Video Representation via Fusion of Static and Motion Features Applied to Human Activity Recognition

Sheeraz Arif[1], Jing Wang[1*], Zesong Fei[1], and Fida Hussain[2]
[1]School of Information and Electronics, Beijing Institute of Technology
Beijing 100081, P.R.China
[e-mail: sheeraz.arif@bit.edu.cn, wangjing@bit.edu.cn, feizesong@bit.edu.cn]
[2]School of Electrical and Information Engineering, Jiangsu University
Nanjing, P.R. China
[e-mail: fidahussain@ujs.edu.cn]
*Corresponding author: Jing Wang

Received June 20, 2018; revised November 15, 2018; accepted January 6, 2019; published July 31, 2019

Abstract

In human activity recognition system both static and motion information play crucial role for efficient and competitive results. Most of the existing methods are insufficient to extract video features and unable to investigate the level of contribution of both (Static and Motion) components. Our work highlights this problem and proposes Static-Motion fused features descriptor (SMFD), which intelligently leverages both static and motion features in the form of descriptor. First, static features are learned by two-stream 3D convolutional neural network. Second, trajectories are extracted by tracking key points and only those trajectories have been selected which are located in central region of the original video frame in order to to reduce irrelevant background trajectories as well computational complexity. Then, shape and motion descriptors are obtained along with key points by using SIFT flow. Next, cholesky transformation is introduced to fuse static and motion feature vectors to guarantee the equal contribution of all descriptors. Finally, Long Short-Term Memory (LSTM) network is utilized to discover long-term temporal dependencies and final prediction. To confirm the effectiveness of the proposed approach, extensive experiments have been conducted on three well-known datasets i.e. UCF101, HMDB51 and YouTube. Findings shows that the resulting recognition system is on par with state-of-the-art methods.

Keywords: Activity recognition; static features; motion features; trajectories; CNN; LSTM

http://doi.org/10.3837/tiis.2019.07.015 **ISSN : 1976-7277**

1. Introduction

Recently, automatic human activity recognition has become a great concern topic in the field of computer vision due to its potential and practical applications in different field such as human computer interaction, sports, healthcare, surveillance and robotics. Video captured from different devices show lots of variations such as variations in environment and variations in recording setting. Variations in environment are due to the occlusion, background cluttering, camera motion, noise and view point. Variations in video recording also cause different kinds of noise in different lighting conditions. To address these challenges, there is immense need of effective and robust activity recognition system to achieve best performance.

Information in videos are in two-dimensional domain such as static information and motion pattern. Static information is related to the background and still objects, which are very important for recognizing an activity in video. Motion information also plays an important role to capture activities related to each actor in the video independently. Therefore, to achieve the ideal recognition system, it is necessary to leverage powerful features from both static and motion components. A number of reference papers have suggested that the combination of different features extracted from different channels can make the framework more specific and robust.

In the past decade, considerable pioneer research efforts have been carried out for action recognition and changed rapidly. Early attempts are an extension of static image based representations and pattern recognition. Many researchers designed descriptors and their extensions on Hand-Crafted features to characterize visual appearance and motion dynamics. HOG (Histogram of oriented gradient) [1] extended into HOG-3D descriptor [2] by kleser et al. Scovanner et al. expanded SIFT (Scale-invariant Feature Transform) into SIFT-3D [3]. Spatial interest point method has been extended into STIP (Spatio-temporal Interest Points) [4]. These aforementioned techniques are easy to implement but very labor intensive. Many trajectory-based approaches [5-8] have been proposed to explore the underlying motion. In these approaches, trajectories are formed in groups to extract features and then local descriptors such as HOG, HOF (Histogram of Optical Flow) [9] and MBH (Motion Boundary Histogram) [7] are computed to represent shape, appearance and motion. These trajectory-based approaches achieved remarkable results and able to present complicated motion effectively. However, they suffer from number of weaknesses such as presence of redundant trajectories caused by camera motion and background. In addition, the extraction of trajectories and computation of descriptors is very complicated and may lack discriminative power for action recognition.

More recently, Convolutional Neural Network (CNN) [10 - 12] has become the research hot spot due to its pre-trained ability and automatic learning of deep representations from raw action videos. One of the prominent method is 3D CNN [13 -15], which is 3D extension of the standard 2D CNN by considering time-domain as third dimension to simultaneously encode both spatial and temporal cues. These deep learning methods provide high discriminative capacity and obtained promising results for action recognition. However, CNN based method only capture the temporal motions in short scale and lack the ability to capture long-range temporal dynamics. Moreover, CNN based method ignore the intrinsic difference between spatial and temporal domain. This problem is addressed by recurrent neural network [16 - 18], which shown remarkable performance on feature representation and temporal dynamic modeling. Especially, LSTM [19] is very popular for its effectiveness for modeling video

frames. Most of the research works [16, 18, 20] presented combination of deep learning network and LSTM, in which input to the LSTM are the high-level features extracted from the top fully-connected layer of CNNs.

In the light of the above discussion, this work is focused on fusion of both static and motion components for robust and efficient video activity recognition. We argue that merging of features extracted from different domains can really boost up the generalization ability of action recognition. Handcrafted methods are more capable of capturing motion patterns in longer temporal duration while high-level features can complement the low-level features. This work intelligently leverage the CNN generated static features and manually generated motion features. First, we propose the relevant trajectory method to reduce the computation and irrelevant background trajectories then we introduce the cholesky decomposition method to make sure the equal contribution of each (static and motion) features. Finally, the fused vectors are submitted to LSTM network to discover dynamic temporal patterns to get the high level of classification. Several extensive experiments have been carried out on different publically available datasets and achieved better results, which make our system on par with existing state-of-the-art approaches. The main contributions of this research work are summarized as follows:

1- Our propose end-to-end system, successfully leverage multiple modalities (hand crafted, deep learning and recurrent neural network) and beneficial for better recognition accuracy.
2- To reduce the background irrelevant trajectories, we apply central region process to get the most relevant trajectories, since object of interest mostly occupies the central region.
3- We introduce an effective fusion model for static and motion based on cholesky transformation, the combined fused descriptor contain the essence of both domain and vital for activity recognition
4- Different experiments have been conducted by varying static-motion contribution ratio, we achieve the optimum contribution value, which is beneficial for a better recognition rate.
5- LSTM network is used to capture underlying temporal dynamics and experimentally demonstrate the super performance of our method when evaluated on publically benchmark datasets.

The rest of article is organized as follows: Section 2 provides an overview of the related works. In section 3, we explain our approach in detail. Fusion and classification methods are addressed in section 4 and section 5 respectively. In Section 6, we demonstrate the experimental evaluation. Finally, conclusion is drawn in section 7.

2. Related Work

Our proposed approach is based on multi-model system, so we distribute our related work into following different recognition models.

2.1 Hand-crafted based representations

Early research efforts mainly rely on hand-crafted local features and have become effective representations. Most of theese approaches used detectors to define informative regions,

which are robust to video noise and background clutter. In [4], Harris3D detector has been proposed to effectively extract the salient regions. Hessian detector [20] is used for blob detection in images. 3-D SIFT [3] and cuboid [21] have shown effectiveness and robustness against noise and partial occlusion. These aforementioned approaches commonly focus on extracting texture and edge characteristics defined by interest points. However, these approaches blend together different types of motion related to human action thus resulting in a loss of discriminative power. Meanwhile many trajectory-based feature extraction methods have been introduced to facilitate motion information in effective way. Dense trajectory features (DTF) [5], make it possible to separate different types of motion information from background information but these methods do not effectively blend the different types of motion related to a human action. Many hand-crafted local descriptors such as HOG-3D [2], histogram of oriented gradient (HOG) [1], histogram of optical flow (HOF) [9] and motion boundary histogram (MBH) [7] have shown remarkable performance. These approaches extract the 3D volume around the interest points. However, unable to capture the local contents and classify the complex actions. Improved DTF (iDTF) [8], which is considered as state-of-the-art method makes use of sample interest points and optical flow to extract dense trajectories and represents each trajectory using different descriptors such as (HOG), (HOF), (MBH). IDTF uses a human detector to suppress camera motion by estimating homography and able to effectively represent the complex motion of human action. However, various issues such as presence of irrelevant and redundant trajectories and computational complexity still need to be addressed in satisfactory way.

2.2 Deep learning-based representations

Due to the remarkable success of deep CNNs in several domains such as speech recognition, object recognition and image classification, recent research is directed to deep learning-based models for action recognition. Many early methods are based on convolution neural networks (CNNs) to learn deep video representations. Ji et al. [13] and Tran et al. [14] extended 2D ConvNet to a video domain and tested ConvNet with deep architecture on short datasets and large datasets respectively. Two-stream ConNet designed by simonyan et al. [11] containing spatial and temporal dependecies has achieved remarkable performance. Deep ConNets are automatic end-to-end trainable networks, and there engineering process for feature representation is not labour-intensive and complicated. We can characterized the deep features by their highly sparsity and discriminative capacity. Despite these strengths, deep learning based approaches still suffer from a number of limitations. CNN-based networks only capture temporal dynamics and ignore the intrinsic difference between spatial and temporal domain. Another problem associated with these methods is that they highly rely on large training datasets while most of the available datasets are very small.

2.3 RNN based representations

Many researchers learn video representation by using Recurrent Neural Network (RNN) due to its ability for modelling video sequences and its multi-disciplinary applications. Especially, LSTM [19] overcome the weaknesses of short-snippet learning approaches by capturing the long range temporal dynamics and has proven very successful in actions prediction and sequence generation tasks. In most of the research works [16, 18, 22, 23], high-level features abstracted from the fully-connected layer of CNN are the input of LSTM, which lack the fine action details in video sequences. Multiple layers recurrent networks and various feature fusion techniques have been introduced in [24 - 25], which are indeed very effective in action recognition tasks. Recently, attention based methods [26 - 27] have been combined into LSTM

to emphasizing the key spatial-temporal segments. However, in [26], attention mechanism largely ignore the spatial cues while method presented in [27] ignores the motion cues of actions. Appearance and motion cues are integrated in [28], which lacks rich spatio-temporal components among video sequences.

2.4 Fusion based representations

Information extracted from different channels/domain can be very effective in various recognition tasks. 3D convolution is applied on stacke of images in [13 - 14] to perform feature level fusion. Auto-encoder and correlation analysis are proposed in [29] to fuse the features obtained from RGB and depth images. Direct concatenation of Flow and RGB features is introduced in [30 - 31]. Many authors proposed late fusion techniques at the output level in their research work [10, 30] to train two-stream network. Different from previous proposed method, our work intelligently incorporates the static and motion cues obtained from different domain. Our propose fusion method provides the power to control the equal contribution of each domain in exact numbers. The resultant fused descriptor is then input to the LSTM network to identify the spatio-temporal regions and certainly beneficial for action recognition enhancement.

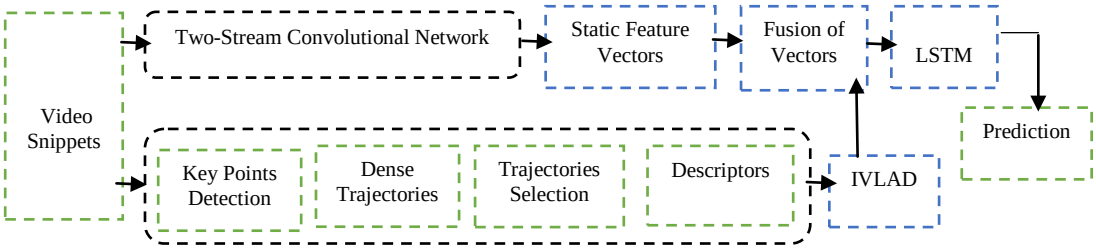

Fig. 1. Overall framework of our propose approach

3. Methodology

This section illustrates our framework, which input an untrimmed snippets video and classify the human activity accordingly. The overall flowchart of our method is demonstrated in **Fig. 1**. We explain the detail description of each component i.e. extraction of static and motion features, fusion method and activity classification by LSTM in subsequent sections.

3.1 Extraction of Static Features

First, we capture static features of video by using widely used two-stream 3D ConNets [13]. Static features include still objects and background information which is very important for determining an action in such scenario, where body movements of group of actors are similar such as group of people fighting is nearly related to body movement of sports event e.g., wrestling. We can decompose a video into two stream i.e. RGB stream and flow stream. RGB frame represented by high dimensional features such as background, objects and actors. As, our frame work consist of different modalities to learn static and motion features. This section is focus on abstracting static features. RGB frames are fed into classic deep convolutional network similar to the two-stream 3D ConNets [13]. Our framework utilizes only spatial stream and accept RGB clips to extract static features, as RGB single frame usually encode static information.

CNNs is a depth model consists of trainable filters and pool operations for abstracting spatiotemporal features hierarchy with increasing degree of complexity. Increasing number of layers enhance the degree of extraction of features learned by filters. Usually, bottom layers are used to learn underlying features such as edge and color and final layers captures the complete key features. There are different variants of CNNs are available and two-stream 3D ConNets [13] is very popular among them, which is ideal network to learn spatio-temporal information simultaneously in end-to-end fashion. The architecture of this network consists of 8 convolutional layers, 5 pooling layers, 2 fully connected layers and then final softmax output layer. All 3D convolutional kernels are 3 x 3 x 3 with stride 1 and all pooling layers are 2 x 2 x 2 with stride 2 except pooling layer1. Pooling layer 1 has the stride of 1 x 2 x 2 with intention to retain the spatial information in early phases. We utilize one-stream 3D ConNets, which takes only RGB clips as input. It adopts spatial 3D ConNet to extract static features for each video sequence clip. Clip level static features are captured from the first fully connected layer, which has 4096 output units.

3.2 Extraction of Motion Features

This section highlights the extraction of motion information in the form of motion descriptors. We adopt hand crafted feature technique to capture motion patterns by arguing that traditional techniques can be extended to longer motion duration. So, it is possible to obtain and discriminate motion classes. Both CNN and traditional descriptors techniques process the input information in a region just like sliding window. Most of the traditional hand-crafted techniques follow the three basic steps to capture the feature vectors. 1- Detection of key/ interest points. 2- Extraction of trajectories. 3- Computation of descriptors to align the trajectories to obtain relationship among the trajectories.

3.2.1 Key point detection

A video frame can be represented by evaluating the feature such as HOG, HOF and MBH on the group of key points. Instead of determining interest points, key points can be captured to compute local motion descriptor. Scale invariant feature transform (SIFT) [32 - 33] is suitable candidate to detect the key points. In our method, we utilize SIFT detector to map the spatial contents of frame such as location, scale and invariant features. A special kind of Gaussian function, which is also known as scale kernel function is used to extract features at different scale. At any point (x, y) of the frame with scale ϕ, the Gaussian function can be given as $G(x, y, \phi)$. If scale space function is defined as $F(x, y, \phi)$ of an image $I(x, y, \phi)$, the difference of Gaussian function L can be computed by following given equation (1).

$$L(x, y, \phi) = F(x, y, C\phi) - F(x, y, \phi) \quad (1)$$

Where C is a constant. This difference of Gaussian function provides the scale-invariant points, which are also known as key points. So, it is likely to be expected, these multi-scale space based key points can characterize something change occur in better way than those points using single scale only.

3.2.2 Extraction of Selected Trajectories

Trajectories are very essential to capture the local information of video, which guarantees the good estimation of foreground motion. The corresponding trajectories can be extracted by tracking the key points frame by frame. The method of tracking the key points is essentially

similar as tracking the interest points [34 - 35]. In our method, we utilize these trajectories for computation of shape descriptor, which is very important component for activity recognition. We adopt the method of [34] for tracking spatial scale points by using median filter with one modification i.e., replacement the optical flow with SIFT flow [36 - 37]. The position of any point $P_t = (x_t, y_t)$ in frame I_t can be tracked in frame I_{t+1} by applying median filter on SIFT flow within 3 x 3 patch:

$$P_t = (x_t, y_t) + (K_{3 \times 3} * \omega_t)|_{(xt, yt)} \tag{2}$$

Where, K is the medial filter kernel of size 3 x 3 pixels. Once the SIFT flow field is computed, key points can be tracked without additional cost. The points (P_t, P_{t+1}, P_{t+2}...) of subsequent frames can be concatenated to form trajectory.

In order to achieve the computation efficiency, we argue that there are many redundant irrelevant background trajectories, which can cause to increase the computation complexity. To tackle this problem, we introduce very simple method to reduce the background trajectories by assuming that object of interest is often occupies the middle portion of the frame and so the relevant trajectories are. We choose the middle region of frame by selecting width and height as two-third of the original frame. If T is the original extracted trajectories, so T_r i.e. relevant trajectory can be selected or rejected by following criteria:

$$T_r = \begin{Bmatrix} \text{select; if}(\overline{x_r}, \overline{y_r}) \subset \text{CR} \\ \text{reject; if}(\overline{x_r}, \overline{y_r}) \not\subset \text{CR} \end{Bmatrix} \tag{3}$$

Where, $(\overline{x_r}, \overline{y_r})$ is the mean value of the coordinates (x, y) of selected trajectory T_r. In this way, we can ignore those trajectories which are related to background and only relevant trajectories can be selected.

3.2.3 Computation of Descriptors

Optical flow is considered as the most popular method for capturing motion information in video frames. It can be define as the displacement of pixel intensity $X(x, y)$ in two consecutive frames i.e. t and $t+1$ and can be defined as $X(x-\alpha, y-\beta)$, where vector (α, β) is called optical flow. However, optical flow may not capture the change of semantics and motion accurately, in addition optical flow process is very complex and gets overwhelmed by the camera motion with respect to background. Recently, SIFT flow [36 - 37] is introduced as an alternative way and provides an effective way to demonstrate the displacement between key points in two consecutive frames.

SIFT flow furnishes a compact way for obtaining the local descriptor and it is invariant to scale changes. We can utilize the key points to compute the gradient at point X and its surrounding grid points. A histogram can be constructed commonly known as histogram of oriented optical flow (HOF). The HOF can be defined as the probability density function of the optical flow at any particular point in the frame. It has been observed that it is very beneficial to decompose the gradient of optical flow at point X and surrounding points into its x and y directions. So, histogram known as motion boundary histogram (MBH) and can be constructed in both directions x and y respectively so we have MBH_x and MBH_y. The motion boundary histogram (MBH) is well known to cancel out the camera motion and distortions caused by camera motion. Thus, Sift flow can be utilized to represent a video by describing the key points using histogram of oriented gradients at the multiple scales instead of evaluating single scale at the group of interest points.

The Sift flow w for the key point p can be obtained by solving the discrete optimization problem as in [38] for the frame i. All the different terms associated with sift flow can be represented by energy function $E(w)$, and can be given as in equation (4):

$$E_i(w) = s_i(p) + w_i(p) + [u_i(p) + v_i(p)] \quad (4)$$

Where, $w(p)$ and $s(p)$ are the flow vector and SIFT descriptor respectively at pixel point p of the i^{th} frame. The last term, allow us to differentiate the vertical $v(p)$ and horizontal $u(p)$ flow. In this way, we can capture the motion feature vectors, which are useful to maintain the robustness against the rotation scale and viewpoints.

In our method, we treat the Sift flow in the similar fashion as optical flow to compute the histogram of oriented Sift flow HOF-S and also motion boundary histogram in both direction x and y known as MBH-S_x and MBH-S_y. We split the key points and its neighbour points in 16 x 16 grid of four cells of dimension 4 x 4 in the frame. At each of the key point the gradient of pixel intensity variation can be computed. The space coordinate in every cell is divided in to 8 different bins to quantize the gradient orientation. The histogram of orientations for each cell can be constructed by counting the number of orientation in each of the 8 bins. So there are 16 cells in a frame and histograms of each individual cell can be combined to obtain 128 feature vectors. If there are N key points then each frame of video can be represented as N x 128 dimensional feature vector. In this way a set of feature vectors can be evaluated by using key points in the form of HOF-S, MBH-Sx and MBH-Sy. Moreover, we can compute the trajectory shape descriptor (TSD) by utilizing selected trajectories extracted in our previous subsequent section 3.2.2. These trajectories are traced by the key points and helpful in determining the movement of key points across the video. We set the maximum length of trajectory as 17 to avoid the drifting problem.

3.3 Feature Coding Scheme (iVLAD)

We adopt improved vector of locally aggregated descriptor (iVLAD) [39] as feature representation method to aggregate all residual vectors i.e. motion feature descriptors and shape feature descriptors. In many recent research work, VLAD achieved the better results than other mid-level video representation schemes such as BoW and iFV. First, difference between each feature descriptor and its closest center is computed as residual vector and then *L2* normalization is applied to the residual vectors to yield the final VLAD vector. This process ensures the equal contribution of all descriptors.

4. Fusion Method

Fusion of information extracted from several domains and channels can make the method more specific and enhance the robustness in activity recognition tasks. Our extracted static and motion features define the internal relationship but the final accuracy depends on the ratio of contribution of each domain. Our fusion method is based on Cholesky decomposition, which has the tendency to precisely control the contribution of both static and motion domains empirically for the final fused descriptor. An abstract version of cholesky transformation is described below.

According to cholesky transformation, two random variables with unknown correlation can be transformed into new variables with known correlation. If R and S are two random variables without any correlation and these two random variables can be transformed into two new

random variables (Y and Z) with known correlation ρ. This transformation can be represented by equation (5).

$$\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{bmatrix} \times \begin{bmatrix} R \\ S \end{bmatrix} \tag{5}$$

So that, $Y = R$

and

$$Z = \rho R + \sqrt{1-\rho^2} S \tag{6}$$

The above representation ensures the correlation between two transformed random variables Y and Z is ρ. By following the above property of the cholesky transformation, we can fuse our static and motion vectors with the known correlation. Let suppose A and B be static and motion vectors respectively. Cholesky transformation can be applied to these two static and motion vectors with the correlation ρ_1 in the following manner.

$$\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho_1 & \sqrt{1-\rho_1^2} \end{bmatrix} \times \begin{bmatrix} A \\ B \end{bmatrix} \tag{7}$$

Therefore, $Y = A$

$$Z = \rho_1 B + \sqrt{1-\rho_1^2} B \tag{8}$$

Similarly, to control and guarantee the contribution of both vectors, this transformation can be applied to motion vector B and static vector A with correlation ρ_2.

$$\begin{bmatrix} S \\ M \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho_2 & \sqrt{1-\rho_2^2} \end{bmatrix} \times \begin{bmatrix} B \\ A \end{bmatrix} \tag{9}$$

$$S = B$$

$$M = \rho_2 A + \sqrt{1-\rho_2^2} A \tag{10}$$

Again, cholesky transformation guarantees the following two properties.

1) ρ_1 is the correlation between vectors A and B.
2) ρ_2 is the correlation between vectors B and A.

Therefore, if the values for ρ_1 and ρ_2 are selected in such a way that they satisfied the following rule,

$$\rho_2 = \sqrt{1-\rho_1^2} \tag{11}$$

In this way, it can guaranteed that $Z = M, \ \forall A, B, \rho_1, \rho_2$. Hence, the resultant vector E can be obtained by following relation,

$$E = Z = M \tag{12}$$

Where the correlation between E and A is ρ_1 and the correlation between E and B is ρ_2. Here A and B represent our extracted static and motion vectors whereas E represents the resultant vector. This representation lead us to an important intuition: by choosing the value of ρ_1, we can choose the degree to which the static and the motion features contribute for our resultant vector. In our upcoming section 6.3.2, it is also shown, how this property is used to explore, the optimal contribution of both static and motion domain information for recognising actions.

5. Action Prediction by LSTM Network

5.1 Long Short-Term Memory (LSTM)

For analysing the hidden sequential patterns, it is natural choice to use RNN to encode the temporal structure of extracted sequential features. In video visual information is represented in many frames which help in understanding the context of an action. RNN can interpret such sequences but in case of long term sequences, it usually forget the earlier input sequence. LSTM has been designed to mitigate the vanishing problem and to learn the long-term contextual information of temporal sequence. LSTM [19] is one kind of recurrent networks, which can capture the long-term dynamics and preserves sequence information overtime. In addition, in LSTM gradient does not tend to vanish when trained with back propagation through time and it can keep the certain state in memory for longer period of time. Its special structure with input, output and, and control gates control the long term sequence pattern identification. The gates are adjusted by sigmoid unit that learns during training when to open and close.

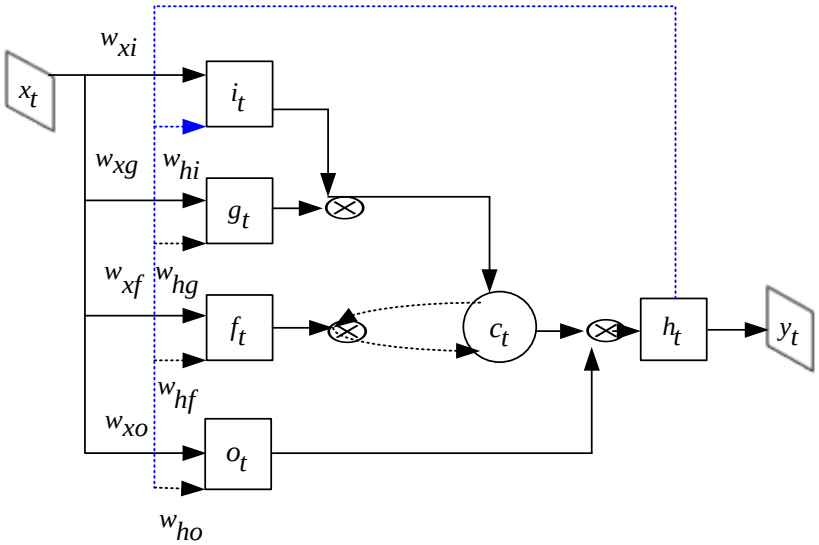

Fig. 2. The architecture of LSTM Unit.

Fig. 2 demonstrates the architecture of a LSTM cell with the working of three existing gates. x_t, c_t, h_t and y_t stand for input vector, cell state, hidden state and output at the t^{th} state, respectively. The output y_t depends on hidden h_t state, while h_t depends on not only the cell state c_t but also its previous state. Intuitively, the LSTM has the capacity to read and write to its internal memory, and hence maintain and process information over time. LSTM neuron contains an input gate i_t, a memory cell c_t, a forget gate f_t, and an output gate o_t. At each time

step *t*, it can choose to write, read or reset the memory cell through these three gates. This strategy helps LSTM to access and memorize information in many steps.

In data flow process, for input x_t at time step *t*, a LSTM cell preserves the last cell state c_{t-1} and last cell output h_{t-1}. The current cell state can be represented by c_t, which is the summation of the previous memory cell state c_{t-1} and a function of the current input and previous hidden state. The output value and the value of the three gates (i_t, f_t and o_t) can be computed by Eq. (13) to Eq. (18) which demonstrates the operation of temporal modelling performed in LSTM unit.

$$i_t = S(w_{xi}\, x_t + W_{hi}\, h_{t-1} + b_i) \quad (13)$$
$$f_t = S(w_{xf}\, x_t + W_{hf}\, h_{t-1} + b_f) \quad (14)$$
$$o_t = S(w_{xo}\, x_t + W_{ho}\, h_{t-1} + b_o) \quad (15)$$
$$g_t = tanh(w_{xq}\, x_t + W_{hq}\, h_{t-1} + b_g) \quad (16)$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes g_t \quad (17)$$
$$h_t = o_t \otimes \tanh(c_t) \quad (18)$$

where *S* is denoted as sigmoid non-linearity function, *tanh* is the hyperbolic tangent of non-linearity function and $\otimes$ indicates the product between elements and the gate value. The cell state and output are computed step by step to extract long-term dependencies. Based on the LSTM unit, for an input x_t at time step *t*, the LSTM computes a hidden/control state h_t and a memory cell state c_t, which is an encoding of everything the cell has observed until time *t*.

5.2 Prediction process

We can represent the generated fused feature vectors as a set of *t* D - dimensional features as $E = \psi f_v(\{X_i\}t=1)$. If e_i is the vector of the *i*-th frame (time step). Let *E* be an input sequence(e_1, ..., e_T) and *y* be an output sequence(y_1, ..., y_T). An LSTM then maps *E* to *y* by using series of intermediate operation as mentioned in Eq. (13) to Eq. (18):

$$y_t = W_{hy}\, h_t + b_y \quad (19)$$

where, *W* and *b* are the trained parameters of LSTM, which denotes the weights and the biases of input layer and the hidden layer respectively. The final single label prediction for a video can be produced by using softmax classifier. Softmax layer can be utilized to achieve the M-way class scores for a given video sequence. This single prediction can be achieved by averaging the label probabilities by the Eq. (20).

$$P(y_t^q = 1) = \text{softmax}(y_t) = \text{softmax}(W_{hy} h_t + b_y) \quad (20)$$

Where *t* is the the current time step, and $q \in Q$ is a prediction. But it is expected that the prediction cannot be very accurate and may be there is large number of uncertainty at the beginning. To take this fact into account, we modify the loss function, Eq. (20), by adding an exponential term:

$$Loss = \sum_{n=1}^{N} \sum_{t=1}^{T} -e^{-(T-t)/2} \log(y_t^q) \quad (21)$$

where y_t^q is the output of the softmax layer, i.e., the estimated probability of the prediction being of category *q* at time step *t*. *N* is the number of the training sequences during the training.

The implication of this modified loss function is that it discounts those predictions at the beginning, or in other words, the importance of the prediction grows with time.

6. Experiments

In this section, the proposed approach is experimentally evaluated on three well-known benchmark human action datasets: UCF101 [40], HDMB51 [41] and YouTube [42]. The description of these datasets, experimental setup and comparative analysis are presented in following subsequent sections.

6.1 Description of Datasets

The **UCF101** dataset [40] is widely adopted benchmark for human action recognition and also the extension of UCF50. It comprises of 101 action classes and around 100 video clips are associated with each action class. There are 13,320 video clips in total. Most of the video clips are realistic, clean and user-uploaded videos with cluttered background and camera motion. We adopt validation scheme of the THUMOS13 challenge [43] and follow the three testing/training split for performance evaluation by reporting average recognition accuracy.
The **HDMB51** dataset [41] is the large collection of variety of realistic videos ranging from YouTube and Google videos to digitized videos collecting from various sources. In total, there are 6,766 manually annotated video sequences of 51 different action categories and each category containing at least 100 video clips. This dataset is very challenging and complex as it contains videos with more interclass difference and complicated background. For experimental setting, we follow the original evaluation guidelines [45] using three different testing/training splits. Each split with each action class has 30 clips for testing and 70 clips for training. We report the average recognition accuracy over these three splits.
The **YouTube** dataset [42] comprises of total 1168 videos with 11 different action classes. Sequences in each class are grouped into 25 categories and in each category, there are at least 4 action clips. We adopt the validation protocol as given in [42] by using leave-one-out strategy. This strategy involves one fold as testing videos and rest of the folds for the training videos.

6.2 Experimental Setup and Parameter Tuning

This section explains the implementation details for the validation scheme of benchmark datasets and training of 3D convolutional network. As UCF101 is larger than HMDB51 dataset so we use it to train 3D convolutional network initially, and transfer the learned model on HMDB51 dataset for extraction of deep features. We use split1 of UCF101 to extract the deep features. Caffe toolbox is used for ConvNet implementation while OpenCV implementation is used for extraction of trajectories and for implementation of LSTM network, we utilize the code provided by [16]. All results are obtained on a single Geforce GTX Titan 3.6GHZ with 6 GB RAM, not using any parallel processing. 3D convolutional net is trained on I380K and fine-tune the model parameter of UCF101 at initial learning rate of 0.003 and is divided by 2 after every 150K iterations. The optimization is stopped at 1.9M iterations.

6.3 Experiments and Comparative Analysis

To analyse the richness and effectiveness of our proposed method in the context of action recognition, we describe series of experiments. The experimental results and comparative analysis are presented in subsequent sections.

6.3.1 Number of Generated Trajectories

In this section, we evaluatedour approach in terms of number of trajectories generated. As we mentioned earlier, most of traditional methods are based on trajectories formation method. However, there may be the presence of many irrelevant and invalid trajectories caused by background and camera movement and processing of these extra trajectories is very complicated process. To address this problem, we propose central region process to capture irredundant and most relevant trajectories. We use the 320 different sample videos with average frame size of 335 x 240 pixels from YouTube dataset and also101 videos from the action class pushing of HMDB51 dataset with average frame size of 335 x 240 pixels. **Fig. 3** illustrates the number of trajectories generated by different trajectory based methods. We compare our central region (CR) approach method with dense trajectories (DT) [5], ordered trajectories (OD) [6] and trajectories rejection (TR) [45] method. According to the results, our approach generates minimum number of trajectories which certainly very effective to minimize the computation complexity of different subsequent operations without any significant loss of accuracy. The possible reason is that, we do not follow any frame skipping scheme as in [45] to prevent the chance of losing some valid information from skipping frames.

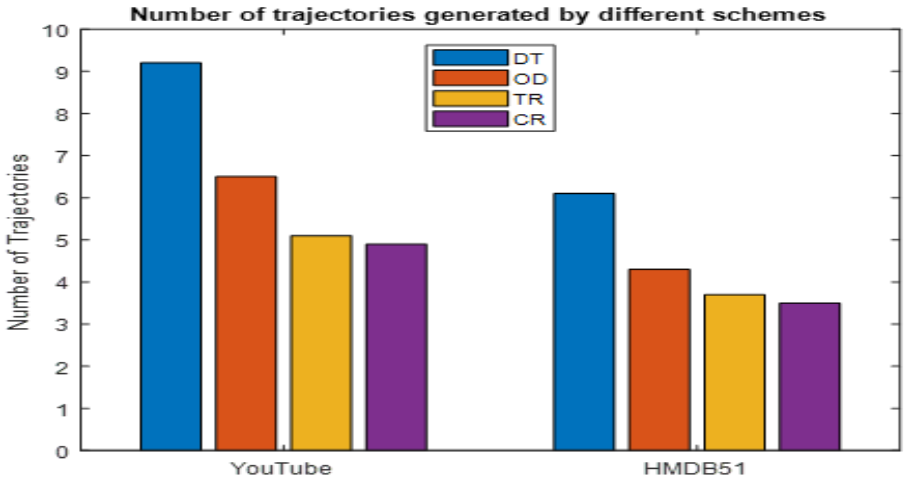

Fig. 3. Number of trajectories generated by different methods on YouTube and HMDB51 datasets

6.3.2 Contribution Level of Static and Motion Components by Varying Ratios

As our fusion method is based on cholesky transformation and according to the formulation derived in section 4, we can control the contribution level of both static and motion components by adjusting the value of ρ. We obtain the results for HMDB51, UCF101 and YouTube datasets by using for the different mathematical values of ρ. As evident from the evaluation from **Fig. 4**, there are some optimum values for the contribution ratio for which we obtain the highest recognition accuracy. Such as 60:40 (motion: static) for HMDB51, 80:20 (motion: static) for UCF101 and 80:20 (motion: static) for YouTube respectively.

Contribution ratio for HMDB51 is different than other datasets used, the possible reason is that most of the videos in HMBD51 dataset are with more interclass differences and complicated background and required more contribution of static features to classify the activities in videos. In our next experimental sections, we use the same contribution ratio, since we obtain the best

results on these contribution ratios. The obtained results proved that the cholesky transformation is very effective and provides the controlling contribution power of each domain in exact value and variance ratio.

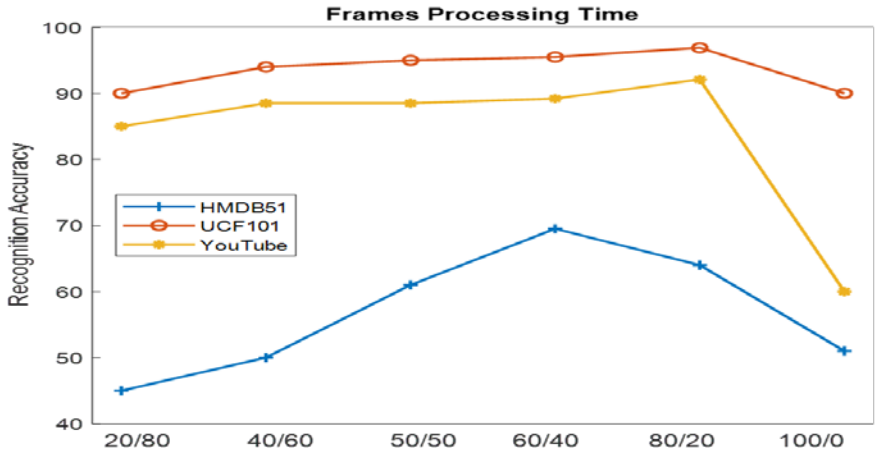

Fig. 4. Overall accuracy for varying contribution ratio between motion and static components on HMDB51, UCF101 and YouTube datasets.

Table 1. Comparison of different fusion method on UCF101 dataset

Fusion Method	PCA	Concatenation	LWF	EWS	Cholesky
T-Jumping	91.2%	90.5%	90.9%	90.0%	93.3%
S-Juggling	92.6%	90.1%	91.4%	90.2%	94.9%
H-Riding	88.7%	91.2%	92.1%	91.1%	93.1%
G-Swinging	92.8%	90.9%	91.1%	89.6%	94.3%
Diving	90.2%	89.9%	91.0%	90.9%	95.1%
Biking	91.2%	90.3%	92.1%	91.1%	94.1%
B-Spiking	90.0%	89.1%	91.1%	90.1%	91.8%
Avg. Accuracy	90.9%	90.2%	91.2%	90.5%	94.3%

6.3.3 Comparison of Different Features Fusion Methods

In this section, we analyse the effect of different early fusion methods. We report the per-class recognition rates obtained for each fusion model in **Table 1**. We compare our fusion approach (Cholesky) with different existing fusion model such as Principle of Component Analysis (PCA), concatenation, linear weighted fusion (LWF) and element-wise sum (EWS). We utilize different classes from three splits of UCF101 dataset such as T-Jumping, S-Juggling, H-Riding, G-swinging, Diving, Biking and B-spiking. In section 6.3.2, we achieved the optimum contribution ratio for UCF-101 dataset as 80:20 (motion: static), so we keep the same contribution ratio for this section for further verification. We observe that Cholesky transformation based fusion method enhances the recognition accuracy of our approach by fair margin as compared to other fusion methods. We obtained overall accuracy of 94.3%. As ratio of contribution of each static and motion information is vital for activity recognition and optimum contribution depends on the richness of motion information in video. Our introduced fusion method addresses these all issues and effectively incorporate the spatial correspondence between static and motion features.

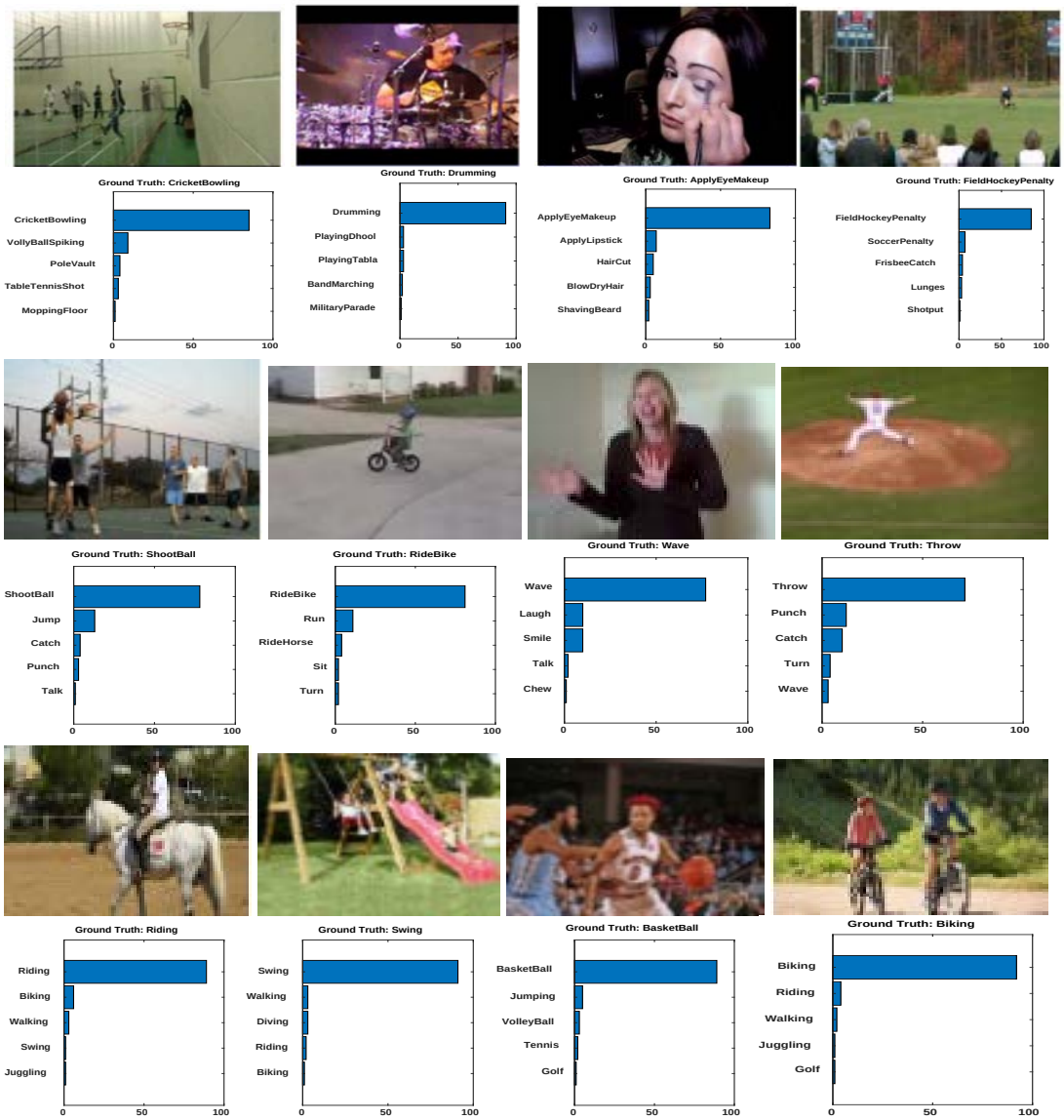

Fig. 5. Examples of correct predictions on UCF-101(First Row), HMDB51 (Second Row) and YouTube (Third Row) datasets

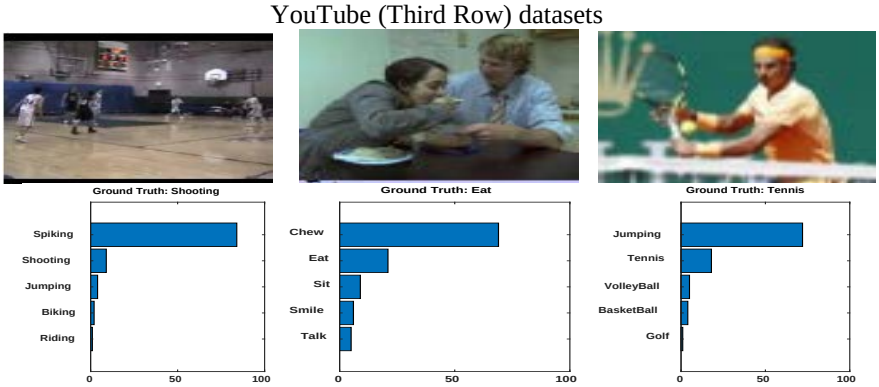

Fig. 6. Examples of incorrect (Failed) predictions on UCF-101, HMDB51 and YouTube datasets

6.3.4 Recognition Visualization

Furthermore, for a better understanding of our proposed method, we provide additional recognition visualizations on video examples from three standard datasets. The proposed approach is tested on 40% videos of UCF-101, HMDB51 and YouTube datasets. Some of the intermediate frames of an action along with correct/successful visual recognition results are shown in **Fig. 5**. In this figure, first, middle and last row show the successful recognition examples from UCF-101, HMDB51 and YouTube dataset respectively. Recognition accuracy of each action frame is shown by bar graph which indicates the ground truth and bars below show model prediction sorted in decreasing confidence. In our model, LSTM returns output for each chunk and finally the video is classified for the highest frequency class in outputs. In **Fig. 6**, we also list some mis-classified/fail predictions from each of the dataset, where "Shooting" is classified as "Spiking" (First Column, UCF-101), "Eat" is predicted as "Chew" (Second Column, HMDB51) and "Tennis" is classified as "Jumping" (Last Column, YouTube). These incorrect predictions are due to the similarity in the scenarios, background, camera motion and motion of body parts of an actor performing actions, so there is possibility of generation of similar appearance and motion based features. It can be also observed from both figures that recognition scores of UCF-101 dataset are more than 90% it is because of UCF-101 dataset is relatively large dataset for training which is able to recognize fine-grained examples. Overall, our model obtained good results from all three datasets. Thus, from the qualitative examples, we conclude that our approach can achieve promising performance in practice.

6.3.5 Comparison to the State-of-the-art Methods

In this section, we further verify the effectiveness and feasibility of our model, we compare our proposed approach to different existing state-of-the-art human action recognition approaches on both UCF101 and HDMB51 datasets over three splits. The comparison results are reported in **Table 2**. We organize these baseline methods into different categories with respect to the type of features and network being used, including traditional, deep-learned features, very deep-learned features and fusion based methods (hybrid features). Compared to the traditional methods our model performs better by 4.9% on both datasets. Compare with RNN based methods such as (LSTM) [24] and (LRCN) [16], our model outperforms these two methods by 4.3% and 10% on both datasets respectively. Different experiments indicating that our approach possess higher discriminative power and our system to be on par with the state-of-the-art. It can be also seen that some methods with both features such as TSN [55] lead to a performance gain by minimal margin on the UCF101 dataset. However, our introduced method outperforms the 3D conv – iDT [13] by 1.7% and TSN [55] method by 0.9% on the HDMB51 dataset and shows higher recognition rate. We can conclude that, combination of LSTM with 3D convolutional network achieve better results and obtains the recognition rate of 94.0% and 70.7% on UCF101 and HDMB51 datasets respectively and show that there is a degree of complimentary among traditional, convolutional neural network and LSTM network.

Table 2. Comparison our method with state-of-the-art existing method on UCF101 and HDMB51 dataset

Modality	Method	Year	UCF101 (%)	HDMB51 (%)
Traditional	iDTF+fisher vector [8]	2013	84.7	57.2
	Ordered Trajectories [6]	2015	72.8	47.3
	MPR [46]	2015	-	65.5
	MoFAP [47]	2016	88.3	61.7
	Trajectory Rejection [45]	2016	85.7	58.9
Deep	Two-Stream [11]	2013	88.9	59.4
	TDD [48]	2015	90.3	63.2
	FSTCN [15]	2015	88.1	59.1
	DANN [49]	2016	89.2	63.3
	Dynamic Images [50]	2016	89.1	65.2
Very deep	C3D [14]	2015	85.2	-
	LSTM [24]	2015	88.6	-
	LRCN [16]	2016	82.9	-
	3D Convolution [13]	2016	91.8	64.6
	STPP-LSTM [52]	2017	91.6	69.0
	FCNs-16 [53]	2017	90.5	63.4
	Hidden-Two-stream [54]	2017	90.3	58.9
	VideoLSTM [51]	2018	89.2	56.4
(Fusion Method) Hybrid features	TDD-iDT [48]	2015	91.5	65.9
	C3D-iDT [14]	2015	90.4	-
	LTC-iDT [17]	2016	92.7	67.2
	TSN [55]	2016	**94.2**	69.4
	3D conv + iDT [13]	2016	93.5	69.2
	FCN-16+iDT [53]	2017	93.0	70.2
Ours	SMFD + iFV	-	92.9	68.1
	SMFD + iVLAD	-	94.0	**70.7**

7. Conclusion

In this paper, we propose an end-to-end approach for human activity classification which is based on static and motion features. Static features are learned by RGB stream of two-stream 3D convolutional neural network and motion features are extracted by adopting traditional trajectory based approach. We utilize SIFT property for the detection of key points and computation of shape and motion descriptors. We introduce central region process to get the most relevant and valid trajectories to reduce the background irrelevant trajectories and speedup the recognition process. Moreover, we design cholesky transformation based fusion method to effectively fuse static and motion information. Cholosky transformation method provides the powerful way to control the contribution of each domain's information in exact numbers, which is certainly very important for action recognition tasks. Finally, LSTM network is used to model the temporal progression and classification of human activity. Experimental results conducted on different public benchmark datasets prove the superiority of our model compared with other state-of-the-art methods.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 886–893, June 20-25, 2005. Article (CrossRef Link).

[2] A. Klaser, M. Marszalek, and C. Schmid, "A Spatio-Temporal Descriptor Based on 3D-Gradients," in *Proc. of 19th British Machine Vision Conference, British Machine Vision Association: Leeds, United Kingdom*, pp.1–10, September, 2008. Article (CrossRef Link).

[3] P. Scovanner, S. Ali and M. Shah, "A 3-Dimensional SIFT Descriptor and its Application to Action recognition," in *Proc. of the 15th International Conference on Multimedia*, pp. 357–360,September 25 – 29, 2007. Article (CrossRef Link).

[4] I. Laptev, "On Space-Time Interest Points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp.107–123, September, 2005.

[5] H. Wang, A. Klaser, and C. Schimid, "Action recognition by dense trajectories," in *Proc. of IEEE conference on computer vision and pattern recognition*, pp.3169-3176, June 20-25, 2011. Article (CrossRef Link).

[6] O.V.R. Murthy and R.Goecke, "Ordered trajectories for large scale human action recognition," in *Proc. of IEEE conference on computer vision and pattern recognition*, pp. 412-419, December 2-8, 2013. Article (CrossRef Link).

[7] H. Wang, A. Klaser A and C. Schimid, "Dense trajectories and motion boundary descriptor for action recognition," in *Proc. of international journal of computer vision*, vol. 103, pp. 60-79, March, 2013. Article (CrossRef Link)

[8] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. of IEEE International conference on computer vision*, pp. 3551-3558, December 1-8, 2013. Article (CrossRef Link).

[9] N. Dalal, B. Triggs and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. of European Conference on Computer Vision* , pp 428-441, May 7-13, 2006. Article (CrossRef Link).

[10] A. Karpathy, G. Toderici, S. Shetty and T. Leung, "Large-scale video classification with convolutional neural networks," in *Proc. of IEEE conference on computer vision and pattern recognition*, pp. 1725 – 1732, June 23-28, 2014. Article (CrossRef Link).

[11] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, vol. 1, pp. 568-576, June, 2014. Article (CrossRef Link).

[12] G.W Taylor, R. Fergus and Y. LeCun, "Convolutional learning of spatio-temporal features," in *Proc. of 11th European conference on Computer vision*, pp. 140-153, September 5-11, 2010. Article (CrossRef Link).

[13] Ji Si, Xu W, Yang M, et al., "3d convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell*., vol.35, no.1, pp.221–231, January, 2013. Article (CrossRef Link).

[14] D. Tran, L. Bourdev and Fergus, "Learning spatiotemporal features with 3d convolutional networks," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, pp. 4489–4497, December 7-13, 2015. Article (CrossRef Link).

[15] L. Sun, K. Jia, and D. Yeung, "Human action recognition using factorized spatio-temporal convolutional networks," in *Proc. of IEEE International Conference on computer vision (ICCV*), pp. 4597 – 4605, December 7-13, 2015. Article (CrossRef Link).

[16] J. Donahue, L.A. Hendricks and S. Guadarrama, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Transactions on Pattern Analysis and Machine* Intelligence, vol, 39, no. 4, pp. 677 – 691, September, 2016. Article (CrossRef Link).

[17] G. Varol, I. Laptev, and C. Schmid, "Long-term Temporal Convolutions for Action Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1510 – 1517, June ,2017. Article (CrossRef Link).

[18] Z. Wu, X. Wang, and Y.G. Jiang, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," in *Proc. of the ACM international conference on Multimedia*, pp. 461-470, October 27-30, 2015. Article (CrossRef Link).
[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *neural computation*, vol.9, no.8, pp. 1735-1780, November, 1997. Article (CrossRef Link).
[20] G. Willems, T. Tuytelaars and L.J.V. Gool, "An efficient dense and scale – variant spatio-temporal interest point detector," in *Proc. of European Conference on Computer Vision (ECCV)*, pp. 650-663, October 12-18, 2008. Article (CrossRef Link).
[21] P. Dollar, V. Rabaud and G. Cottrell, "Behavior recognition via sparse spatio-temporal features," *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 65-72, October 15-16, 2005. Article (CrossRef Link).
[22] N. Srivastava, E. Mansimov and R. Salakhutdinov, "Unsupervised Learning of Video Representations using LSTMs," *in Proc. of the International Conference on Machine Learning*, pp. 843-852, July 6-11, 2015. Article (CrossRef Link).
[23] N. Ballas, L. Yao and C. Pal C, "Delving deeper into convolutional networks for learning video representations," in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, March, 2016. Article (CrossRef Link).
[24] J.Y. Ng, M. Hausknecht and S. Vijayanarasimhan , "Beyond short snippets: Deep networks for video classification," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), pp. 4694–4702, June 7-12, 2015.Article (CrossRef Link).
[25] H. Gammulle, S. Denman and S. Sridharan, "Two stream lstm: A deep fusion framework for human action recognition," in *Proc. of IEEE Winter Conference on Applications of Computer Vision* (WACV), Santa Rosa, USA, pp. 177 – 186, March 24-31, 2017. Article (CrossRef Link).
[26] S. Yeung, O. Russakovsky and N. Jin, "Every moment counts: Dense detailed labeling of actions in complex videos," *International Journal of Computer Vision*, vol.126, no.2-4, pp. 375–389, April, 2018. Article (CrossRef Link).
[27] S. Sharma, R. Kiros and R. Salakhutdinov, "Action recognition using visual attention," in *Proc. of Neural Information Processing Systems (NIPS) Time Series Workshop*, December, 2015. Article (CrossRef Link).
[28] Y. Wang, S. Wang and J. Tang, "Hierarchical attention network for action recognition in videos," *ArXiv*, July, 2016. Article (CrossRef Link).
[29] H. Zhu, J. Weibel and S.Lu, "Discriminative multi-modal feature fusion for RGBD indoor scene recognition," in *Proc. of IEEE Conference on Computer Vision and Pattern* Recognition (CVPR), pp. 2969 – 2976, June 27-30, 2016. Article (CrossRef Link).
[30] Z. Wu, X. Wang and Y. Jiang, "Modeling spatial temporal clues in a hybrid deep learning framework for video classification," in *Proc. of ACM international conference on Multimedia*, pp. 461-470, October 26 – 30, 2015. Article (CrossRef Link).
[31] C. Feichtenhofer, A. Pinz and R.P. Wildes, "Spatiotemporal residual networks for video action recognition," in *Proc. of Conference on Neural Information Processing Systems*, pp. 1-9, December, 2016. Article (CrossRef Link).
[32] D.G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proc. of international Conference on Computer Vision*, pp. 1150 – 1157, September 20-27, 1999. Article (CrossRef Link).
[33] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Key points," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November, 2004. Article (CrossRef Link).
[34] Farneb¨ack, "Two-frame motion estimation based on polynomial expansion," in *Proc. of the Scandinavian Conference on Image Analysis (SCIA)*, pp 363-370, June 29 -July 2, 2003. Article (CrossRef Link).
[35] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," in *Proc. of IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no.3, pp.500–513, August, 2011. Article (CrossRef Link).

[36] C. Liu, J. Yuen and A. Torralba, "SIFT Flow: Dense Correspondence across Different Scenes," in *Proc. of European Conference on Computer Vision* (ECCV), pp. 28-42, October 12-18, 2008. Article (CrossRef Link).
[37] C. Liu, J. Yuen and A. Torralba, "SIFT Flow: Dense Correspondence across Scenes and its Applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 978 – 994, May, 2011. Article (CrossRef Link).
[38] Y. Boykov, O. Veksler and R. Zabih, "Fast approximate energy minimization via graph cut," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no.11, pp. 1222 – 1239, November, 2001. Article (CrossRef Link).
[39] J. Delhumeau, P.H. Gosselin and H. Jegou, "Revisiting the VLAD image representation," in *Proc. of the 21st ACM international conference on Multimedia*, Barcelona, Spain, pp. 653-656, October 21-25, 2013. Article (CrossRef Link).
[40] K. Soomro, A.R. Zamir and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *Published in OALib journal*, 2012. Article (CrossRef Link).
[41] H. Kuehne, H. Jhuang and H. Garrote, "HMDB: a large video database for human motion recognition," in *Proc. of IEEE International Conference on Computer Vision*, pp. 2556-2563, November 6-13, 2011. Article (CrossRef Link).
[42] J. Liu, J. Luo and Shah, "Recognizing realistic actions from videos in the wild," in *Proc. of IEEE conference on computer vision and pattern recognition*, pp. 1996 – 2003, June 20-25, 2009. Article (CrossRef Link).
[43] Y.G. Jiang, J. Liu and A.R. Zamir, "THUMOS challenge: Action recognition with a large number of classes," 2013. Article (CrossRef Link).
[44] P. Wang, Y. Cao and C. Shen, "Temporal pyramid pooling based convolutional neural networks for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, pp. 2613 – 2622, June, 2017. Article (CrossRef Link).
[45] J.J Seo, H.I. Kim and DE. Neve, "Effective and efficient human action recognition using dynamic frame skipping and trajectory rejection," *Journal of Image and Vision Computing*, vol.58, pp. 76-85, February, 2017. Article (CrossRef Link).
[46] B. Ni, P. Moulin, X. Yang and S. Yan, "Motion part regularization: Improving action recognition via trajectory selection," in *Proc. of IEEE conference on* (CVPR), Boston, MA, USA, pp. 3698 – 3706, June 7-12, 2015. Article (CrossRef Link).
[47] L. Wang, Y. Qiao and X. Tang, "Mofap: a multi-level representation for action recognition," *International Journal of Computer Vision*, vol.119, no.3, pp. 254-271, 2016. Article (CrossRef Link).
[48] L. Wang, Y. Qiao and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Proc. of IEEE Conference on Computer Vision and Pattern* Recognition (CVPR), pp. 4305–4314, June 7-12, 2015. Article (CrossRef Link).
[49] J. Wang, W. Wang and R. Wang, "Deep alternative neural network: exploring contexts as early as possible for action recognition," *Advances in Neural Information Processing Systems* (NIPS), pp.811–819, December, 2016. Article (CrossRef Link).
[50] H. Bilen, B. Fernando and E. Gavves, "Dynamic image networks for action recognition," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3034-3042, June 27-30, 2016. Article (CrossRef Link).
[51] Z. Li, E. Gavves, M. Jain and C.G.M. Snoek, "VideoLSTM convolves, attends and flows for action recognition," *Computer Vision and Image Understanding*, vol. 166, pp. 41-50, January 2018. Article (CrossRef Link).
[52] X. Wang, L. Gao, and P. Wang, "Two-stream 3D convNet Fusion for Action Recognition in Videos with Arbitrary Size and Length," *IEEE transaction on multimedia*, vol.20, no. 3, pp. 634-644, March 2018. Article (CrossRef Link).
[53] S. Yu, Y. Cheng and L. Xie, "Fully convolutional networks for action recognition," *Institution of Engineering and Technology (IET) Computer vision*, vol.11, no.8, pp. 744 -749, December, 2017. Article (CrossRef Link).

[54] Y. Zhu, Z. Lan and S. Newsam, "Hidden two-stream convolutional networks for action recognition," *ArXiv*, April, 2017. Article (CrossRef Link).
[55] L. Wang and Z. Wang, "Temporal segment networks: towards good practices for deep action recognition," in *Proc. of Euro. Conf. on Computer Vision*, pp. 20–36, October 11-14, 2016. Article (CrossRef Link).

Sheeraz Arif received the M.S degree in Telecommunications Engineering and Computer Networks from London South Bank University, London, UK, in 2006. He is currently pursuing the PhD. Degree from the School of Information and Electronics Engineering, Beijing Institute of Technology China. His research interests include machine learning, human action recognition, computer vision and video analysis.

Jing Wang received the Ph.D. degree in Electronic Engineering in 2007 from Beijing Institute of Technology (BIT), China. She is now an Associate Professor in School of Information and Electronics Engineering, Beijing Institute of Technology China and currently associated with the Research Institute of Communication Technology (RICT) in Beijing Institute of Technology. Her research interests include speech and audio signal processing, multimedia quality assessment and mobile communication

Zesong Fei received the Ph.D. degree in Electronic Engineering in 2004 from Beijing Institute of Technology (BIT), China. He is now a Professor in School of Information and Electronics Engineering, Beijing Institute of Technology China. He is currently associated with the Research Institute of Communication Technology (RICT) in Beijing Institute of Technology. Where, he is involved in the research of designing of the next generation high-speed wireless communication. His research interest include wireless communication and multimedia signal processing. He is also chief investigator of National Natural Science Foundation of China and senior member of Chinese Institute of Electronics and China Institute of Communication.

Fida Hussain received the B.E degree from D.U.E.T. Pakistan, in 2009 and the M.E. degree from Hamdard University Pakistan, in 2011. The Ph.D. degree from School Electrical and Information Engineering, Jiangsu University, China in 2018. His research interests include smart grids, power system automation machine learning and hydropower automation.
E-mail: fida.hussain07@yahoo.com
fidahussain@ujs.edu.cn