IOP Conference Series: Materials Science and Engineering

PAPER • OPEN ACCESS

# Deep Positional Attention-based Bidirectional RNN with 3D Convolutional Video Descriptors for Human Action Recognition

To cite this article: N Srilakshmi and N Radha 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1022** 012017

View the article online for updates and enhancements.

# Deep Positional Attention-based Bidirectional RNN with 3D Convolutional Video Descriptors for Human Action Recognition

**N Srilakshmi[1] and N Radha[2]**

[1]Ph.D. Scholar, Department of Computer Science, PSGR Krishnammal College for Women, Coimbatore, Tamilnadu, India
[2]Associate Professor, Department of Computer Science, PSGR Krishnammal College for Women, Coimbatore, Tamilnadu, India

srilakshmi.mphil@gmail.com

**Abstract**. This article presents the Joints and Trajectory-pooled 3D-Deep Positional Attention-based Bidirectional Recurrent convolutional Descriptors (JTPADBRD) for recognizing the human activities from video sequences. At first, the video is partitioned into clips and these clips are given as input of a two-stream Convolutional 3D (C3D) network in which the attention stream is used for extracting the body joints locations and the feature stream is used for extracting the trajectory points including spatiotemporal features. Then, the extracted features of each clip is needed to aggregate for creating the video descriptor. Therefore, the pooled feature vectors in all the clips within the video sequence are aggregated to a video descriptor. This aggregation is performed by using the PABRNN that concatenates all the pooled feature vectors related to the body joints and trajectory points in a single frame. Thus, the convolutional feature vector representations of all the clips belonging to one video sequence are aggregated to be a descriptor of the video using Recurrent Neural Network (RNN)-based pooling. Besides, these two streams are multiplied with the bilinear product and end-to-end trainable via class labels. Further, the activations of fully connected layers and their spatiotemporal variances are aggregated to create the final video descriptor. Then, these video descriptors are given to the Support Vector Machine (SVM) for recognizing the human behaviors in videos. At last, the experimental outcomes exhibit the considerable improvement in Recognition Accuracy (RA) of the JTDPABRD is approximately 99.4% achieved on the Penn Action dataset as compared to the existing methods.

## 1. Introduction

Human Activity Recognition (HAR) is the method of using the videos that include a specific activity and recover videos of interest to identify an individual's conduct. This has been applied in potential fields including video processing, human-computer interface design, medical services and so on. By the day, an incredible number of videos are generated due to monitoring devices, media, YouTube and others. Correspondingly, HAR is significant in the field of machine learning in the current era. Many deep learning models were suggested based on either supervised or unsupervised learning algorithms that can support HAR systems [1-3].

Among many deep learning methods, Joints-pooled 3D-Deep convolutional Descriptors (JDD) [4-5] has better efficiency by aggregating the convolutional activations of the 3D-deep Convolutional Neural

Network (3DCNN) into the discriminative descriptors based on the joint locations. On the contrary, the estimation of joints locations takes more time for a huge dataset and also the estimation of skeletons has a high complex. As a result, Joints and Trajectory-pooled 3D-Deep convolutional Descriptors (JTDD) is suggested [6] that extracts both body joints and trajectory points between two video sequences by multiplying two C3D streams: feature and attention with the bilinear product function. Also, the pooled descriptors are generated to extracting the spatiotemporal features together. Then, the video descriptors are obtained by training the whole network in an end-to-end manner according to the class labels. Moreover, these video descriptors are classified via the SVM to recognize individual behaviors. Nonetheless, the max-min poling was applied as the feature aggregation method that has high flexibility to spatially smooth over the adjacent kernels. This eliminates the necessary spatiotemporal variances between class labels.

Therefore in this article, JTDPABRD is proposed that integrates the PABRNN model into a two-stream C3D network to extract significant spatiotemporal features and increase the accuracy of recognizing individual activities. Initially, the video is split into many clips and these clips are fed to the two-stream C3D network as input. In a two-stream C3D network, the attention stream is used to extract the guidance of body joints locations and the feature stream is used to extract the trajectory points along with significant spatiotemporal features. After, each convolutional feature vector representations of each clip belonging to the single video are aggregated using the PABRNN to create the clip descriptor. Also, these two streams are multiplied by the bilinear product and end-to-end trained via class labels. Moreover, the activations of fully connected layers and their spatiotemporal variances are also aggregated to generate the final video descriptor. This video descriptor is applied to the SVM to recognize the individual activities in video sequences. Thus, the accuracy of recognizing human activities is increased efficiently.

## 2. Literature Survey

Rahman et al. [7] investigated the HAR system using textural features with classical shape and motion features from low-quality videos. But, it needs to learn the richer features from video sequences for enhancing the performance. Li et al. [8] proposed a novel two-layer framework for HAR via defining the video with low-level local and mid-level motion features. However, several groups in the video were not represented the activity part and it cannot determine the number of groups in various datasets which affects the efficiency of mid-level encoding.

Jin et al. [9] proposed a multilevel action descriptor that provides absolute information on human activities. But, it was not able to learn deep motion flow from the video sequences. Shou et al. [10] suggested a lightweight generator network to get more Discriminative Motion Cue (DMC) for HAR. Conversely, it has less accuracy. Huo et al. [11] proposed the new mobile HAR system, but it needs to consider the attention scheme for further improving the accuracy.

Nida et al. [12] proposed a feedforward learning method for recognizing the instructor's action in the classroom. However, an overfitting problem occurred while increasing the hidden layers. Sudhakaran et al. [13] proposed a Long Short-Term Attention (LSTA) to extract the features from relevant spatial parts and recognize the egocentric activity. But, the accuracy was less.

## 3. Proposed Methodology

This section explains the JTDPABRD method in brief. The block diagram of the JTDPABRD method is depicted in Figure 1.

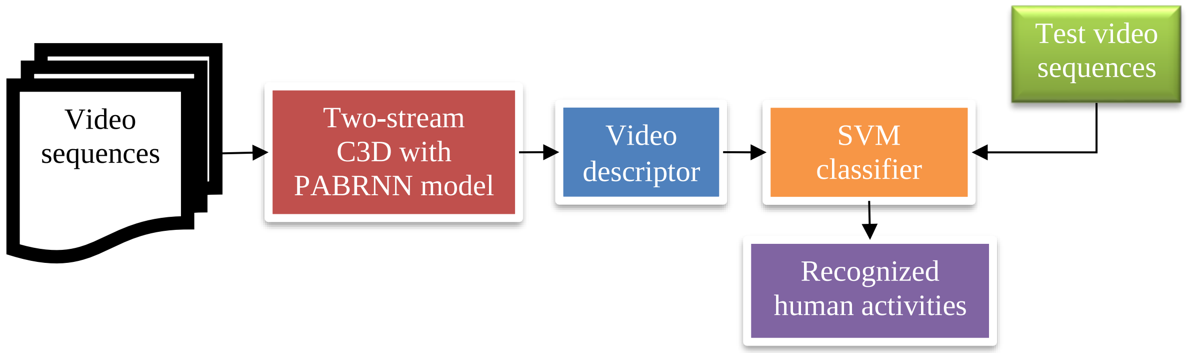

**Figure 1.** Block diagram of JTDPABRD based HAR system.

Originally, each video sequence is split into many clips or frames and given as input to the two-stream C3D network. In this network, the input is given to the attention stream and feature stream, accordingly. The attention stream is used for extracting the guidance of body joint locations and the feature stream is used for extracting the trajectory points or optical flow between each clip including spatiotemporal features. The activations of each corresponding body joint location and trajectory point are pooled from each channel. To obtain the pooled feature vectors belonging to one clip, RNN, namely JTDRD method is applied instead of max-min pooling. But, the standard RNN has a problem of how to aggregate network outputs in an optimized manner as various networks trained on similar data can no longer be regarded as independent. Therefore, JTD-Bidirectional RNN-Descriptor (JTDBRD) is applied to solve the problems in the standard RNN and trained using all available input information in the past and future of particular time frames i.e., video clips. The concept is splitting the state neurons of a standard RNN in a part for both forward and backward states. The results from forward-states are not linked to inputs of backward-states and vice versa. Using both states, input information in the previous and the future of the currently estimated frames can be directly used for reducing the objective function without the requirement for delays to include future information.

This BRNN can be trained with similar algorithms as a standard RNN since there are no interactions between two kinds of state neurons and so can be extended into the common feed-forward network, Few specific solutions are required only at the beginning and the end of training samples. The forward-state input at $t = 1$ and the backward-state inputs $t = T$ are not observed. But, they are set randomly to a predetermined value (0.5). Also, the local state derivatives at $t = T$ for the forward-states and at $t = 1$ for the backward-states are not known and are set to 0, considering that the information beyond that point is not significant for the current update. On the other hand, BRNN cannot be used for providing significant feature vectors with the highest likelihood. Also, the problem of BRNN is how to aggregate the hidden vectors for feature representations. As a result, PABRNN is proposed in this JTDPABRD method that assumes if a feature in one video frame occurs in another video frame, it will have guidance on the adjacent context. In other words, the adjacent features should be given more attention that those far away since they may include more body joint and trajectory relevant information. The entire trainable end-to-end two-stream C3B with the PABRNN framework is depicted in Figure 2.

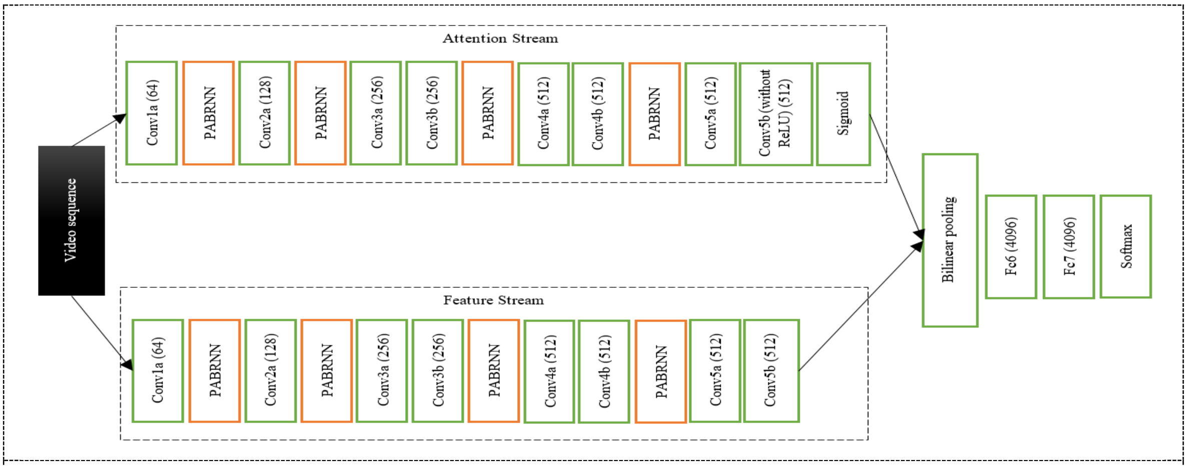

**Figure 2.** Block diagram of two-stream bilinear C3D with PABRNN-based feature aggregation method.

### *3.1. PABRNN model*

This PABRNN adopts the BRNN for feature vector representation which takes the pre-trained body joints and trajectory points embeddings as the input and creates the hidden vectors by recurrent updates. To aggregate the feature vector representations, the standard attention is used which especially relies on the hidden vectors for the attentive weight generation. For this purpose, a positional attention scheme is proposed and additional steps are performed based on the standard attention as:

- Discover the occurrence feature positions in each clip related to a single video sequence.
- Propagate the guidance of feature vectors to other positions with a position-aware guidance propagation approach.
- Create the position-aware guidance vector for every feature in clips according to the propagated guidance.
- Combine the position-aware guidance vector into the standard attention scheme.

By using the attentive representations of both original and aggregated feature vectors, different similarity functions are used for measuring the relevance between each dimension. The Manhattan distance similarity function $(sim)$ is used with $l_1$-norm as:

$$sim(F, F_a) = e^{-(\|F-F_a\|_1)} \qquad (1)$$

In Eq. (1), $F$ and $F_a$ are the original feature vector and aggregated feature vectors corresponding to in each clip and $\|\cdot\|_1$ is the $l_1$-norm. The structure of this PABRNN model is illustrated in Figure 3.

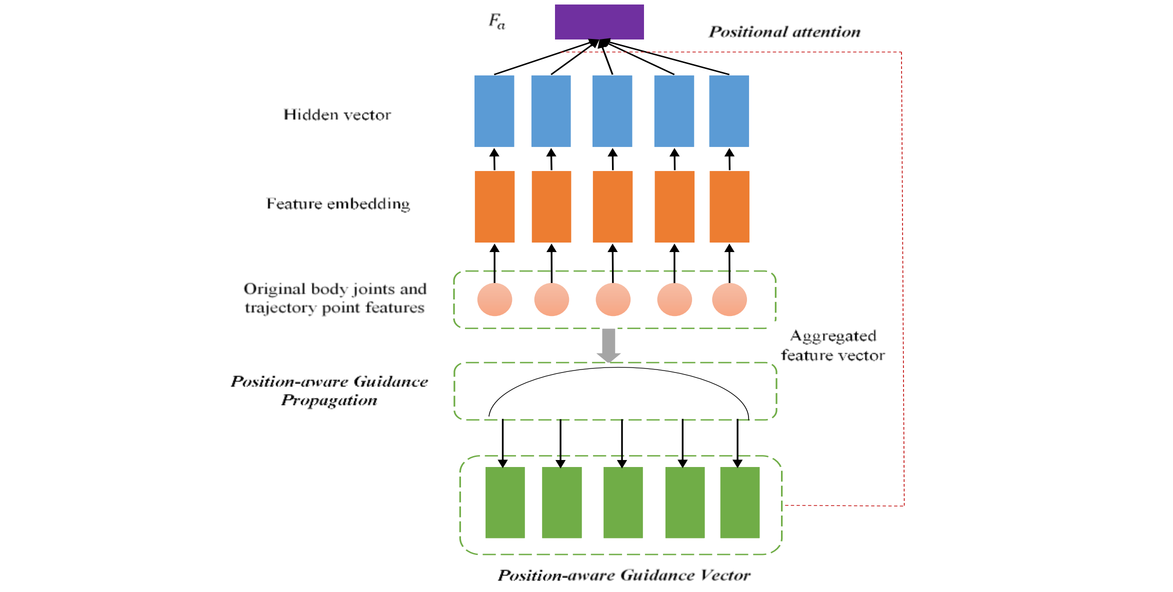
$F_a$
Positional attention
Hidden vector
Feature embedding
Original body joints and trajectory point features
Aggregated feature vector
Position-aware Guidance Propagation
Position-aware Guidance Vector

**Figure 3.** Feature vector representation with PABRNN model.

### *3.2. Position-aware guidance propagation*

Based on the above consideration, the features will have guidance on the adjacent context if it occurs in other clips. Here, the position-aware guidance propagation is modeled with the Gaussian kernel as:

$$Kernel(d) = e^{\left(-d^2/_{2\sigma^2}\right)} \qquad (2)$$

In Eq. (2), $d$ is the distance between the original and aggregated features, $\sigma$ is a parameter that constraints the propagation scope and $Kernel(d)$ is the obtained guidance related to the distance of $d$ based on the kernel. Observe that the position-aware guidance is fading while the distance increases. Particularly, when $d = 0$, the maximum propagated guidance is obtained. Here, a fixed $\sigma$ value is applied for all feature vectors and focused on combining the positional context into attentions.

### *3.3. Position-aware guidance vector*

The guidance in a high-dimensional space for attention is modeled by obtaining the position-aware guidance vector for each feature vector in the video clips. Initially, consider the guidance for a particular distance follows the Gaussian distributions over the hidden dimensions. After, a guidance base matrix $G$ is defined based on the assumption where each column is the guidance base vector related to the particular distance. Each element of $G$ is described as follows:

$$G(i, d) \sim N(Kernel(d), \sigma') \qquad (3)$$

In Eq. (3), $G(i, d)$ is the guidance related to the distance of $d$ in the $i^{th}$ position and $N$ is the normal density with a predicted value of $Kernel(d)$ and standard variance of $\sigma'$. Using the guidance base matrix, the guidance vector for a feature at a particular position is obtained by aggregating the guidance of all features occurring in the video clips:

$$A_j = Gc_j \qquad (4)$$

In Eq. (4), $A_j$ is the aggregated guidance vector for the feature at position $j$ and $c_j$ is the distance count vector which estimates the count of features with different distances. Particularly, for the feature at position $j$, the count of body joint and trajectory point features with a distance of $d$ i.e., $c_j(d)$ is computed as follows:

$$c_j(d) = \sum_{f \in F}[(j - d) \in pos(f)] + [(j + d) \in pos(f)] \qquad (5)$$

In Eq. (5), $F$ is the 3D feature maps containing multiple features, $f$ is either a body joint location or a trajectory point feature in $F$, $pos(f)$ is the group of $f$'s occurrence positions in different clips and $[\cdot]$ is an indicator function which equals to 1 if the criteria satisfy, or else equals to 0.

### 3.4. Positional attention

A positional attention scheme is proposed that incorporates the position-aware guidance of the features into the aggregated feature's attentive representations. In particular, the attentive weight of a feature at position $j$ in the aggregated feature vector is formulated as:

$$\alpha_j = \frac{e^{\left(e\left(h_j,A_j\right)\right)}}{\sum_{k=1}^{l} e^{\left(e\left(h_k,A_k\right)\right)}} \qquad (6)$$

In Eq. (6), $h_j$ is the hidden vector at position $j$ based on BRNN, $A_j$ is the aggregated position-aware guidance vector obtained by Eq. (4), $l$ is the video sequence length and $e(\cdot)$ is the score function which estimates the feature significance based on the hidden vector and the position-aware guidance vector. Then, the score function is defined as:

$$e\left(h_j, A_j\right) = v^T tanh\left(W_H h_j + W_A A_j + b\right) \qquad (7)$$

In Eq. (7), $W_H$ and $W_A$ are matrices, $b$ is the bias vector, $tanh$ is the hyperbolic tangent function, $v$ is the global vector and $v^T$ is its transpose. By using the obtained attentive weights, the resultant aggregated feature vector is represented by the weighted sum of all the hidden vectors:

$$F_a = \sum_{j=1}^{l} \alpha_j h_j \qquad (8)$$

Thus, the aggregated all the pooled feature vectors belonging to one clip is achieved to get the clip descriptors. Then, these clip descriptors obtained from different convolutional layers are fused using the bilinear production for improving its representation ability [6]. By aggregating the clip descriptors, the final video descriptor is generated and the entire network is trained end-to-end with softmax loss supervised by the class label. Once the video descriptor is obtained, these are fed to the SVM for recognizing the human activities in a specific video sequence.

```
Algorithm:
Input: Video sequences from Penn Action Dataset
Output: Extracted body points, trajectory points (Video descriptor)
    Begin
    Split video sequences into clips;
    $for(each\ clip)$
            Initialize CNN parameters for both attention and feature streams;
            Compute all the activations in convolutional layers;
            Aggregate activations of each convolutional layers using PABRNN;
            //PABRNN
            Formulate the position-aware guidance propagation via Gaussian kernel;
            Calculate the guidance base matrix related to a certain distance;
            Aggregate the guidance of all features in convolutional layers;
            Obtain the aggregated guidance vector;
            Determine the score function and the attentive weight of features;
            Find the resultant aggregated feature vector (clip descriptors) belonging to one clip;
            Combine attention and feature streams using bilinear product function;
            Apply fully connected and softmax layer;
            Train the C3D using aggregated guidance feature vector;
            Predict the video descriptors for a video sequence;
            Perform SVM classifier;
            Recognize the individual activities in a particular video sequence;
            End
```

## 4. Experimental Results

In this section, the JTDPABRD method is implemented in MATLAB 2017b as well as its efficiency is evaluated with the JTDBRD, JTDRD and JTDD based on the RA. In this experiment, the Penn Action dataset is taken into consideration which includes 2326 video sequences of 15 activity classes. The videos are captured from various online video repositories. The length of each video is ranging between 50-100 frames. For every frame, 13 body joints are annotated.

To validate the efficiency, 80% of the data is taken from the entire dataset for training and 20% of data is taken for testing. The body joint coordinates, trajectory points and C3D features are used as baselines. As a result, JTDPABRD with these features is evaluated with various pooling i.e., feature aggregation configurations.

The RA is the percentage of True Positive (TP) and True Negative (TN) rates among the overall amount of trails performed.

$$RA = \frac{TP+TN}{TP+FP+FN+TN} \qquad (9)$$

In Eq. (9), FP and FN stand for the false positive and false negative. TP is the amount of correctly recognized legal activities and they are legal. TN is the amount of correctly recognized illegal activities and they are illegal. FP is the amount of wrongly recognized legal activities but they are illegal. FN is the amount of incorrectly recognized illegal activities but they are legal. The results of body joints and trajectory points extraction are portrayed in Figure 4.

**Figure 4(a).** Sample input video sequence.

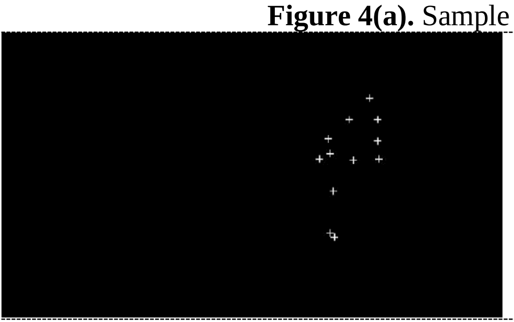

**Figure 4(b).** Results for body joints extraction of input video sequence.

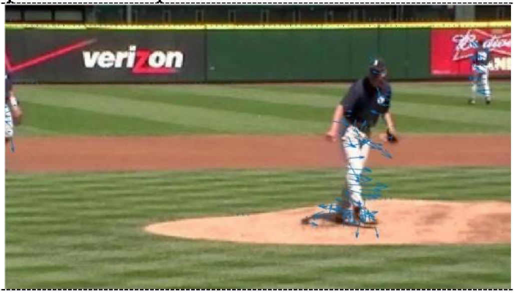

**Figure 4(c).** Results for trajectory points extraction of input video sequence.

The RA results on the Penn Action dataset are provided in Table 1.

**Table 1.** RA of baselines and JTDPABRD with various configurations on Penn action dataset.

| | Concatenate all the activations | JTDPABRD Ratio Scaling (1×1×1) | JTDPABRD Coordinate Mapping (1×1×1) | JTDPABRD Ratio Scaling (3×3×3) | JTDPABRD Coordinate Mapping (3×3×3) |
|---|---|---|---|---|---|
| Joint coordinates+ trajectory coordinates | 0.6452 | - | - | - | - |

| | | | | | |
|---|---|---|---|---|---|
| $fc7$ | 0.7638 | - | - | - | - |
| $fc6$ | 0.7811 | - | - | - | - |
| $conv5b$ | 0.7345 | 0.8358 | 0.8829 | 0.8385 | 0.8683 |
| $conv5a$ | 0.6675 | 0.7768 | 0.8047 | 0.7722 | 0.7831 |
| $conv4b$ | 0.5684 | 0.7965 | 0.7873 | 0.8135 | 0.8258 |
| $conv3b$ | 0.4602 | 0.7268 | 0.7059 | 0.7336 | 0.7315 |

In Table 1, the 1st column is the RAs of directly utilizing body joint coordinates with trajectory point coordinates as C3D features. The other columns are the RAs achieved by the aggregation of all the features in the particular layer. This scrutiny observes that the RA of $fc7$ is marginally lower to that of $fc6$ since the actual C3D on the Penn Action dataset is not able to fine-tune the $fc7$ layer which is highly preferable to construct the video descriptor for the pre-learned dataset. Also, it observes the results of PABRNN-based feature aggregation at various 3D $conv$ layers. To end, it concludes the JTDPABRD has better efficiency as compared to the JTDBRD, JTDRD and JTDD for aggregating the guided feature vectors of body joint and trajectory points in the video sequence.

The results of various combinations of layers using the scores of SVM with late fusion on the Penn Action dataset are given in Table 2.

**Table 2.** RA of fusing JTDPABRD from multiple layers together on Penn action dataset.

| Fusion layers | JTDD | JTDRD | JTDBRD | JTDPABRD |
|---|---|---|---|---|
| | RA | | | |
| $conv5b + fc6$ | 0.867 | 0.871 | 0.875 | 0.883 |
| $conv5b + conv4b$ | 0.987 | 0.989 | 0.991 | 0.994 |
| $conv5b + conv3b$ | 0.873 | 0.875 | 0.879 | 0.883 |

Figure 5 indicates that the fusion of JTDPABRD of various layers particularly improves the feature extraction and recognition results. The mixture of JTDPABRD from $conv5b + conv4b$ can maximize the accuracy of recognizing individual activities efficiently. This is because aggregating more significant features in the $conv$ layers.

The results of the impact of estimated body joints + trajectory points versus Ground-Truth (GT) body joints + trajectory points for different HAR methods on the Penn Action dataset is given in Table 3.

**Table 3.** Impact of estimated body joints + trajectories versus GT body joints + trajectories for different methods on Penn action dataset.

| Methods | GT | Estimated | Difference |
|---|---|---|---|
| JTDD ($conv5b$) | 0.835 | 0.810 | 0.025 |
| JTDRD ($conv5b$) | 0.838 | 0.815 | 0.023 |
| JTDBRD ($conv5b$) | 0.843 | 0.821 | 0.022 |
| JTDPABRD ($conv5b$) | 0.847 | 0.828 | 0.019 |

From Figure 6, it is observed that the JTDPABRD gives more efficiency than compared with the other methods on Penn Action Dataset. The JTDPABRD attains the maximum efficiency not only with GT body joints and trajectory points, however also with the estimated body joints and trajectory points, beyond the other methods.

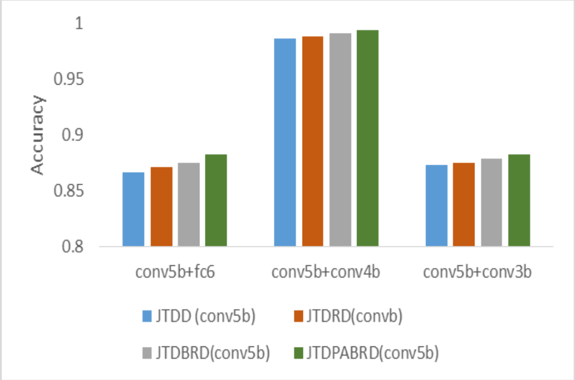

**Figure 5.** RA of fusing JTDPABRD from multiple layers together on Penn action dataset.

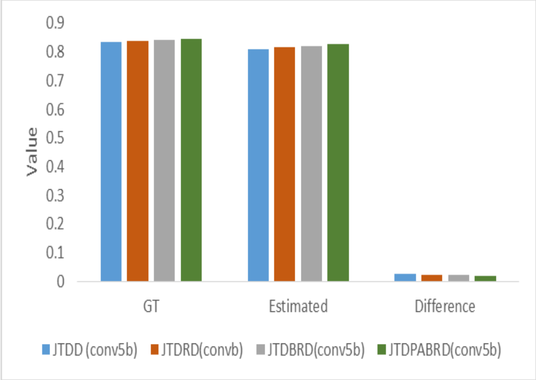

**Figure 6.** Impact of estimated body joints + trajectories versus GT body joints + trajectories for different methods in Penn action dataset.

## 5. Conclusion

In this article, JTDPABRD is suggested to combine the PABRNN and two-stream C3D network for extracting the necessary spatiotemporal features and increasing the accuracy of recognizing individual activities. First, the video is divided into several clips and these clips are fed to the two-stream C3D network as input. In a two-stream C3D network, the attention stream is used to extract the guidance of body joints locations and the feature stream is used to extract the trajectory points along with significant spatiotemporal features. After, every convolutional feature vector representation of each clip belonging to the single video is aggregated via the PABRNN to create the clip descriptor. Also, these two streams are multiplied by the bilinear product and end-to-end trained via class labels. Moreover, the activations of fully connected layers and their spatiotemporal variances are also aggregated to generate the final video descriptor. This video descriptor is fed to the SVM to identify the individual activities in videos. To end, the experimental outcomes proved that the RA of JTDPABRD is improved by fusion of $conv5b$ and $conv4b$ with GT feature vectors as compared to the other methods for HAR systems.

## References

[1] Wan S, Qi L, Xu X, Tong C and Gu Z 2019 Deep learning models for real-time human activity recognition with smartphones *Mob. Netw. Appl.* 1-13

[2] Ding S, Qu S, Xi Y, Sangaiah A K and Wan S 2019 Image caption generation with high-level image features *Pattern Recognit. Lett.* **123** 89-95

[3] Nweke H F, Teh Y W, Al-Garadi M A and Alo U R 2018 Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges *Expert Syst. Appl.* **105** 233-261

[4] Cao C, Zhang Y, Zhang C and Lu H 2017 Body joint guided 3-D deep convolutional descriptors for action recognition *IEEE Trans. Cybern.* **48** 1095-1108

[5] Ji S, Xu W, Yang M and Yu K 2012 3D convolutional neural networks for human action recognition *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 221-231

[6] Srilakshmi N and Radha N 2019 Body joints and trajectory guided 3D deep convolutional descriptors for human activity identification *Int. J. Innov. Technol. Explor. Eng.* **8** 1016-1021

[7] Rahman S, See J and Ho C C 2017 Exploiting textures for better action recognition in low-quality videos *EURASIP J. Image Video Process.* **2017** 74

[8] Li X, Wang D and Zhang Y 2017 Representation for action recognition using trajectory-based low-level local feature and mid-level motion feature *Appl. Comput. Intell. Soft Comput.* **2017** 1-7

[9] Jin C B, Do T D, Liu M and Kim H 2018 Real-time action recognition using multi-level action descriptor and DNN *Intell. Video Surveill.* IntechOpen.

[10] Shou Z, Lin X, Kalantidis Y, Sevilla-Lara L, Rohrbach M, Chang S F and Yan Z 2019 Dmc-net: generating discriminative motion cues for fast compressed video action recognition *Proc. IEEE Conf. Comput. Vis. Pattern Recog.* 1268-1277

[11] Huo Y, Xu X, Lu Y, Niu Y, Lu Z and Wen J R 2019 Mobile video action recognition *arXiv preprint arXiv:1908.10155*.

[12] Nida N, Yousaf M H, Irtaza A and Velastin S A 2019 Instructor activity recognition through deep spatiotemporal features and feedforward extreme learning machines *Math. Probl. Eng.* **2019** 1-13

[13] Sudhakaran S, Escalera S and Lanz O 2019 LSTA: long short-term attention for egocentric action recognition *Proc. IEEE Conf. Comput. Vis. Pattern Recog.* 9954-9963