Neural Computing and Applications (2021) 33:469–485
https://doi.org/10.1007/s00521-020-05018-y

ORIGINAL ARTICLE

A deeply coupled ConvNet for human activity recognition using dynamic and RGB images

Tej Singh[1] · Dinesh Kumar Vishwakarma[2]

Received: 6 May 2019 / Accepted: 7 May 2020 / Published online: 21 May 2020

Abstract
This work is motivated by the tremendous achievement of deep learning models for computer vision tasks, particularly for human activity recognition. It is gaining more attention due to the numerous applications in real life, for example smart surveillance system, human–computer interaction, sports action analysis, elderly healthcare, etc. Recent days, the acquisition and interface of multimodal data are straightforward due to the invention of low-cost depth devices. Several approaches have been developed based on RGB-D (depth) evidence at the cost of additional equipment's setup and high complexity. Contrarily, the methods that utilize RGB frames provide inferior performance due to the absence of depth evidence and these approaches require to less hardware, simple and easy to generalize using only color cameras. In this work, a deeply coupled ConvNet for human activity recognition proposed that utilizes the RGB frames at the top layer with bi-directional long short-term memory (Bi-LSTM). At the bottom layer, the CNN model is trained with a single dynamic motion image. For the RGB frames, the CNN-Bi-LSTM model is trained end-to-end learning to refine the feature of the pre-trained CNN, while dynamic images stream is fine-tuned with the top layers of the pre-trained model to extract temporal information in videos. The features obtained from both the data streams are fused at decision level after the softmax layer with different late fusion techniques and achieved high accuracy with max fusion. The performance accuracy of the model is assessed using four standard single as well as multiple person activities RGB-D (depth) datasets. The highest classification accuracies achieved on human action datasets are compared with similar state of the art and found significantly higher margin such as 2% on SBU Interaction, 4% on MIVIA Action, 1% on MSR Action Pair, and 4% on MSR Daily Activity.

Keywords Bi-LSTM · Deep learning · Dynamic motion image · Human activity recognition

1 Introduction

Recently, the vision-based understanding of event entices numerous real-life applications such as gaming, robotics, patients monitoring, content-based retrieval, video surveillance, and security [1–4]. Over the decade, many efforts made to recognize the human activity in videos, but still, it is a challenging task due to intra-class action similarity, occlusions, view variations, and environmental conditions [5, 6]. Since the introduction of inexpensive depth Kinect sensor, various solutions proposed based on multiple modalities using RGB, depth, and 3D coordinate data and their multiple combinations. However, such approaches face limitations due to positions of joints and coordinates of upper or lower body parts used to represent activity. An analysis of existing challenges in RGB and RGB-D datasets of human activity is discussed in [7, 8]. It is observed that most of the real-life applications for activity recognition use RGB only cameras such as CCTVs installed on public places for monitoring purposes. However, the solutions based on features extraction from RGB only frames are not sufficient enough to identify suspicious human activity due to the lack of temporal information and

✉ Dinesh Kumar Vishwakarma
dinesh@dtu.ac.in

Tej Singh
tejsingh_2k16phdec05@dtu.ac.in

[1] Department of Electronics and Communication Engineering, Delhi Technological University, New Delhi 110042, India

[2] Department of Information Technology, Delhi Technological University, New Delhi 110042, India

existing environmental conditions. Formerly, the optical flow estimation is most popular motion representation method in video sequences. However, optical flow-based solutions are less effective in challenging conditions such as view variations and occlusions present in videos. Therefore, we utilized the concept of a compact single dynamic motion image (DMI) obtained from the whole action video using an approximate rank pooling mechanism [9]. The DMIs captured the long-term dynamics and motion cues to understand temporal patterns in videos automatically. In this work, we utilized RGB only frames to learn the features that make our model less complex, robust, and efficiently apply for daily life activities.

In the proposed hybrid model, we utilized the latest pre-trained Inception-v3 deep architecture [10] that outperforms and shows excellent accuracy results in ILSVR 2012 image classification competition as compared to other pre-trained deep models such as AlexNet [11] and VGGNet [12]. Further, the Bi-LSTM model is used to deal with sequential data that increases the overall efficiency of the proposed architecture. Our deep model outperforms the state-of-the-art methods on standard human activity RGB-D datasets by utilizing RGB only frames of video.

The remaining sections of the proposed work are as follows: Sect. 2 consists of related work of various handcrafted and deeply learned approaches for human activity analysis. Section 3 consists of the underlying architecture of proposed deeply coupled ConvNet. The experimental setup, training details, results, and a discussion on standard video benchmarks are explained in Sect. 4. Finally, the work is concluded in Sect. 5.

2 Related work

In the last decade, many handcrafted and automatically learned feature-based approaches developed for human action recognition in the videos. Earlier human activity recognition approaches are based on handcrafted features mainly focused on simple atomic actions that seem to be somewhat less useful to practical applications [13–18]. The main drawback of these approaches is data pre-processing and difficult to generalize in real life despite gaining a high accuracy model. Later on, after the success of convolutional neural networks (CNNs) on text and image classification, various spatiotemporal approaches for video activity analysis were proposed that can automatically learn the features and classify from raw RGB video only [19–21]. However, such approaches could not achieve higher accuracy due to data dependency for training the CNN models and the lack of hardware resources [22–26].

Initially, a global representation-based features approaches are proposed for human activity analysis. Bobick and Davis [27] introduced the average energy-based approach motion history image (MHI) and motion energy image (MEI) to recognize human activities in controlled environmental conditions. Blank et al. [28] addressed the activity classification problem using space–time 3D template MEI based on extracted human silhouettes. However, such holistic solutions are less sensitive to measure the possible variation in the videos, e.g., view variation and occlusions [29]. To overcome the limitations of global approaches, Laptev [30] proposed an efficient local feature extractor space–time interest points (STIPs) for action recognition by utilizing the extension of Harris detector. Matikainen et al. [31] introduced a trajectory-based local feature extractor to recognize human action in the video sequence. In order to recognize the actions of different length and timescales in real time, a method based on the string kernel is proposed by Brun et al. [32]. They represented an action with the help of a string called "aclet," and similarity between these "aclets" is described based on Gaussian kernel. Charletti et al. [33] proposed a hybrid feature extractors approach to recognize human activity using depth videos. The dimensionality of obtained feature vectors is reduced with the help of LDA and PCA techniques. Finally, the activities are classified using GMM classier. However, they claimed superior accuracy, but their modal is sensitive to view variations and noises. Ji et al. [23] introduced a soft regression-based transition maps approach for early detection of human activities using depth frames only. They divided human action into different patterns and evaluated the temporal coherence between action sequences. Seggesi et al. [34] proposed an automatic configurable trained feature extractor for the representation of skeleton pose from training samples data. However, their modal showed promising results on static poses, but the response is found slow for real-time action recognition. To recognize the online action in a complex background using depth cameras, Ji et al. [22] proposed a hybrid approach by embedding skeleton coordinates into depth frames and extracted features using a spatiotemporal pyramid on a partitioned set of action sequences. However, these methods are showing satisfactory performance but less competent to tackle the environmental changes such as camera jitters, occlusion, and illumination variations [35–38].

The first step toward the automatic features learning from raw RGB frames was introduced by Ji et al. [20]. This approach is based on 3D CNN for human action recognition in videos. However, due to rigid architecture for accepting fixed input frames, such a model is not so efficient for video of different lengths. Simoyan and Zisserman [21] proposed a two-stream deep model that accepts raw RGB frames and optical flow as an input to the pre-trained deep learned model. Baccouche et al. [19] introduced a

hybrid model for action recognition that extracts spatial features using CNN and LSTM for motion cues. Currently, a combination of both in-depth and local features are popular for activity recognition [21, 39–41]. Feng et al. [26] proposed a geometrical relational features approach based on multilayer LSTM network for recognition of human activities using skeleton joints information. Keçeli et al. [42] presented an approach to recognize the dyadic activity from depth sequence that is the combination of 3D and 2D CNN architectures. They extract the temporal features through 3D CNN trained 3D depth volume, while 2D CNN is fine-tuned on weighted sum depth sequences. The obtained features are ranked using relief algorithm and classify using an SVM classifier. Ijjina and Chalavadi [43] proposed a multimodal action recognition model based on feature extraction from RGB and depth videos using CNN architecture. An ELM classifier is used to recognize the human activities from these fused features architecture. Jing et al. [44] presented a spatiotemporal-based hybrid neural network which characterizes human activity in RGB-D video based on a two-stream neural network. Further, they claimed to improvement in the classification accuracy by utilizing joint loss function to exploit the spatial and temporal features of videos. To address the human activity, Srihari et al. [45] introduced a four-stream CNN network consisted of two RGB-D video data and two temporal motion optical flow streams. Elboushaki et al. [46] proposed a multi-dimensional CNN that learned high-level features for gesture recognition in RGB-D videos in conjunction with LSTM network. They investigated various fusion schemes at different layers of a deep network for the classification task.

Based on the above literature review, it can be observed that shape and motion information is correlated with in-depth sequences. It is challenging to record both of these information simultaneously. Furthermore, the 3D skeleton joint coordinates are not capable of discriminating some activities due to noise and occlusion errors such as self-occlusion with body parts, etc. There are some action examples such as 'eating' and 'drinking' having the same motion pattern which cannot be distinguished by using 3D skeleton joints coordinates. Hence, motivated from the earlier state of the art, we have designed a tightly coupled deep ConvNet for human activity recognition using RGB frames of the videos.

3 The proposed ConvNet architecture

The improved image recognition approaches extend and motivate automatic human action recognition in video sequences. Most of the video action recognition approaches are based on shallow higher-dimensional spatiotemporal features extraction from stacked raw video frames. Human action can be disintegrated into spatial and temporal variations in a video. The spatial features contain the appearance information about an object in each sequence of a given video. On the other hand, temporal features represented in the form object moving across the video sequences. The proposed human action recognition model accordingly divided into two streams model as depicted in Fig. 1. Hence, in this work, a hybrid two-stream deep architecture based on two different data streams (spatial and temporal) is proposed, which are then fused by late fusion techniques. The spatial features are extracted from RGB frames fed at regular interval to the input of first or upper stream as shown in Fig. 1, and at the same time, the temporal or second or bottom stream has input as DMI, which captures the full temporal dynamics of human action. Both the streams are trained using pre-trained Inception-v3 deep architecture.

The first or upper stream of the network is trained end-to-end learning through a pre-trained ConvNet and followed by the Bi-LSTM network for the acquisition of additional sequential information. The fully connected layer flattened the output of Bi-LSTM, and softmax layer gives the probabilistic score, which is going to fuse with the temporal score obtained by the second or bottom stream.

The second or bottom stream of the network is fine-tuned on the pre-trained ConvNet, and in this stream, a temporal enrich images called DMIs are inputted to the network, which extracts temporal features. These obtained features are passed to a fully connected layer, which flattened the features, and further softmax layer gives the probabilistic temporal score.

These streams of networks are connected parallelly, as shown in Fig. 1. Further, the scores obtained by these two streams of the network are fused with late fusion techniques at the decision level to enhance the classification accuracy of the proposed model.

3.1 Features extraction with pre-trained Inception-v3 architecture

In this section, we briefly discussed the underlying architecture of deep Inception-v3. Figure 2 shows the deep Inception-v3 architecture consisted of one input block, three blocks of Inception Modules A, B and C, two blocks of grid size reduction, one auxiliary classifier block and one output block. Inception-v3 network enriched with some advanced features such as RMSProp optimizer, batch normalization, label smoothing to reduce overfitting and add loss function as compared with the previous version of inception architectures. This network consists of 42 deep layers that accept input data $299 \times 299 \times 3$ of spatial

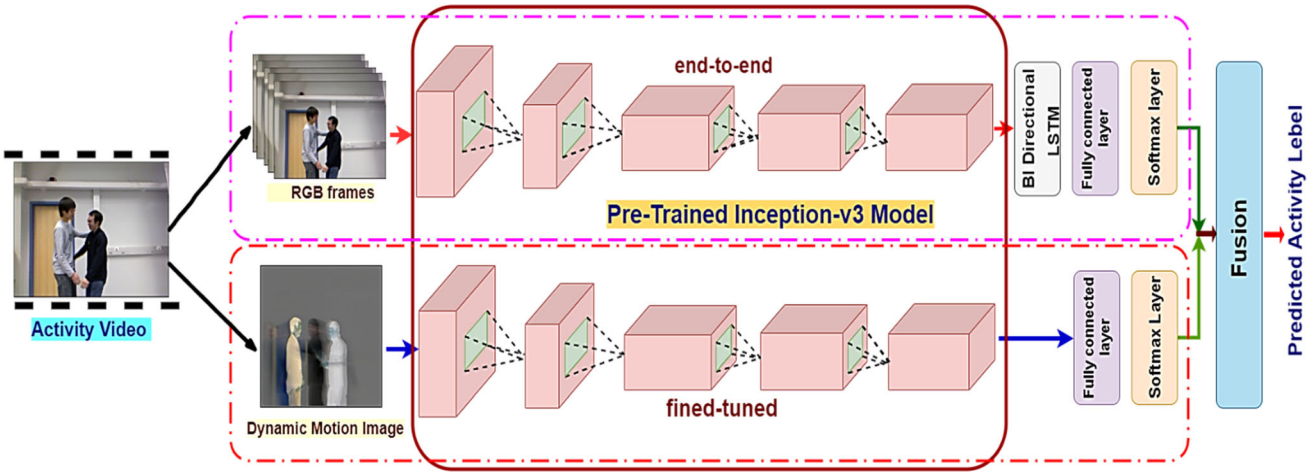

Fig. 1 The proposed ConvNet architecture for human activity recognition

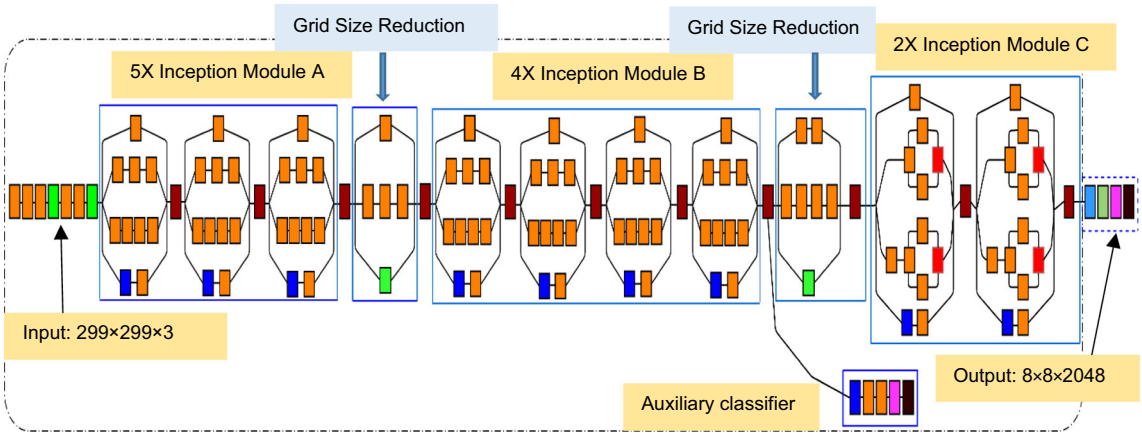

Fig. 2 Block diagram of Inception-v3 improved deep architecture

dimension at input block. The Inception Module A is used for smaller factorization convolutional (Conv), and a 5×5 (Conv) filter is converted into two 3×3 Conv filters which results in reducing parameters between the layers without decreasing the efficiency of the network. The Inception Module B utilized for asymmetric spatial factorization convolutions which converts a 3×3 Conv filter into 1×3 Conv followed by 3×1 Conv filters. It can be observed that a $n \times n$ Conv can be represented by $1 \times n$ Conv followed by a $n \times 1$ Conv saving the computational cost and reduced the overall parameters. The Inception Module C is introduced for stimulating the high-dimensional representations similarly working as Module B in this network. The grid size reduction block is used for downsizing the feature map such as in deep AlexNet or VGGNet models. The main difference between traditional models and Inception-v3 is that a $m \times m$ grid with n filters is divided into $m/2 \times m/2$ grid with $2n$ filters. Thus, the overall computational cost is decreased by using convolutional operation followed by pooling operation. This network contains one auxiliary classifier on the top of the last 17×17 layer, which is used as a regularizer for enhancing the convergence of the deep network. In the following subsection, we briefly discussed the underlying architecture of LSTM and Bi-LSTM for sequential data. In the proposed work, the deep Inception-v3 network is trained in an end-to-end manner followed by the Bi-LSTM architecture, which represents one of the action descriptors for the two-stream HAR model.

3.2 The Bi-Directional LSTM (Bi-LSTM)

It is observed that the traditional feedforward network and convolutional neural network are inefficient to deal with sequential data such as video analysis. These networks take input of fixed length. To overcome the problem of fixed size, input padding is used, but the performance of such approaches is not comparable with RNNs [47] and LSTM network [48]. The RNN deep network has compatibility with the help of the chain and loop structure to deal with

sequential data. However, RNNs are inefficient as compared to long short-term memory for longer duration sequential input because of the vanishing gradient in the backpropagation process. To overcome the problem of vanishing gradient, the LSTM architecture was proposed, which utilized two parallel RNNs for enhancing the long-term dependencies and training for bigger input networks. The gated structure of the LSTM network helped to overcome the many sequential learning problems in a very effective way. Figure 3 illustrates the underlying blocks architecture of LSTM network. It can be observed from Fig. 3 that LSTM network has a similar chain-like structure like RNN. It consisted of four neural layers, while RNN is having only one neural sigmoid layer (tan$h$). A horizontal line at the top of Fig. 3 is called the "cell state" of LSTM which works like a conveyor belt. This cell state has the capabilities to remove and add information. The LSTM consists of three gates: input (i_t), forget and output (o_t) gates to control and protect the cell states. These gates have sigmoid neural layer and capable of processing the information through pointwise multiplication units.

In the beginning, the "forget gate" (f_t) decides what information between (h_{t-1}, x_t) is passed through the cell states and defined as Eq. (1):

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

where the activation function is denoted by the $\sigma(\bullet)$ and x_t, h_{t-1}, W_f, and b_f represent the input, output at previous LSTM block, weight, and bias at the forget gate layer, respectively, at time t. On the next step, the input gate (i_t), decides what new information is to store in the cell state. The input gate equation is given as Eq. (2):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

where W_i, and b_i represent the weight and bias at the input gate layer, respectively.

For the update, a new state in the cell state, a sigmoid function creates a new vector $\hat{C}_t$ as defined by Eq. (3). Later on, Eqs. (2) and (3) are combined for updating a new state:

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

In order to update the old cell (C_{t-1}) into new (C_t), the old state is multiplied with forget gate and add other parameters as shown in Eq. (4):

$$C_t = f_t * C_{t-1} + C_{t-1} + i_t * \hat{C}_t \tag{4}$$

Finally, the output o_t is given the cell state based on the output of the sigmoid of output gates and defined by:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

where W_o, and b_o represent the weight and bias at the output layer, respectively.

It is observed that unidirectional LSTM network stores the information data from the past direction only. On the other hand, a Bi-LSTM network preserves the information data both direction from past to future and the future to past. Further, the Bi-LSTM network performed much better than LSTM for sequential data applications. The basic architecture of a Bi-LSTM network is depicted in Fig. 4. The output sequence in the forward layer $\vec{h}$ is computed for given inputs in positive sequence from $x_{t-1}\ldots\ldots x_n$. The backward output sequence $\overleftarrow{h}$ is computed for reversed inputs sequences. The final layer output is evaluated with the help of Eqs. (1)–(6) for the backward and forward layer outputs. The bi-directional LSTM layer yields an output vector as: $Y_T = y_{t-1}, y_t, \ldots y_{t+n}$ in which each $y_t = \sigma\left(\vec{h}, \overleftarrow{h}\right)$,

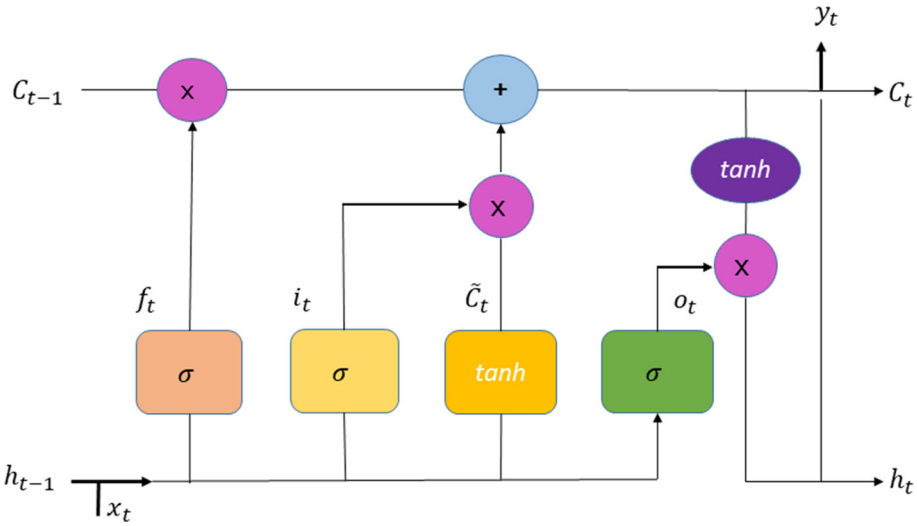

Fig. 3 Basic LSTM architecture

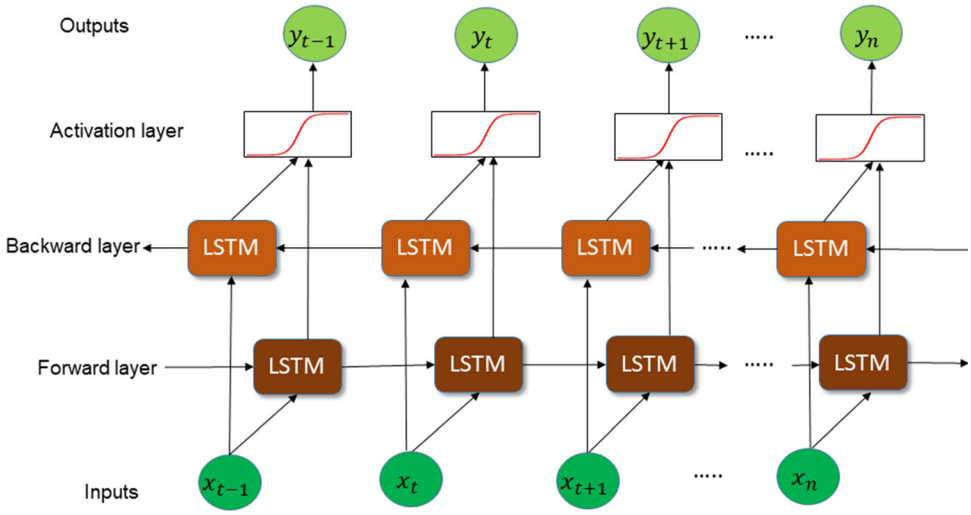

Fig. 4 The bi-directional LSTM architecture

where σ is called a summation function or concatenating function.

It is observed that fine-tuning the all sequential frames related to action labels is not a good idea for learning useful discriminative features. Therefore, in this work, RGB video is sampled out ten frames at a fixed interval of to the pre-trained CNN architecture and trained, excluding the fully connected layer. The upper half of the proposed ConvNet is end-to-end trained with only the last five layers. For each frame, a features vector of dimension $10 \times 4 \times 4 \times 2048$ is obtained from CNN architecture. Further, these extracted features are given to average pooling layer and got a vector of size $10 \times 1 \times 2048$ for each layer. Finally, the feature vectors acquired from the pre-trained model are further fed to the Bi-LSTM layer for better feature extractions. It helps for better generalization in terms of learning features from frames.

In the following subsection, the concept of dynamic motion images and fine-tuning with CNN for temporal extraction features is explained in detail.

3.3 Dynamic motion image (DMI) from video sequences

In this section, we emphasized understanding the long-term temporal dynamics in terms of a single dynamic image. Later on, these images are fed as an input for fine-tuned Inception-v3 architecture for features extraction. It is a paramount task to understand the content of videos precisely on a large scale because videos are consisted of a sequence of still images. Therefore, the summarization of the whole video sequence into a single still dynamic image using standard CNN architecture is introduced in work [9]. A rank pooling mechanism is adopted that utilized the work of Fernando et al. [24] for obtaining a dynamic image from whole video sequences. This technique encrypts the temporal variation cues of the video sequences into one single image. A rank pooling function directly applied to raw RGB frames to produces a single dynamic image for each activity video.

The idea behind the creation of dynamic images depends on the ranking function [24] that ranks its each video frames $(I_1, I_2, I_3, \ldots\ldots, I_T)$ according to the time axis. Let a feature vector represented by $(I_t) \in \mathbb{R}^{\mathbf{d}}, t \in [1, T]$, where $\varphi(\cdot)$ is showed the rank for each frame I_t at time instance t. A score function $\mathcal{S}(t|\mathbf{d}) = \langle \mathbf{d}, \mathcal{V}_t \rangle$, is associated with ranking function, where $\mathbf{d} \in \mathbb{R}^{\mathbf{d}}$ is a vector of parameters and $\mathcal{V}_t = \frac{1}{t}\sum_{\tau=1}^{t} \varphi(I_t)$ is the time average of these features at this instant. According to the RankSVM [49], the learning of vector $\mathbf{d}$ is modelled as a convex optimization problem and given by Eq. (7),

$$\mathbf{d}^* = \rho(I_1, I_2, I_3, \ldots\ldots, I_T; \varphi) = \underbrace{\operatorname{argmin}}_{d} E(\mathbf{d}) \tag{7}$$

$$E(\mathbf{d}) = \frac{\delta}{2} \parallel \mathbf{d} \parallel^2 + \frac{2}{T(T-1)} \times \sum_{q>t} \max\{0, 1 - \mathcal{S}(q|\mathbf{d}) - \mathcal{S}(t|\mathbf{d})\} \tag{8}$$

where $\rho(I_1, I_2, I_3, \ldots\ldots, I_T; \varphi)$ maps a sequence of T number of video frames into a single vector $\mathbf{d}^*$ is called a rank pooling function that aggregates information from all video frames. The first term $\frac{\delta}{2} \parallel \mathbf{d} \parallel^2$ of Eq. (8) is a quadratic regularized function used for SVM. The second term related to soft counting loss that calculates how many pairs are not correctly ranked for $q > t$ by the ranking function. The score function is calculated for video frames based on ranking function, and a pair of the frame is chosen for which score having unit margin i.e. $\mathcal{S}(q|\mathbf{d}) > S(t|\mathbf{d}) + 1$.

The dynamic motion image obtained using approximate rank pooling function [9] is fifty times faster than rank pooling function for similar performance. The approximate

rank pooling mechanism is based on a gradient optimization algorithm and derived using Eqs. (9)–(12) as follows:

For

$$\mathbf{d} = 0,\ \mathbf{d}^* = \vec{0} - \eta \nabla E(\mathbf{d})\Big|_{\mathbf{d}=\vec{0}} \propto -\nabla E(\mathbf{d})|_{\mathbf{d}=\vec{0}}$$

for all $\eta > 0$ where

$$\begin{aligned} \nabla E\left(\vec{0}\right) &\propto \sum_{q>t} \max\{0, 1 - \mathcal{S}(q|\mathbf{d}) - \mathcal{S}(t|\mathbf{d})\}\Big|_{\mathbf{d}=\vec{0}} \\ &= \sum_{q>t} \nabla \mathbf{d}, \\ \mathcal{V}_t - \mathcal{V}_q &= \sum_{q>t} \mathcal{V}_t - \mathcal{V}_q \end{aligned} \tag{9}$$

Further, the function $\mathbf{d}^*$ used as video descriptor because it aggregates information from all stacks frame and defined as:

$$d^* \propto \sum_{q>t} \mathcal{V}_t - \mathcal{V}_q = \sum_{q>t} \left[\frac{1}{q}\sum_{i=1}^{q} \varphi_i - \frac{1}{t}\sum_{j=1}^{t} \varphi_j\right] = \sum_{t=1}^{T} \Omega_t \varphi_t \tag{10}$$

where the Ω_t is denoted as:

$$\Omega_t = 2(T - t + 1) - (T + 1)(\boldsymbol{H}_T - \boldsymbol{H}_{t-1}) \tag{11}$$

where $H_t = \sum_{i=1}^{T} \frac{1}{t}$ is the ith Harmonic number and $H_0 = 0$.

Therefore, the updated approximate rank pooling function is defined as the weighted sum of adjacent video frames as:

$$\hat{\rho}(I_1, I_2, I_3, \ldots\ldots, I_T; \varphi) = \sum_{t=1}^{T} \Omega_t \varphi_t \tag{12}$$

The weights parameter $\Omega_t \in (1, T)$ is calculated for fixed video length accordingly, as shown in Fig. 5.

This demonstration showed that Ω_t depends on the consecutive video frames belongs to frames (1, T). The weight parameter Ω_T is the combination of the sum of all weight parameters obtained from each frame $\frac{2*i-T-1}{i}$, where $i \in (t, T)$. It is observed that DMIs computation is limited only to by pre-multiplying function Ω_t for all video frames. Therefore, approximate rank pooling function does not require to compute the intermediate average features vector $\mathcal{V}_t = \frac{1}{t}\sum_{\tau=1}^{t} \varphi(I_t)$.

Instead, it can be directly calculated by using individual frame feature $\varphi(I_t)$ and $\Omega_t = 2t - T - 1$, which is a linear function of time t. The example of dynamic image formation using the approximate rank pooling technique from 'approaching' activity video is illustrated in Fig. 6. It can be noted that each video frames is multiplied with the corresponding weight, i.e. frame I_1 is multiplied by Ω_1 for each channel individually. The final obtained DMI is the weighted sum of each Red, Green, and Blue channels of each frame and having the same size as the frames. It is observed that the temporal action modelling pattern can be easily seen from a dynamic image when a person is approaching another person irrespective to background and illumination conditions. The so obtained DMIs are fed to pre-trained ConvNet to extracts the temporal features to recognize the action in a video.

3.4 Late fusion

The late fusion techniques for two-stream networks are categorized as Sum Fusion, Maximum fusion and Concatenation Fusion [50]. The main objective of fusing the two-stream network is that features extracted from the same pixel's location through the different channel are combining for better prediction. For example, to differentiate between the activities of 'talking on mobile phone' and 'drinking water through glass' can be recognized easily through the recognition of hand movement pattern by the temporal network at some spatial location. In contrast, the spatial network can identify the location of the head and their combination help to increase the overall prediction accuracy. In the proposed work, these techniques are defined using the scores of a decision level of Inception-Bi-LSTM stream (a) and DMI stream (b).

A general function f for fusing the two feature maps a and $b$at a given time t is denoted by Eq. (13).

$$f : x_t^a, x_t^b \rightarrow y_t \tag{13}$$

where $x_t^a \in \mathbb{R}^{H^a \times W^a \times D^a}$ and $x_t^a \in \mathbb{R}^{H^b \times W^b \times D^b}$ are two different features maps. The output feature map is denoted as: $y_t \in \mathbb{R}^{H \times W \times D}$, where W, H and D are represented width, height and number of the channel of respective feature maps. For simplicity, we assume that $W^a = W^b = W, H^a = H^b = H, D^a = D^b = D$.

The late fused score obtained by these approaches are denoted as y^{sum} y^{sum} (Sum), $y_{\max}$ (Maximum), and y^{cat} (Concatenation).

Sum fusion It calculates the sum $y^{\text{sum}} = f^{\text{sum}}\left(x^a, x^b\right)$ of two features maps in the feature channels d, at the same spatial location (i, j) and expressed using in Eq. (14).

$$y_{i,j,d}^{\text{sum}} = x_{i,j,d}^a + x_{i,j,d}^b \tag{14}$$

where $1 \le i \le H, 1 \le j \le W, 1 \le d \le D$ and $x^a, x^b, y \in \mathbb{R}^{HxWxD}$.

The sum fusion scores y^{sum} show a random correlation between the network layers.

Maximum fusion In this technique, $y^{\max} = f^{\max}\left(x^a, x^b\right)$ the maximum score is selected between the two feature maps. It defined using Eq. (15).

$$
\begin{aligned}
&I_1 \quad \oplus \quad I_2 \quad \oplus \quad I_3 \quad \cdots\cdots\cdots\cdots \quad I_{T-1} \quad \oplus \quad I_T \\
\Omega_1 \rightarrow\ & \frac{2*\mathbf{1}-T-1}{\mathbf{1}} \quad \frac{2*\mathbf{2}-T-1}{\mathbf{2}} \quad \frac{2*\mathbf{3}-T-1}{\mathbf{3}} \quad \cdots\cdots\cdots\cdots \quad \frac{2*(\boldsymbol{T}-\mathbf{1})-T-1}{\boldsymbol{T}-\mathbf{1}} \quad \frac{2*(\boldsymbol{T})-T-1}{\boldsymbol{T}} \\
\Omega_2 \rightarrow\ & \frac{2*\mathbf{2}-T-1}{\mathbf{2}} \quad \frac{2*\mathbf{3}-T-1}{\mathbf{3}} \quad \cdots\cdots\cdots\cdots \quad \frac{2*(\boldsymbol{T}-\mathbf{1})-T-1}{(\boldsymbol{T}-\mathbf{1})} \quad \frac{2*(\boldsymbol{T})-T-1}{\boldsymbol{T}} \\
\Omega_3 \rightarrow\ & \frac{2*\mathbf{3}-T-1}{\mathbf{3}} \quad \cdots\cdots\cdots\cdots \quad \frac{2*(\boldsymbol{T}-\mathbf{1})-T-1}{(\boldsymbol{T}-\mathbf{1})} \quad \frac{2*(\boldsymbol{T})-T-1}{\boldsymbol{T}} \\
&\vdots \qquad\qquad\qquad\qquad \vdots \qquad\qquad \vdots \\
\Omega_{T-1} \rightarrow\ & \frac{2*(\boldsymbol{T}-\mathbf{1})-T-1}{(\boldsymbol{T}-\mathbf{1})} \quad \frac{2*(\boldsymbol{T})-T-1}{\boldsymbol{T}} \\
\Omega_T \rightarrow\ & \frac{2*(\boldsymbol{T})-T-1}{\boldsymbol{T}}
\end{aligned}
$$

Fig. 5 The process of calculation of parameter Ω_T for finite length video sequences T. The bold part shows the dependency of parameter Ω_T on consecutive video frames $\in (1, T)$

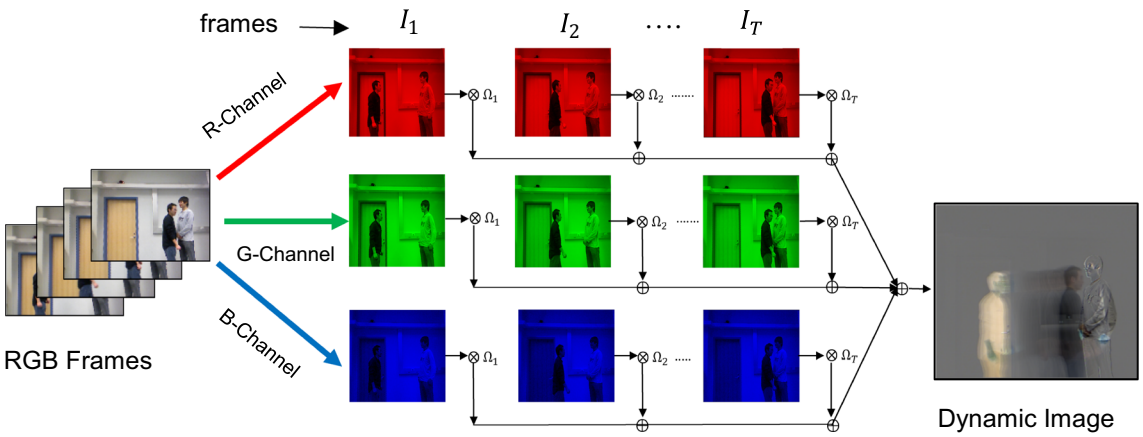

Fig. 6 The formation of dynamic motion image using approximate rank pooling mechanism from red, green and blue channel of each video frame (color figure online)

$$y^{\max}_{i,j,d} = \max\left\{x^a_{i,j,d} + x^b_{i,j,d}\right\} \tag{15}$$

where all the parameter representations are similar as in Eq. (14).

It increases accuracy as it incorporates best the predictions of both the models and this approach is used in our model.

Concatenation fusion In this fusion method, $y^{\text{cat}} = f^{\text{cat}}(x^a, x^b)$ features extracted from both streams are stacked across the feature channel d as:

$$y^{\text{cat}}_{i,j,2d} = x^a_{i,j,d} \quad y^{\text{cat}}_{i,j,2d-1} = x^b_{i,j,d} \tag{16}$$

where $y_t \in \mathbb{R}^{H\times W\times 2D}$, and the concatenation fusion does not show any correlation between the two feature maps.

4 Implementation details

Keras API deep learning library is used for the implementation of two streams of convolutional networks. This deep network architecture is trained on one NVIDIA GTX 2 GB graphic with 8 GB RAM, GPU machine. In the following section, model descriptions, training, results and comparison with similar state-of-the-art approaches on four standard RGB-D datasets are discussed in detail.

4.1 Datasets

To evaluate the performance of our two-stream network, we use the four standard datasets one is focused on a dyadic activity as SBU Interaction [51], single and human–object interaction activity datasets as MIVIA Action [33], MSR Daily Activity [52] and MSR Action Pairs [53]. The detailed description of these datasets such as a number of actions, actors, and challenges are explained in the following subsection.

4.1.1 The SBU Interaction Dataset

Yun et al. [51] introduced this interaction activity dataset. It is recorded with three different modalities RGB, depth frames, and 3D coordinates with the help of Kinect Sensor. It consists of seven subjects performing eight human–human interaction activities: departing (S1), approaching (S2), hugging (S3), Pushing (S4), kicking (S5), punching (S6), exchanging object (S7), and shaking hands (S8). The sample images from this dataset are depicted in Fig. 7a.

4.1.2 MIVIA Action

Carletti et al. [33] proposed this dataset intending to recognize human activities and human–object interaction in an indoor lab environment. This dataset is recorded with two different modalities RGB and depth frames with the of Kinect sensors. It contains 14 actors (7 females and 7 males) performing 7 activities such as: "drinking" (M1), "sleeping" (M2), "opening a jar" (M3), "sitting" (M4), "interacting with a table" (M5), "stopping" (M6), and "random motion" (M7). The sample images from this dataset are depicted in Fig. 7b.

4.1.3 MSR Daily Activity 3D Dataset

Wang et al. [52] proposed this dataset intending to recognize the daily use of human activity in an indoor room environment. It is recorded with two different modalities: RGB video and depth frames by a Kinect sensor. It contains 10 subjects doing 16 different daily activities: tossing paper (MD1), playing game (MD2), stand up (MD3), playing guitar (MD4), walking (MD5), using laptop (MD6), cheer up (MD7), using vacuum cleaner (MD8), calling on cellphone (MD9), sit still (MD10, drinking (MD11), lay down on sofa (MD12), reading book (MD13), writing on a paper (MD14), sitting down (MD15), and eating (MD16). All such activities are repeated twice for the sitting and standing position. The sample images from this dataset are shown in Fig. 7c.

4.1.4 MSR Action Pairs 3D

Oreifej and Liu [53] introduced this dataset consists of pairs of action videos. It is a challenging dataset because similar activities have the same shape and motion cues such as 'Put down' and 'Put up'. Ten subjects are performing 6 actions pairs: "Pushing and Pulling a chair" (MA1), "Putting and Taking off a backpack" (MA2), "Sticking and Removing a poster" (MA3), "Wearing and Taking off a hat" (MA4), "Lifting and Placing a box" (MA5), "Picking up and Putting down a box" (MA6). Each action was repeated three times by subjects in which 5 subjects are used for testing and 5 for training purposes. The sample image from this dataset is depicted in Fig. 7d.

4.2 Model parameter description and training settings

The proposed hybrid ConvNet is trained for two different data streams, i.e. RGB frames and dynamic image independently. The overfitting problem that occurs due to smaller training datasets in LSTM is compensated with the

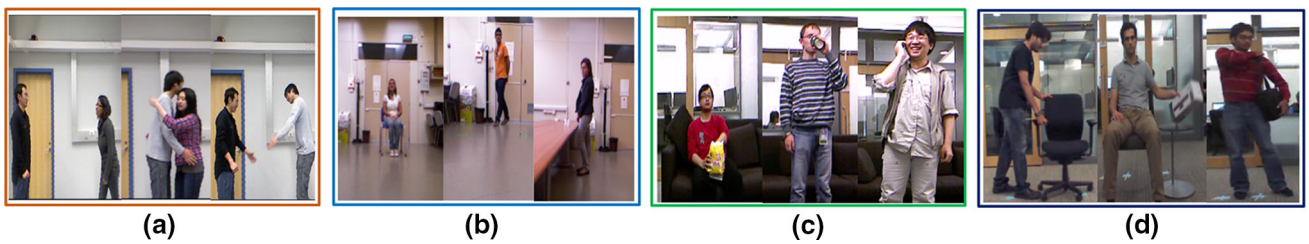

Fig. 7 Sample RGB frames from: **a** SBU Interaction Dataset **b** MIVIA Action Dataset **c** MSR Daily Activity 3D Dataset **d** MSR Action Pairs 3D Dataset

implementation of L2 regularization and dropout mechanism. The CNN-LSTM stream extracted the features from RGB frames that are fed with a batch size of 7 videos. The RGB stream is trained in an end to end fashion up to 150 epochs to refine the features of the pre-trained CNN. Initially, the learning rate of 10^{-4}, and a momentum constant equal to 0.9 is used for training the SGD optimizer. A recurrent dropout of 0.6 is used and added with each Bi-LSTM layer. In the SGD optimizer, the Minimum Square Error (MSE) loss function is selected for calculated the loss during training and test process.

Alternatively, the dynamic motion images are fine-tuned with last fully connected layers on the pre-trained CNN model. This DMIs stream is trained on CNN with a batch size of 8 videos up to 150 epochs in Adam Optimizer [54]. The initial learning rate and various parameters in Adam optimizer are used as: 10^{-4}, $\beta_1 = 0.9, \beta_2 = 0.999$, and $\varepsilon = 10^{-8}$, respectively, and 0.8 of dropout added after every fully connected layer. We used Adam optimizer for DMIs because it is easy to implement, computationally efficient, requires less memory as compared with other optimizers. It requires less tuning and suitable for non-static patterns objective. Figure 8 illustrates the training and test loss curves using MSE loss function for all four datasets. It is observed from the loss curves that our model attained a lower value of square error near 147 epochs.

All the datasets used in the evaluation of the algorithm is multiclass, and hence, we adopted a multiclass classification cross-validation scheme. The leave one out fivefold cross-validation multiclass classification scheme is applied for SBU dataset, where the number of test samples predicted as true class samples are defined as true positives and the test samples predicted as any of other negative classes of the action datasets is considered as false negatives or true negatives. For MIVIA Action Dataset, leave-one-subject-out (LOSO)-CV is applied. There is a total of 14 actors performing the 7 human activities. In this evaluation protocol, 13 actors are used for training and the remaining one actor is for testing. This process is repeating 14 times, always leaving another actor's data for testing. The evaluation method adopted for MSR Daily Activity and MSR Action pairs datasets in which half of the subject used for training and half of the subject for testing. The scores generated by both the softmax layers are fused using some late fusion techniques for prediction the final label activity.

4.3 Results analysis and comparisons

The proposed ConvNet model is tested on four datasets. The evaluation performance is measured in terms of average recognition accuracy (ARA) per class for many classes $(\mathcal{C}_i)$, which is calculated as in Eq. (17),

$$\mathrm{ARA} = \sum_{i=1}^{k} \frac{\frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{k} \tag{17}$$

where tp_i true positive of $(\mathcal{C}_i)$, tn_i true negative of $(\mathcal{C}_i)$, fp_i false positive of $(\mathcal{C}_i)$, fn_i false negative of $(\mathcal{C}_i)$.

4.3.1 The Mann–Whitney *U* test (Wilcoxon Rank Sum Test)

In order to have a better understanding of classification accuracy, a Wilcoxon rank-sum test is used to analyze the result. It is a nonparametric test [55] which is used to test whether two samples are similar in a distribution or not. The null hypothesis that comes from the same observation samples (i.e. have the equal median) or alternatively, whether observation in one sample tends to be greater than observed in the other samples. In the proposed work, we have tested the accuracies of both streams i.e. RGB frames and DMI using nonparametric test. Because both streams extracted the features from the same input video sequences independently. The null and alternative hypotheses for the test are stated:

H_0 = the two independent samples accuracy are the same verses,
H_1 = the two independent samples' accuracy is not the same.

This nonparametric test is conducted as a two-tailed test and observed that populations are not the same as opposed to specifying directionality. The test statistic for Wilcoxon rank-sum test is represented as U and is chosen from a minimum of U_{RGB} and U_{DMI} given by the following equations.

$$U_{\mathrm{RGB}} = n_1 n_2 - \frac{n_1(n_1+1)}{2} - \sum R_1 \tag{18}$$

$$U_{\mathrm{DMI}} = n_1 n_2 - \frac{n_2(n_2+1)}{2} - \sum R_2 \tag{19}$$

where $\sum R_1$, and $\sum R_2$ are the sum of the ranks for RGB Frame samples and DMI samples, respectively, n_1 is the total samples of RGB frame accuracy and n_2 is the total samples of DMI accuracy as illustrated in Tables 1, 2, 3 and 4.

Each time we tested and observed the value of U (U_{Stat}) whether it supports the null or alternative hypothesis like parametric testing. Further, we have determined the critical value ($U_{\mathrm{Cri}(0.05\,\mathrm{or}\,0.01)}$) and compared with a minimum value of U_{Stat}. If the critical value is higher, then we reject the null hypothesis H_0 and if the U_{Stat} value is higher than the critical value we reject the alternate hypothesis H_1 i.e.

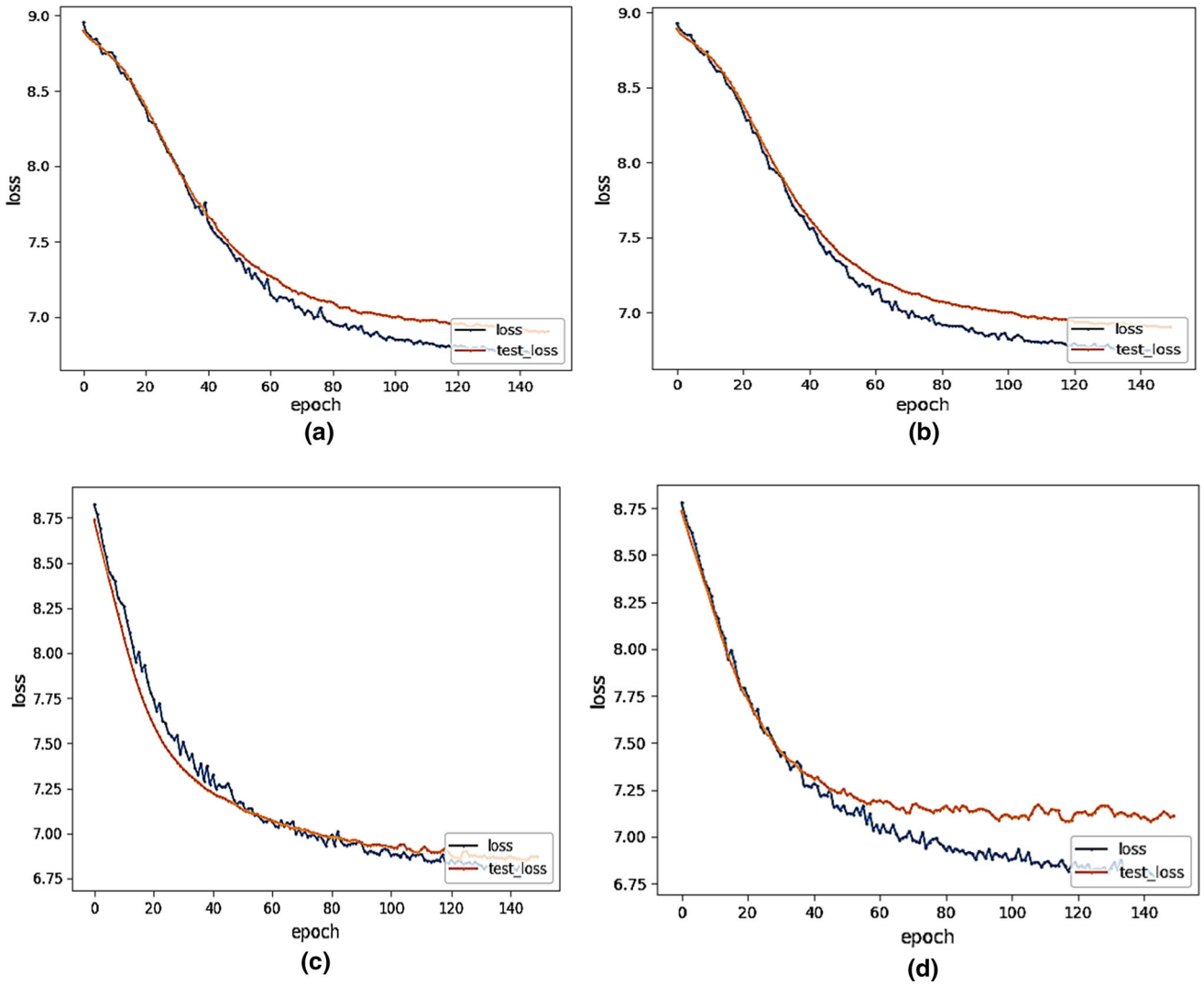

Fig. 8 Shows the training minimum squared (MSE) loss and test loss for activity datasets: **a** SBU Interaction **b** Mivia Action **c** MSR Action Pairs **d** MSR Daily Activity

Table 1 Activity-wise results of RGB frames and DMI streams on SBU Interaction Dataset

Activity	S1	S2	S3	S4	S5	S6	S7	S8
Accuracy (%)								
RGB	82	84	72	71	72	73	76	83
DMI	98	86	95	96	92	94	82	88

Table 2 Activity-wise results of RGB frames and DMI streams on MIVIA Action Dataset

Activity	M1	M2	M3	M4	M5	M6	M7
Accuracy (%)							
RGB	82	74	92	85	76	83	79
DMI	95	94	96	89	94	97	84

$U_{\text{Stat}} = \min(U_{\text{RGB}}, U_{\text{DMI}})$; and if $U_{\text{Stat}} < U_{\text{Cri}(0.05\,\text{or}\,0.01)}$, then H_0 (Rejected), and (Accepted).

It can be observed from Table 5 that for each dataset the null hypothesis H_0 is rejected and alternate hypothesis H_1 is accepted because in the value of $U_{\text{Stat}} < U_{\text{Cri}(0.05\,\text{or}\,0.01)}$. Therefore, our observation on the samples from independent input streams such as RGB and DMI is not accepted in the same way according to a two-tailed analysis of Wilcoxon rank-sum hypothesis.

The result obtained by various late fusion techniques is compared to four datasets in Table 6. It can be observed that max fusion and sum fusion techniques give high scores as compared to average and concatenation fusion technique. The max fusion yields highest recognition accuracy

Table 3 Activity-wise results of RGB frames and DMI streams on MSR Action Pairs Dataset

Activity	MA1	MA2	MA3	MA4	MA5	MA6
Accuracy (%)						
RGB	91	88	90	87	84	86
DMI	97	96	98	95	87	96

at softmax layers because it selects the maximum probability from both of the softmax prediction scores and assigns a label activity in correspondence to that probability. The highest accuracy is highlighted with a bold letter.

The comparison of average recognition accuracy (ARA) achieved on these datasets with two different input data stream as RGB frames, and single dynamic motion image (DMI) and the max fusion scores of RGB with DMI is shown in Fig. 9. It is clear from Fig. 9, the max fusion of both RGB with DMI gives the best results as compared with independent input data streams: RGB and single dynamic motion image. It can be seen from work Bilen et al. [9] that the dynamic image obtained with approximate rank pooling with CNN showed excellent results in many indoor/outdoor activities recognition tasks in videos. Further, features extracted from RGB only frames are not sufficient to represent the complex activities of human activities. Therefore, the proposed approach fused the discriminative features extracted from DMI and RGB frames to represents spatiotemporal variation in activity video. The proposed model shows excellent results on all four video benchmarks. The results obtained on the MSR Daily activity dataset show somewhat less accuracy as compared with SBU Interaction, MIVIA Action and MSR Action pairs due to intra-class similarity exists between activity classes and complex background.

Table 4 Activity-wise results of RGB frames and DMI streams on MSR Daily Activity

Activity	MD1	MD2	MD3	MD4	MD5	MD6	MD7	MD8
Accuracy (%)								
RGB	81	67	82	80	71	66	82	84
DMI	88	72	90	87	79	72	94	90
Activity	MD9	M10	MD11	MD12	MD13	MD14	MD15	MD16
Accuracy (%)								
RGB	72	73	78	84	80	67	80	79
DMI	80	81	86	91	89	72	88	84

Table 5 Wilcoxon rank-sum test (two-tailed test) research hypothesis results on activity datasets

Dataset	Samples	U_{RGB}	U_{DMI}	U_{Stat}	$U_{Critical}$	
	$n_1(RGB) = n_2(DMI)$				$\alpha = 0.05$	$\alpha = 0.01$
SBU Interaction	8	61.50	2.5	2.5	13	7
Mivia Action	7	46	3	3	8	4
MSR Action Pairs	6	32.5	3.5	3.5	5	2
MSR Daily Activity	16	201	55	55	75	60

Table 6 Comparison of accuracy (%) of different fusion techniques on human activity datasets

Dataset	RGB + DMI late fusion scores			
	Sum fusion	Average fusion	Concatenation fusion	Max fusion
SBU	98.10	96.45	96.50	**98.70**
MIVIA	98.40	97.20	95.75	**99.41**
MSR Action Pair	96.90	94.60	94.80	**98.30**
MSR Daily Activity	**95.36**	90.40	91.60	94.37

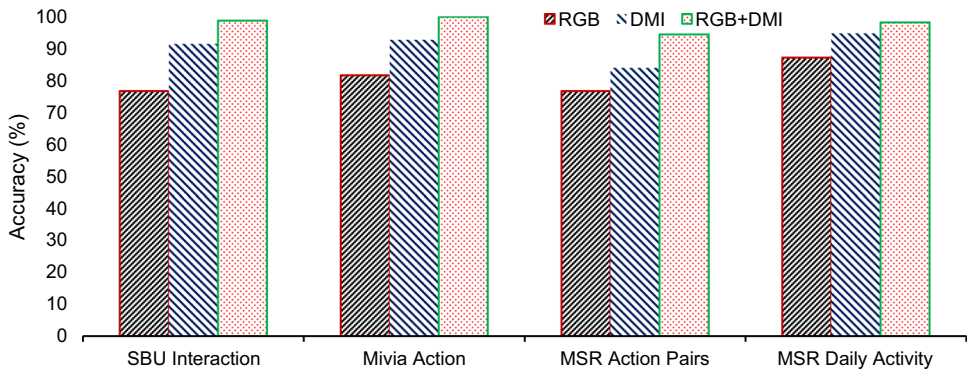

Fig. 9 Results on four datasets with different data inputs: only RGB frames, dynamic motion image (DMI) and RGB + DMI. The accuracy obtained using a max fusion of RGB + DMI outperforms over RGB and DMI inputs

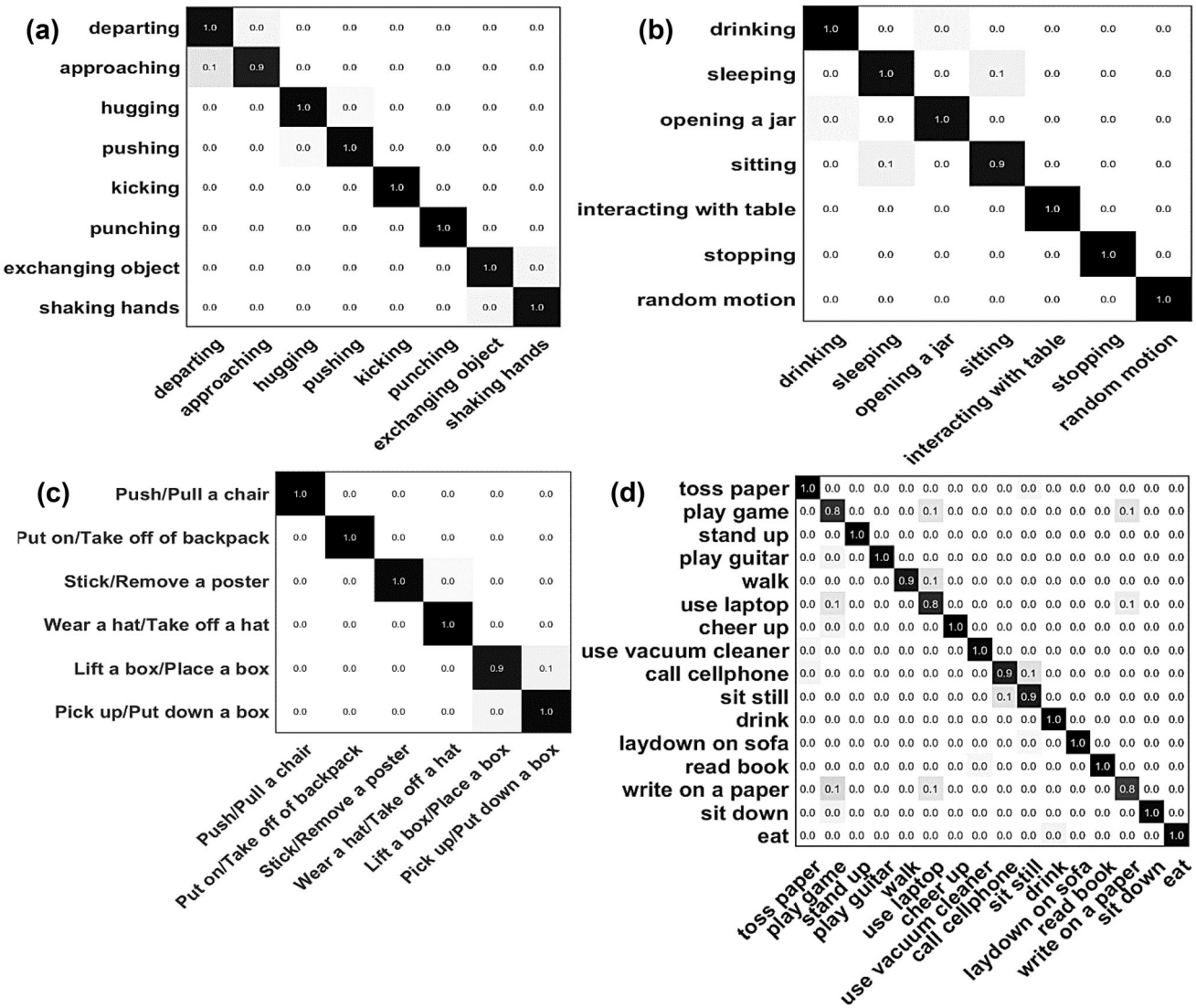

Fig. 10 The confusion matrix of datasets: **a** SBU Interaction, **b** MIVIA Action, **c** MSR Action Pairs, **d** MSR Daily Activity

The classification result on these datasets is shown in the form of a confusion matrix is as shown in Fig. 10. The confusion matrix of SBU dataset is shown in Fig. 10a. It is a challenging dataset because of the similarity of action and the same motion cues in video frames. It is clear from the confusion matrix that the main confusion exists with two similar activities such as "approaching" and "departing" and "handshaking" and "exchanging object." The

Table 7 State-of-the-art comparison of SBU Interaction Dataset

Works	Methods	Accuracy (%)
Yun et al. [51]	SVM and MIL Boost	87.30
Feng et al. [26]	BiVector and LSTM	82.70
Ijjina and Chalavadi [43]	ConvNet + ELM classifier	86.58
Keçeli et al. [42]	2D and 3D ConvNet	94.70
Elboushaki et al. [46]	MultiD-CNN	97.51
Proposed approach (RGB)	Inception-v3 + Bi-LSTM	76.60
Proposed approach (DMI)		91.40
Proposed approach (RGB + DMI) max fusion		**98.70**

Table 8 State-of-the-art comparison of MIVIA Action Dataset

Works	Method	Accuracy (%)
Carletti et al. [33]	Reject	79.80
Foggia et al. [56]	Deep learning	84.70
Brun et al. [57]	Edit distance	85.20
Ijjina and Chalavadi [43]	ConNnets + ELM classifier	93.37
Saggese et al. [34]	Skeleton feature	95.00
Brun et al. [32]	Aclets sequences	95.40
Proposed approach (RGB)	Inception-v3 + Bi-LSTM	81.59
Proposed approach (DMI)		92.71
Proposed approach (RGB + DMI) max fusion		**99.41**

state-of-the-art comparison of similar works on SBU dataset is listed in Table 7. Our deep model is evaluated on SBU dataset with fivefold cross-validation similar in the work [51]. The recognition accuracy of the proposed approach with two-stream fusion shows the best accuracy (bold text) on this dataset as shown in Table 7.

The experimental result of MIVIA action and comparison with similar approaches are shown in Table 8 and the accuracy achieved by proposed approach is highlighted in bold text. The LOSO cross-validation evaluation protocol is followed for training and testing the proposed model similar in [33]. It is observed from the confusion matrix shown in Fig. 10b that actions having no motion in the video found confusion for recognize for example "sitting" and "sleeping" activities. The spatial features for both the activities are the same but lack of temporal cue due to no movement of actors. In this situation, DMI stream features extraction is not so useful for activity prediction. However, the proposed ConvNet model obtained the best results with the fusion of CNN-LSTM and dynamic images as compared with existing methods.

The result comparison of MSR Action Pairs is shown in Table 9 and the accuracy achieved by proposed approach is highlighted in bold text. The evaluation protocol is followed as in similar works by Wang et al. [52]. Out of total performer, half of the actors are used for training the model and half of the actors for testing the action classes. It is observed that from the confusion matrix in Fig. 10c that all the six action pairs are correctly recognized by the proposed approach and no occurrence of intra-class pair confusion. There is slight confusion between "lift a box" and "put down a box" action pairs but our hybrid model

Table 9 State-of-the-art comparison of MSR Action Pairs Dataset

Works	Method	Accuracy (%)
Wang et al. [52]	LOP	82.22
Jia et al. [58]	LTTL	91.40
Oreifej and Liu [53]	HON4D	93.33
Vemulapalli and Chellapa [59]	FTP representation	94.67
Ji et al. [23]	One-shot learning	95.10
Ji et al. [22]	Skeleton embedded feature	97.70
Proposed approach (RGB)	Inception-v3 + Bi-LSTM	87.70
Proposed approach (DMI)		94.76
Proposed approach (RGB + DMI) max fusion		**98.30**

Table 10 State-of-the-art comparison of MSR Daily Activity Dataset

Works	Method	Accuracy (%)
Amor et al. [25]	Rate-invariant analysis	70.00
Seidenari et al. [60]	NBNN bag-of-poses	70.00
Cai et al. [61]	Markov random field	78.20
Zhang and Parker [62]	BIPOD	79.70
Ji et al. [22]	Skeleton embedded feature	81.30
Jing et al. [44]	Joint loss function	88.00
Srihari et al. [45]	4-stream CNN	89.05
Huynh-The et al. [63]	PAM + pose-transition	90.63
Proposed approach (RGB)	Inception-v3 + Bi-LSTM	76.64
Proposed approach (DMI)		83.90
Proposed approach (RGB + DMI) max fusion		**94.37**

shows good performance to recognized each action pairs activity.

The confusion matrix for MSR Daily Activity Dataset is shown in Fig. 10d. It is seen from the confusion matrix that the main confusion occurs in similar activities such as play game using a laptop, and write on the paper. Most of the activities are correctly classified with a high confidence level. As illustrated in Table 10, our proposed model performed well and achieved superior accuracy with similar state-of-the-art approaches, which is highlighted in bold text.

5 Conclusion

In this paper, a robust two-stream deep ConvNet model is developed for the recognition of single, multi-person and human–object interaction (HOI) activities in the video sequence. This model uses two deep learned architectures: First is the pre-trained CNN and Bi-LSTM to extract the discriminative features from the given RGB frames, and second is a pre-trained CNN fine-tuned with fully connected layers fed with dynamic motion image as an input. It is observed that the activities which do not have motion, the CNN-Bi-LSTM combination, classify the activity classes with better recognition accuracy. On the other hand, the activities which have high motion, the dynamic images are used to boost the prediction with the CNN-LSTM stream after late fusion at softmax layer. The prediction accuracy is computed on four publically available video benchmarks such as SBU (98.70%), MIVIA (99.41%), MSR Action Pairs (98.30%), and MSR Daily Activity (94.37%). The comparisons with other state of the art are outlined for the proposed deep architecture proving the dominance of the framework in terms of accuracy.

In future work, depth frames along with 3D skeleton coordinate information and multi-view actions classes may be used to make the action prediction more dynamic to intra-class-life applications. It may also be applied for real-time detection of human activities, and other useful applications such as crowd anomaly detection, sports actions classification, and development of intelligent surveillance system, etc.

Acknowledgements The authors would like to acknowledge the computation of the work was supported by Biometric Research Laboratory, Department of Information Technology, Delhi Technological University, New Delhi, India.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Aggarwal JK, Xia L (2013) Human activity recognition from 3D data—a review. Pattern Recognit Lett 48:70–80
2. Dhiman C, Vishwakarma DK (2018) A review of state-of-the-art techniques for abnormal human activity recognition. Eng Appl Artif Intell 77:21–45
3. Suto J, Oniga S, Lung C, Orha I (2018) Comparison of offline and real-time human activity recognition results using machine learning techniques. Neural Comput Appl 1–14
4. Vishwakarma DK, Kapoor R, Maheshwari R, Kapoor V, Raman S (2015) Recognition of abnormal human activity using the changes in orientation of silhouette in key frames. In: IEEE international conference on computing for sustainable global development (INDIACom), New Delhi
5. Schuldt C, Laptev I, Caputo B (2004) Recognizing human actions: a local SVM approach. In: 17th International conference on pattern recognition
6. Vishwakarma DK, Kapoor R (2015) Integrated approach for human action recognition using edge spatial distribution, direction pixel, and R-transform. Adv Robot 29(23):1551–1561
7. Singh T, Vishwakarma DK (2018) Video benchmarks of human action datasets: a review. Artif Intell Rev 52(2):1107–1154
8. Zhang J, Li W, Ogunbona PO, Wang P, Tang C (2016) RGB-D based action recognition datasets: a survey. Pattern Recognit 60:86–105

9. Bilen H, Fernando B, Gavves E, Vedaldi A, Gould S (2016) Dynamic image networks for action recognition. In: IEEE international conference on computer vision and pattern recognition (CVPR), Las Vegas, NV, pp 3034–3042
10. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2015) Rethinking the inception architecture for computer vision, arXiv:1512.00567 [cs.CV]
11. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 1097–1105
12. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556
13. Herath S, Harandi M, Porikli F (2017) Going deeper into action recognition: a survey. Image Vis Comput 60:4–21
14. Ladjailia A, Bouchrika I, Merouani H, Harrati N, Mahfouf Z (2019) Human activity recognition via optical flow: decomposing activities into basic actions. Neural Comput Appl 1–14
15. Wang H, Klaeser A, Schmid C, Liu C-L (2013) Dense trajectories and motion boundary descriptors for action recognition. IJCV 103:60–79
16. Liu J, Luo J, Shah M (2009) Recognizing realistic actions from videos "in the Wild". In: IEEE international conference on computer vision and pattern recognition (CVPR)
17. Vishwakarma DK, Singh K (2016) Human activity recognition based on spatial distribution of gradients at sub-levels of average energy silhouette images. IEEE Trans Cogn Dev Syst 99:1
18. Dhiman C, Vishwakarma DK (2019) A robust framework for abnormal human action recognition using R-transform and Zernike moments in depth videos. IEEE Sens J 19(13):5195–5203
19. Baccouche M, Mamalet F, Wolf C, Garcia C, Baskurt A (2011) Sequential deep learning for human action recognition. In: Proceedings of the second international conference on human behavior understanding
20. Ji S, Xu W, Yang M, Yu K (2013) 3D convolutional neural networks for human action recognition. IEEE Trans Pattern Anal Mach Intell 35(1):221–231
21. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: Proceedings of the advances in neural information processing systems
22. Ji X, Cheng J, Feng W, Tao D (2017) Skeleton embedded motion body partition for human action recognition using depth sequences. Sig Process 143:56–68
23. Ji Y, Yang Y, Xu X, Shen HT (2018) One-shot learning based pattern transition map for action early recognition. Sig Process 143:364–370
24. Fernando B, Gavves E, Oramas M, Ghodrati A, Tuytelaars T (2015) Modeling video evolution for action recognition. In: IEEE international conference on computer vision and pattern recognition (CVPR)
25. Amor BB, Su J, Srivastava A (2016) Action recognition using rate-invariant analysis of skeletal shape trajectories. IEEE Trans Pattern Anal Mach Intell 38(1):1–13
26. Feng J, Zhang S, Xiao J (2017) Explorations of skeleton features for LSTM-based action recognition. Multimed Tools Appl 78:591–603
27. Bobick AF, Davis JW (2001) The recognition of human movement using temporal templates. IEEE Trans Pattern Anal Mach Intell 23(3):257–267
28. Blank M, Gorelick L, Shechtman E, Irani M, Basri R (2005) Actions as space-time shapes. In: Tenth IEEE international conference on computer vision (ICCV'05), Beijing
29. Gorelick L, Blank M, Shechtman E, Irani M, Basri R (2007) Actions as space-time shapes. Trans Pattern Anal Mach Intell 29:2247–2253
30. Laptev I (2005) On space-time interest points. Int J Comput Vision 64(2–3):107–123
31. Matikainen P, Hebert M, Sukthankar R (2009) Trajectons: action recognition through the motion analysis of tracked features. In: IEEE 12th international conference on computer vision
32. Brun L, Percannella G, Saggesea A, Vento M (2016) Action recognition by using kernels on aclets sequences. Comput Vis Image Underst 144:3–13
33. Carletti V, Foggia P, Percannella G, Saggese A, Vento M (2013) Recognition of human actions from RGB-D videos using a reject option. In: International workshop on social behaviour analysis
34. Saggese A, Strisciuglio N, Vento M, Petkov N (2018) Learning skeleton representations for human action recognition. Pattern Recognit Lett 118:23–31
35. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: Proceedings of the European conference on computer vision
36. Laptev I, Lindeberg T (2004) Local descriptors for spatio-temporal recognition. In: ECCV workshop on spatial coherence for visual motion analysis
37. Rodriguez MD, Ahmed J, Shah M (2008) Action MACH: a spatio-temporal maximum average correlation height filter for action recognition. In: IEEE conference on computer vision and pattern recognition, Anchorage, AK
38. Al-Nawashi M, Al-Hazaimeh O, Saraee M (2017) A novel framework for intelligent surveillance system based on abnormal human activity detection in academic environments. Neural Comput Appl 28:565–572
39. Wang H, Schmid C (2013) Action recognition with improved trajectories. In: Proceedings of the international conference on computer vision (ICCV)
40. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L, (2014) Large-scale video classification with convolutional neural networks. In: IEEE conference on computer vision and pattern recognition, Columbus, OH
41. Peng X, Zou C, Qiao Y, Peng Q (2014) Action recognition with stacked fisher vectors. In: ECCV
42. Keçeli AS, Kaya A, Can AB (2018) Combining 2D and 3D deep models for action recognition with depth information. SIViP 12:1197–1205
43. Ijjina EP, Chalavadi KM (2017) Human action recognition in RGB-D videos using motion sequence information and deep learning. Pattern Recognit 72:504–516
44. Jing C, Wei P, Sun H, Zheng N (2019) Spatiotemporal neural networks for action recognition based on joint loss. Neural Comput Appl 32:4293–4302
45. Srihari D, Kishore PVV, Kumar EK, Kumar A, Kumar MTK, Prasad MVD, Prasad CR (2020) A four-stream ConvNet based on spatial and depth flow for human action classification using RGB-D data. Multimed Tools Appl 79:11723–11746. https://doi.org/10.1007/s11042-019-08588-9
46. Elboushaki A, Hannane R, Afdel K, Koutti L (2020) MultiD-CNN: a multi-dimensional feature learning approach based on deep convolutional networks for gesture recognition in RGB-D image. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2019.112829
47. Williams RJ, Hinton GE, Rumelhart DE (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
48. Hochreiter S, Schnidhuber J (1997) Long short-term memory. Neural Comput 9(1997):1735–1780
49. Smola AJ, Scholkopf B (2004) A tutorial on support vector regression. Stat Comput 14:199–222
50. Feichtenhofer C, Pinz A, Zisserman A (2016) Convolutional two-stream network fusion for video action recognition, arXiv:1604.06573v2 [cs.CV]
51. Yun K, Honorio J, Chattopadhyay D, Berg TL, Samaras D (2012) Two-person interaction detection using body-pose features and

multiple instance learning. In: IEEE international conference computer vision and pattern recognition workshops (CVPRW), Rhode Island
52. Wang J, Liu Z, Wu Y, Yuan J (2012) Mining Actionlet ensemble for action recognition with depth cameras. In: IEEE conference on computer vision and pattern recognition, Rhode Island
53. Oreifej O, Liu Z (2013) HON4D: histogram of oriented 4D normals for activity recognition from depth sequences. In: IEEE international conference on computer vision and pattern recognition (CVPR), Portland, OR
54. Kingma PD, Ba JL (2015) ADAM: a method for stochastic optimization. In: International conference on learning representations, San Diego
55. Mann HB, Whitney DR (1947) On a test of whether one of two random variables is stochastically larger than the other. Ann Math Stat 18(1):50–60
56. Foggia P, Saggese A, Strisciuglio N, Vento M (2014) Exploiting the deep learning paradigm for recognizing human actions. In: IEEE AVSS
57. Brun L, Foggia P, Saggese A, Vento M (2015) Recognition of human actions using edit distance on aclet strings. In: VISAPP
58. Jia C, Kong Y, Ding Z, Fu Y (2014) Latent tensor transfer learning for RGB-D action recognition. In: Proceedings of the 22nd ACM international conference on multimedia, Orlando, FL, USA
59. Vemulapalli R, Chellapa R (2016) Rolling rotations for recognizing human actions from 3D skeletal data. In: IEEE international conference on computer vision and pattern recognition (CVPR)
60. Seidenari L, Varano V, Berretti S, Bimbo AD, Pala P (2013) Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In: IEEE international conference on computer vision and pattern recognition (CVPR), Portland
61. Cai X, Zhou W, Wu L, Luo J, Li H (2016) Effective active skeleton representation for low latency human action recognition. IEEE Trans Multimed 18(2):141–154
62. Zhang H, Parker LE (2015) Bio-inspired predictive orientation decomposition of skeleton trajectories for real-time human activity prediction. In: IEEE international conference on robotics and automation (ICRA), Seattle, WA
63. Huynh T-T, Hua C-H, Tu NA, Hur T, Bang J, Kim D, Amin MB, Kang BH, Seung H, Shin S-Y, Kim E-S, Lee S (2018) Hierarchical topic modeling with pose-transition feature for action recognition using 3D skeleton data. Inf Sci 444:20–35