	One hand	O	Walking	W
Running	R	Kick	K	Punch	P	–	–
Skipping	S	Punch	P	Standing still	S	–	–
Slide one way	D	Run stop	R	Throwing	T	–	–
Walking	W	Walk turn back	B	Two hand wave	V	–	–

3.2 Evaluation Methods

The proposed framework is validated on four large action datasets: Weizmann, KTH, Muhavi, and WVU. The selected action classes and their respective class labels are depicted in Tab. 1. To assess the proposed method performance, 10-fold cross-validation is made on all three datasets.

The MC-SVM is used for action recognition and we compare their performance with eight other classification algorithms: Fine-KNN, weighted-KNN, ensemble boosted tree (EBT), subspace discriminant analysis, DT, QDA, logistic regression, and Q-SVM. To measure the authenticity of the proposed algorithm, we implement five statistical measures of accuracy: FNR, precision, sensitivity, FPR, and correct recognition rate (CCR). The proposed performance is compared with PCA based features reduction model and then a comparison is made with existing methods. MATLAB 2019b based simulations are carried out on a personal computer.

3.3 Results and Discussion

The proposed framework is workflow of five major steps such as preprocessing, segmentation of ROI, features extraction and reduction, features selection, and recognition whereas each step is series of sub-steps as shown in Fig. 1. The proposed framework is evaluated in two stages: a) Features reduction has been carried out by PCA which is then sent to MC-SVM for recognition; b) features reduction is performed by the novel ES method and then GA base selected features are provided to MC-SVM for recognition. The detailed description of each of these modules is given in Fig. 7. Four publicly available datasets, namely, Weizmann, KTH, Muhavi, and WVU multi-view are selected for evaluation. For testing and training 50:50 strategy is adopted. A comprehensive comparison of the proposed algorithm is performed with eight classifiers and their performance is evaluated by five measures such as sensitivity, precision, FPR, FNR, and CRR. Additionally, we also compare our proposed method with existing works on the selected datasets just to support our claim of achieving best accuracy even with the most recent articles.

Fig. 9 summarizes the results of features reduction by PCA and muti-class SVM. The multi-class SVM achieved best recognition results of 91.7%, 98.9%, 99.8%, and 99.90% on Weizmann, KTH, Muhavi, and WVU muti-view dataset, respectively. Moreover, the average recognition execution time of PCA based reduction approach for selected datasets is 51.729 s.

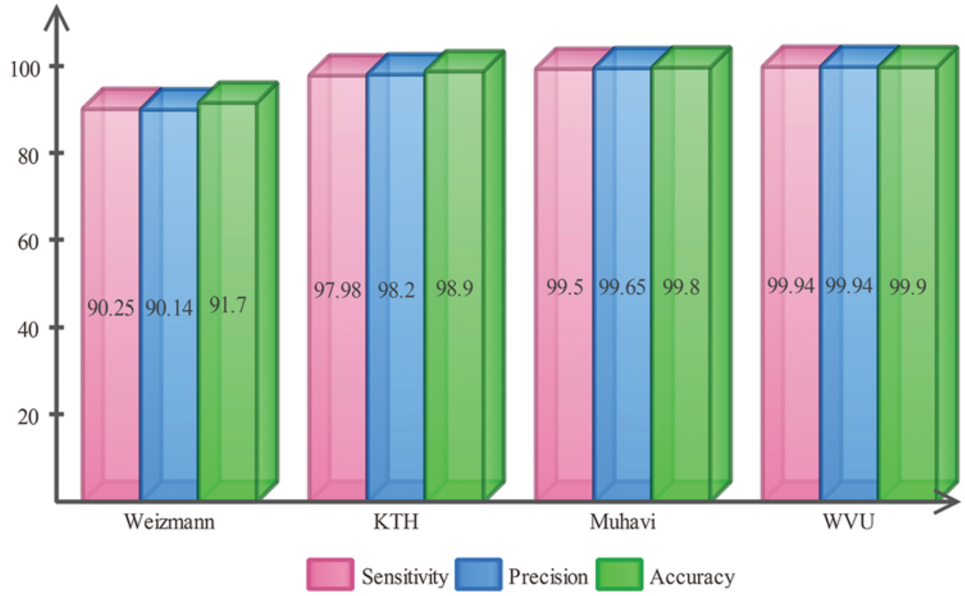


Figure 9: Classification results using PCA based reduction approach

The proposed ES based features reduction and the GA based features selection results are shown in Tab. 2. It is evident that the proposed method achieved best recognition results of 96.80%, 100%, 100%, and 100% on Weizmann, KTH, Muhavi, and WVU multi-view datasets, respectively. The recognition rate of the proposed method is explained by the confusion matrix in Tab. 3. The selected classifiers such as W-KNN, Q-SVM, F-KNN, and QDA also achieved maximum recognition rate of 100% on the WVU multi-view dataset. The average recognition execution time for the proposed ES based reduction and GA based selection is 23.901 s, which is significantly lower as compared to PCA.

Table 2: Proposed results using entropy-skewness (ES) based features reduction and GA based features selection

Classifier	Dataset				Performance measures				
	Weiz	KTH	Muh	WVU	FPR	Sen (%)	Pre (%)	FNR (%)	CRR (%)
Fine K-nearest neighbor	✓				0.007	91.67	92.07	6.7	93.3
		✓			0.001	99.60	99.65	0.2	99.8
			✓		0.001	99.80	99.80	0.2	99.8
				✓	**0.000**	99.88	99.88	0.1	**100**
Weighted K-nearest neighbor	✓				0.012	86.05	87.21	11.0	89.0
		✓			0.005	98.70	98.9	1.0	99.0
			✓		0.001	99.50	99.70	0.2	99.8
				✓	0.001	99.76	99.82	0.1	**100**
Ensemble boosted tree	✓				0.004	94.85	95.17	4.1	95.6
		✓			0.009	97.60	97.90	2.0	98.0
			✓		0.002	99.30	99.40	0.4	99.6
				✓	0.003	99.17	99.23	0.6	99.4
Subspace discriminant analysis	✓				0.016	81.97	83.22	14.2	85.8
		✓			0.011	94.70	95.10	4.7	95.3
			✓		0.003	99.10	99.30	0.6	99.4
				✓	0.007	97.17	97.21	2.7	97.3
Decision tree	✓				0.021	76.81	78.24	19.6	80.4
		✓			0.030	76.93	78.33	15.8	84.2
			✓		0.007	92.10	92.30	7.6	92.4
				✓	0.014	95.09	95.03	4.9	95.1
Quadratic discriminant analysis	✓				0.022	77.98	78.20	21.3	78.7
		✓			0.014	89.86	90.91	7.1	92.9
			✓		0.013	89.70	89.70	10.2	89.8
				✓	**0.000**	99.92	99.96	0.1	**100**
Logistics regression	✓				0.005	92.82	93.17	5.9	94.1
		✓			0.009	97.10	97.35	2.0	98.0
			✓		0.001	99.70	99.75	0.2	99.8
				✓	0.003	99.10	99.27	0.4	99.6
Quadratic support vector machine	✓				0.008	90.35	91.86	8.0	92.0
		✓			0.002	99.10	99.21	0.4	99.6
			✓		**0.000**	99.50	99.55	0.4	99.6
				✓	**0.000**	99.94	99.96	0.1	**100**
Proposed (multi-class SVM)	✓				0.003	**96.61**	**96.60**	**3.40**	**96.80**
		✓			**0.000**	**99.78**	**99.80**	**0.1**	**100**
			✓		**0.000**	**99.80**	**99.85**	**0.1**	**100**
				✓	**0.000**	**100**	**100**	**0.0**	***100***

Table 3: Confusion matrix of the proposed approach

Weizman dataset										
Class	**Classification class**									
	B	D	J	K	O	P	R	S	T	W
B	**100%**									
D		**99.0%**	1.2			2.6				
J		0.3	**87.7%**			0.3	1.0	0.3	0.2	2.2
K		0.3	0.2	**100%**		0.6				
O			0.2		**100%**			0.3		0.2
P			0.5	0.2		**96.5%**				
R		0.3	2.5				**95.3%**			2.0
S			1.7	0.2			3.3	**94.5%**		1.5
T			0.7						**100%**	
W			5.1				0.3	0.5		**93.1%**

KTH dataset						
Class	B	C	H	G	R	W
B	**100%**					
C		**98.8%**			0.2%	
H			**100%**			
G		1.1%		**100%**	0.1%	
R		0.1%			**100%**	
W						**100%**

MUHAVI dataset										
Class	L	N	U	G	J	O	K	P	R	B
L	**100%**									
N		**100%**								
U			**100%**							
G				**100%**						
J					**100%**					
O						**100%**				
K							**100%**			
P								**99%**	0.1	
R								0.1	**99%**	
B										**100%**

WVU dataset										
Class	C	G	K	I	N	O	P	S	T	V
C	**100%**									
G		**100%**								
K			**100%**							
I		0.3		**100%**						
N					**100%**					
O						**100%**				
P							**100%**			
S								**100%**		
T									**100%**	
V										**100%**

Finally, the proposed method results are compared with existing HAR methods for all selected datasets as given in Tab. 4. In this table, the proposed method is evaluated on Weizmann dataset and achieved recognition accuracy of 96.80%, that when compared with existing approaches such as [29] shows improved performance. Secondly, the proposed recognition accuracy on the KTH dataset is 100%, that is quite good performance as compared to [30]. Similarly, the recognition performance for the proposed algorithm on WVU and Muhavi datasets is 100%, that is significantly robust as compared to [31,32]. From the experimental results, this is quite evident that the proposed feature selection approach performs better as compared to PCA based feature selection. It is noted that our proposed algorithm outperforms existing techniques in terms of recognition rate. The visual results are shown in Fig. 8, where we can accurately observe the binary results and in turn get the most accurate label.

Table 4: Comparison of proposed algorithm with recent techniques using selected datasets

Weizmann dataset		
Method	Description	Accuracy (%)
Xiao et al. [29]	Dynamic framework	96.50
Proposed		**96.80**
KTH dataset		
Xu K. et al. [30]	Two-Stream approach	95.80
Proposed		**100**
WVU dataset		
Wang et al. [31]	Statistical translation framework	96.30
Proposed		**100**
Muhavi dataset		
Murtaza et al. [32]	Motions template based approach	96.36
Proposed		**100**

4 Conclusion

In this article, we have introduced the Entropy Skewness (ES) based feature reduction and classification approach for the segmentation of regions of interest. The reduced features are optimized by a custom made genetic algorithm and the prominent features are selected, which are then provided to the multi-class classification algorithm (MC-SVM) for the classification of multiple action classes. The ES based features reduction technique performs far better as compared to PCA. The proposed system is evaluated on four publically available datasets including Weizmann, KTH, Muhavi, and WVU. Excellent results have been obtained with the recognition accuracy of 96.80%, 100%, 100%, and 100% respectively. We noticed that the proposed algorithm performs significantly better for a limited number of testing samples demonstrating scalability and efficiency of the proposed approach. The main limitation of this work is the limited number of training and testing samples. In future, we will focus on more complex action recognition challenges such as detecting suspicious behavior and forensic analysis of moving objects. To achieve this, we will investigate deep learning features to accurately and efficiently recognize complex movements.

Funding Statement: This research was supported by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist) and the Soonchunhyang University Research Fund.

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

References

[1] A. R. Zamani, M. Zou, J. Diaz-Montes, I. Petri, O. F. Rana *et al.,* "Deadline constrained video analysis via in-transit computational environments," *IEEE Transactions on Services Computing*, vol. 13, no. 1, pp. 59–72, 2017.

[2] M. Ali, A. Anjum, O. Rana, A. R. Zamani, D. Balouek-Thomert *et al.,* "RES: Real-time video stream analytics using edge enhanced clouds," *IEEE Transactions on Cloud Computing*, vol. 1, no. 4, pp. 1–8, 2020.

[3] A. Ullah, K. Muhammad, I. U. Haq and S. W. Baik, "Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments," *Future Generation Computer Systems*, vol. 96, no. 2, pp. 386–397, 2019.

[4] T. Hussain, K. Muhammad, J. Del Ser, S. W. Baik and V. H. C. de Albuquerque, "Intelligent embedded vision for summarization of multiview videos in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2592–2602, 2019.

[5] M. Ali, A. Anjum, M. U. Yaseen, A. R. Zamani, D. Balouek-Thomert *et al.,* "Edge enhanced deep learning system for large-scale video stream analytics," in *2018 IEEE 2nd Int. Conf. on Fog and Edge Computing*, Washington, DC, USA, pp. 1–10, 2018.

[6] M. U. Yaseen, A. Anjum, O. Rana and N. Antonopoulos, "Deep learning hyper-parameter optimization for video analytics in clouds," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 253–264, 2018.

[7] A. Anjum, T. Abdullah, M. Tariq, Y. Baltaci and N. Antonopoulos, "Video stream analysis in clouds: An object detection and classification framework for high performance video analytics," *IEEE Transactions on Cloud Computing*, vol. 7, no. 4, pp. 1152–1167, 2016.

[8] L. Liu, L. Shao and P. Rockett, "Human action recognition based on boosted feature selection and naive Bayes nearest-neighbor classification," *Signal Processing*, vol. 93, no. 6, pp. 1521–1530, 2013.

[9] F. Afza, M. A. Khan, M. Sharif, S. Kadry, G. Manogaran *et al.,* "A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection," *Image and Vision Computing*, vol. 4, pp. 104090, 2020.

[10] M. A. Khan, I. Haider, M. Nazir, A. Armghan, H. M. J. Lodhi *et al.,* "Traditional features based automated system for human activities recognition," in *2020 2nd Int. Conf. on Computer and Information Sciences*, Sakaka, Saudi Arabia, pp. 1–6, 2020.

[11] Q. Wu, Z. Wang, F. Deng, Y. Xia, W. Kang *et al.,* "Discriminative two-level feature selection for realistic human action recognition," *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 1064–1074, 2013.

[12] Y. Gao, M. Wang, D. Tao, R. Ji and Q. Dai, "3-D object retrieval and recognition with hypergraph analysis," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 4290–4303, 2012.

[13] Y. Yi and H. Wang, "Motion keypoint trajectory and covariance descriptor for human action recognition," *Visual Computer*, vol. 34, no. 3, pp. 391–403, 2018.

[14] S. Althloothi, M. H. Mahoor, X. Zhang and R. M. Voyles, "Human activity recognition using multi-features and multiple kernel learning," *Pattern Recognition*, vol. 47, no. 5, pp. 1800–1812, 2014.

[15] E. Zhang, W. Chen, Z. Zhang and Y. Zhang, "Local surface geometric feature for 3D human action recognition," *Neurocomputing*, vol. 208, no. 6, pp. 281–289, 2016.

[16] C. Chen, K. Liu and N. Kehtarnavaz, “Real-time human action recognition based on depth motion maps,” *Journal of Real-Time Image Processing*, vol. 12, no. 1, pp. 155–163, 2016.

[17] M. A. Khan, Y.-D. Zhang, S. A. Khan, M. Attique, A. Rehman *et al.,* “A resource conscious human action recognition framework using 26-layered deep convolutional neural network,” *Multimedia Tools and Applications*, vol. 7, no. 3, pp. 1–23, 2020.

[18] M. A. Khan, K. Javed, S. A. Khan, T. Saba, U. Habib *et al.,* “Human action recognition using fusion of multiview and deep features: An application to video surveillance,” *Multimedia Tools and Applications*, vol. 2, no. 4, pp. 1–27, 2020.

[19] M. Sharif, M. A. Khan, F. Zahid, J. H. Shah and T. Akram, “Human action recognition: A framework of statistical weighted segmentation and rank correlation-based selection,” *Pattern Analysis and Applications*, vol. 23, no. 1, pp. 281–294, 2020.

[20] M. A. Khan, T. Akram, M. Sharif, N. Muhammad, M. Y. Javed *et al.,* “Improved strategy for human action recognition; Experiencing a cascaded design,” *IET Image Processing*, vol. 14, no. 5, pp. 818–829, 2019.

[21] Y. Zhao, H. Di, J. Zhang, Y. Lu, F. Lv *et al.,* “Region-based mixture models for human action recognition in low-resolution videos,” *Neurocomputing*, vol. 247, no. 1, pp. 1–15, 2017.

[22] M. Liu, H. Liu and C. Chen, “Enhanced skeleton visualization for view invariant human action recognition,” *Pattern Recognition*, vol. 68, no. 3, pp. 346–362, 2017.

[23] I. Habib, A. Anjum, R. Mcclatchey and O. Rana, “Adapting scientific workflow structures using multi-objective optimization strategies,” *ACM Transactions on Autonomous and Adaptive Systems*, vol. 8, no. 1, pp. 1–21, 2013.

[24] H. Arshad, M. A. Khan, M. I. Sharif, M. Yasmin, J. M. R. Tavares *et al.,* “A multilevel paradigm for deep convolutional neural network features selection with an application to human gait recognition,” *Expert Systems*, vol. 2, no. 7, pp. e12541, 2020.

[25] M. Rashid, M. A. Khan, M. Alhaisoni, S.-H. Wang, S. R. Naqvi *et al.,* “A sustainable deep learning framework for object recognition using multi-layers deep features fusion and selection,” *Sustainability*, vol. 12, pp. 5037, 2020.

[26] N. Hussain, M. A. Khan, M. Sharif, S. A. Khan, A. A. Albesher *et al.,* “A deep neural network and classical features based scheme for objects recognition: An application for machine inspection,” *Multimedia Tools and Applications*, vol. 7, no. 3, pp. 1–22, 2020.

[27] A. Mehmood, M. A. Khan, M. Sharif, S. A. Khan, M. Shaheen *et al.,* “Prosperous human gait recognition: An end-to-end system based on pre-trained CNN features selection,” *Multimedia Tools and Applications*, vol. 3, no. 2, pp. 1–22, 2020.

[28] J. M. Chaquet, E. J. Carmona and A. Fernández-Caballero, “A survey of video datasets for human action and activity recognition,” *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 633–659, 2013.

[29] Q. Xiao and R. Song, “Action recognition based on hierarchical dynamic Bayesian network,” *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 6955–6968, 2018.

[30] K. Xu, X. Jiang and T. Sun, “Two-stream dictionary learning architecture for action recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 2, pp. 567–576, 2017.

[31] J. Wang, H. Zheng, J. Gao and J. Cen, “Cross-view action recognition based on a statistical translation framework,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 1461–1475, 2014.

[32] F. Murtaza, M. H. Yousaf and S. A. Velastin, “Multi-view human action recognition using 2D motion templates based on MHIs and their HOG description,” *IET Computer Vision*, vol. 10, no. 7, pp. 758–767, 2016.