Hindawi
Journal of Robotics
Volume 2020, Article ID 3096858, 16 pages
https://doi.org/10.1155/2020/3096858

Research Article

Enhanced Human Action Recognition Using Fusion of Skeletal Joint Dynamics and Structural Features

S. N. Muralikrishna,[1] Balachandra Muniyal,[2] U. Dinesh Acharya,[1] and Raghurama Holla[3]

[1]*Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India*
[2]*Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India*
[3]*Department of Computer Applications, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India*

Correspondence should be addressed to Raghurama Holla; raghu247@gmail.com

Received 14 October 2019; Revised 12 June 2020; Accepted 9 July 2020; Published 1 August 2020

Academic Editor: L. Fortuna

In this research work, we propose a method for human action recognition based on the combination of structural and temporal features. The pose sequence in the video is considered to identify the action type. The structural variation features are obtained by detecting the angle made between the joints during the action, where the angle binning is performed using multiple thresholds. The displacement vector of joint locations is used to compute the temporal features. The structural variation features and the temporal variation features are fused using a neural network to perform action classification. We conducted the experiments on different categories of datasets, namely, KTH, UTKinect, and MSR Action3D datasets. The experimental results exhibit the superiority of the proposed method over some of the existing state-of-the-art techniques.

1. Introduction

The rapid growth in hardware and software technologies has resulted in continuous generation of a huge amount of video data through video capturing devices such as smartphones and CCTV camera. Also, a large amount of video content is being uploaded to YouTube every minute. Therefore, it is very important to extract useful information from these huge video databases and to recognize high-level activities for various applications such as automated surveillance systems, human-computer interaction, sports video analysis, real-time patient/children monitoring, shopping-behavior analysis, and dynamical systems [1]. Hence, human action recognition (HAR) from videos is an active area of research as it attracted the attention of several researchers in recent years.

Human action recognition focuses on detecting and tracking people, in particular, understanding human behaviors from a video sequence. The research in this area focuses mainly on the development of techniques for an automated visual surveillance system. It requires a combination of computer vision and pattern recognition algorithms. However, in the literature, *activity*, *behavior*, *action*, *gesture*, and '*primitive/complex event*' are frequently used to describe essentially the same concepts. HAR is challenging because of the intraclass variation and interclass similarity. The same activity may vary from subject to subject, known as the intraclass variation. Without the contextual information, different activities may look similar, which leads to interclass variation, for example, *playing* and *running*. There are many challenges in HAR, such as multisubject interactions, group activities, and complex visual background.

The two main approaches used for HAR are based on global descriptors and local descriptors. The local descriptors are robust to noise and can be applied to a wide range of action recognition problems. However, in recent years, the skeleton-based approaches have been widely used due to the availability of depth sensors. Several datasets are available for

the evaluation of action recognition algorithms. They vary in terms of the number of classes, sensors used, duration of action, view point, complexity of action performed, and so on. In this work, we address the problem of action recognition using skeleton-based approach.

Contributions: (a) We propose a method for human action recognition based on encoded joint angle information and joint displacement vector. (b) A neural network-based method to perform score-level fusion for action classification is proposed. (c) We experimentally show that the proposed method can be applied on datasets containing the skeletal joint information acquired using Kinect sensors and also on datasets where explicit pose estimation needs to be done. Thus, the proposed method can be used with a vision-based sensor or Kinect sensor.

The rest of the paper is organized as follows. Section 2 gives an overview of the existing techniques for human action recognition. Section 3 describes the proposed approach. The experimental results are demonstrated in Section 4. The conclusions and discussions are given in Section 5.

2. Review of Existing Techniques

Human activities can be broadly classified into four categories: gestures, actions, interactions (with objects and others), and group activities. Early approaches developed in 1990s mainly focused on identifying gestures and simple actions based on motion analysis. A detailed review of motion analysis-based techniques is presented by Aggarwal and Cai [2]. However, the motion analysis-based methodologies were found to be less robust as they were insufficient to describe human activities containing complex structures. Therefore, an improved approach was discussed by Aggarwal and Ryoo [3], who focused on methodologies to perform high-level activity recognition designed for the analysis of human actions, interactions, and group activities.

Ben-Arie et al. [4] have proposed a technique to perform human action recognition by computing a set of pose and velocity vectors for body parts such as hands, legs, and torso. These features are stored in a multidimensional hash table to achieve indexing- and sequence-based voting. Kellokumpu et al. [5] proposed another approach based on texture descriptor by combining motion and appearance cues. The movement dynamics are captured using temporal templates, and the observed movements are characterized using texture features. A spatiotemporal space is considered, and the human movements are described with dynamic texture features. Also, the use of motion energy features for human activity analysis is presented by Gao et al. [6]. The motion energy template is constructed for the video using a filter bank, and the actions are classified using SVM. Xu et al. [7] have proposed a hierarchical spatiotemporal model for human activity recognition. The model consists of a two-layer hidden conditional random field (HCRF), where the bottom layer is used to describe the spatial relations in each frame, and the top layer uses high-level features for characterizing the temporal relations throughout the video sequence. The bottom layer also provides high-level semantic representations. A learning algorithm is used, and human activities are identified. To improve the robustness of action recognition task, a combination of features consisting of dense trajectories and motion boundary histogram descriptors has been used by Wang et al. [8]. The descriptor captures different kinds of information such as shape, appearance, and motion to address the problem of camera motion.

The deep learning models gained popularity because of their superior performance in the field of pattern recognition and computer vision research. A review by Guo et al. [9] highlights the important developments in deep neural models. Ji et al. [10] proposed a 3D CNN model for human action recognition. The features are extracted from both spatial and temporal dimensions using 3D convolutions, thus capturing discriminative features. In another work, Wang et al. [11] proposed a technique where the spatiotemporal information obtained from 3D skeleton sequences is encoded into multiple 2D images forming Joint Trajectory Maps (JTMs), and ConvNets are applied to accomplish the action recognition task. As Joint Distance Maps (JDMs) describe texture features which are less sensitive to view variations, Li et al. [12] have developed an approach for action recognition by encoding spatiotemporal information of skeleton sequences into color texture images. Then, using convolutional neural networks, the discriminative features are obtained from the JDMs for achieving both single-view and cross-view action recognition. Hou et al. [13] have proposed a method for effective action recognition based on skeleton optical spectra (SOS), where discriminative features are learned using convolutional neural networks (ConvNets). The spatiotemporal information of a skeleton sequence is effectively captured using skeleton optical spectra. This method is more suitable in case of limited annotated training video data. Wang et al. [14] have presented a detailed survey of recent advances in RGB-D based motion recognition using deep learning techniques. In another approach, Rahmani et al. [15] have developed an improved version of deep learning model based on nonlinear knowledge transfer model learning, achieving invariance to viewpoint change. A general codebook is generated using k-means to encode the action trajectories, and then the same codebook is used for encoding action trajectories of real videos. Li et al. [16] have used multiple deep neural networks to achieve multiview learning for three-dimensional human action recognition. These multiple networks help to effectively learn the discriminative features and also capture spatial and temporal information. The recognition scores of all views are combined using *multiply fusion*. Xiao et al. [17] have introduced an end-to-end trainable architecture-based model for human action recognition. The model consists of deep neural networks and attention models for learning spatiotemporal features from the skeleton data. Li et al. [18] have proposed an approach for skeleton-based human action recognition. A deep model, namely, 3DConvLSTM, is used to learn spatiotemporal features from the video sequences, and an attention-based dynamic map is built for action classification.

An approach for online action recognition has been proposed by Tang et al. [19] based on weighted covariance descriptor by considering the importance of frame sequences with respect to their temporal order and discriminativeness. The combination of nearest neighbour search and Log-Euclidean kernel–based SVM is used for classification. In another work, an optical acceleration-based descriptor has been used by Edison and Jiji [20] for human action recognition. Two descriptors have been computed for effectively capturing the motion information, namely, the histogram of optical acceleration and histogram of spatial gradient acceleration. An approach based on rank pooling method was introduced by Fernando et al. [21] for action recognition, which is capable of capturing both the appearance and the temporal dynamics of the video. A ranking function generated by the ranking machine provides important information about actions. In another work, Wang et al. [22] have presented a technique for action recognition based on order-aware convolutional pooling, focusing mainly on effectively capturing the dynamic information present in the video. After extracting features from each video frame, a convolutional filter bank is applied to each feature dimension, and then filter responses are aggregated. Hu et al. [23] introduced a new approach for early action prediction based on soft regression applied on RGB-D channels. Here, the depth information is considered to achieve more robustness and discriminative power. Finally, Multiple Soft labels Recurrent Neural Network (MSRNN) model is constructed, where feature extraction is done based on Local Accumulative Frame Feature (LAFF). Some more approaches for action recognition can be found, which have been developed based on sparse coding, Yang and Tian [24]; exemplar modeling, Hu et al. [25]; max-margin learning, Zhu et al. [26]; Fisher vector, Wang and Schmid [27]; and block-level dense connections, Hao and Zhang [28]. Through literature survey, it is found that several techniques have been proposed for human action recognition. A detailed review on action recognition research is reported by Ramanathan et al. [29], Gowsikhaa et al. [30], and Fu [31].

A lot of approaches are available in the literature for human action recognition. Most of the existing techniques use either the local features extracted temporally or the skeleton representation of the human pose in the temporal sequence. However, the combination of temporal features and spatial features provides better recognition rate. In this direction, we propose a method to recognize human action based on the combination of appearance and temporal features at the classifier decision level.

3. Proposed Work

In this work, we propose a method for human action recognition by considering the structural variation feature and the temporal displacement feature. The proposed method extracts features from the pose sequence in a given video. Figure 1 depicts the methodology of the proposed system. We extract the structural variation feature by detecting the angle made between the joints during an action. There are several methods available to estimate the pose. Some of the pose estimation techniques found in the literature are based on sensor readings, and other methods are based on vision-based techniques.

3.1. Pose Estimation for Action Recognition. The OpenPose library [32, 33] is one of the well-known vision-based libraries used to extract the skeletal joints. The performance of the OpenPose library to detect the joint locations is limited when compared to sensor based methods. It uses VGG-19 deep neural network model to estimate the pose. The COCO model [34] consists of 18 skeletal joints, whereas the BODY_25 model gives 25 skeletal joint locations. In our experiments, we have used OpenPose to estimate the pose for the KTH dataset; however, for the other datasets, the pose information is taken from sensor readings. In the following section, we present the idea of structural feature extraction.

3.2. Structural Variation Feature Extraction. Let us consider the skeleton represented by a set of points $S = \{j_i \mid i = 1, 2, \ldots n_s\}$ having n_s joints, where $j_i = (x_i, y_i)$ indicating the estimated joint location in the 2D image location. Our goal is to obtain the angle between the joint j_k and a set of joints $J = \{j_i \mid i = 1, 2, \ldots, n_s \text{ and } i \neq k\}$ which contributes to structural variation in the skeleton. In a video having N frames, the angle θ_{ij}^k is found, where θ_{ij}^k represents the angle between the joints j_i and j_j in the k^{th} frame, where $k = 1, 2, 3, \ldots N$. For each joint j_i, a binary vector $\overrightarrow{v_i}$ is computed using

$$\overrightarrow{v_i} = \left[v_1, v_2, \ldots, v_p, \ldots, v_{n_s} \right], \tag{1}$$

where v_p is given by

$$v_p = \frac{1}{N} \sum_{k=1}^{N} \left| \theta_{ip}^k \right| \geq T. \tag{2}$$

The procedure followed to fix the threshold "T" is given in Section 3.2.1. The feature vectors $\overrightarrow{v_i}$, for $i = 1, 2, \ldots, n_s$, are concatenated to obtain the structural feature vector $\overrightarrow{v}$. The dimension of $\overrightarrow{v}$ is $(n_s - 1) \times (n_s - 1)$.

3.2.1. Feature Extraction Based on Angle Binning. The vector $\overrightarrow{v}$ does not provide the variation in the angle at a finer level, as it is binarized with a single threshold value. Accordingly, we perform angle binning, where multiple thresholds are used in (3) to quantize the angle to a b-bit number, by modifying (2). This captures the angle between the joints at a finer level; at the same time, the quantization helps to suppress minute variations in the angle during an action. The process of feature extraction is shown in Figure 2.

$$v_p = \sum_{l=0}^{b-1} f\left(\widehat{\theta}_{ij}^k, T_l \right) * 2^l, \tag{3}$$

where $T_l = l * (\theta_{\max}/b), 0 \leq \theta_{\max} \leq (\pi/2)$. The terms $\widehat{\theta}_{ij}^k$ and $f(x, y)$ are defined using

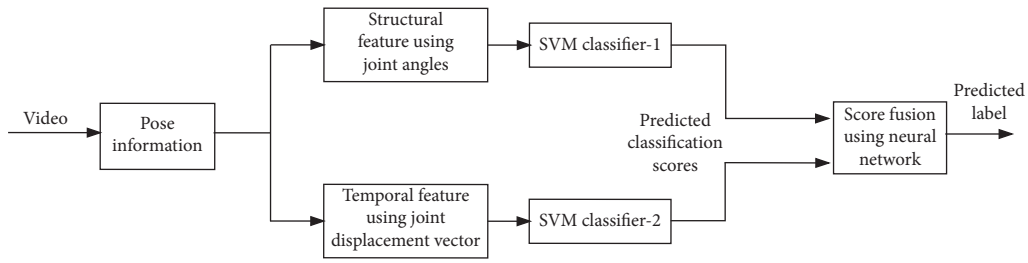

Figure 1: Overall methodology of action classification.

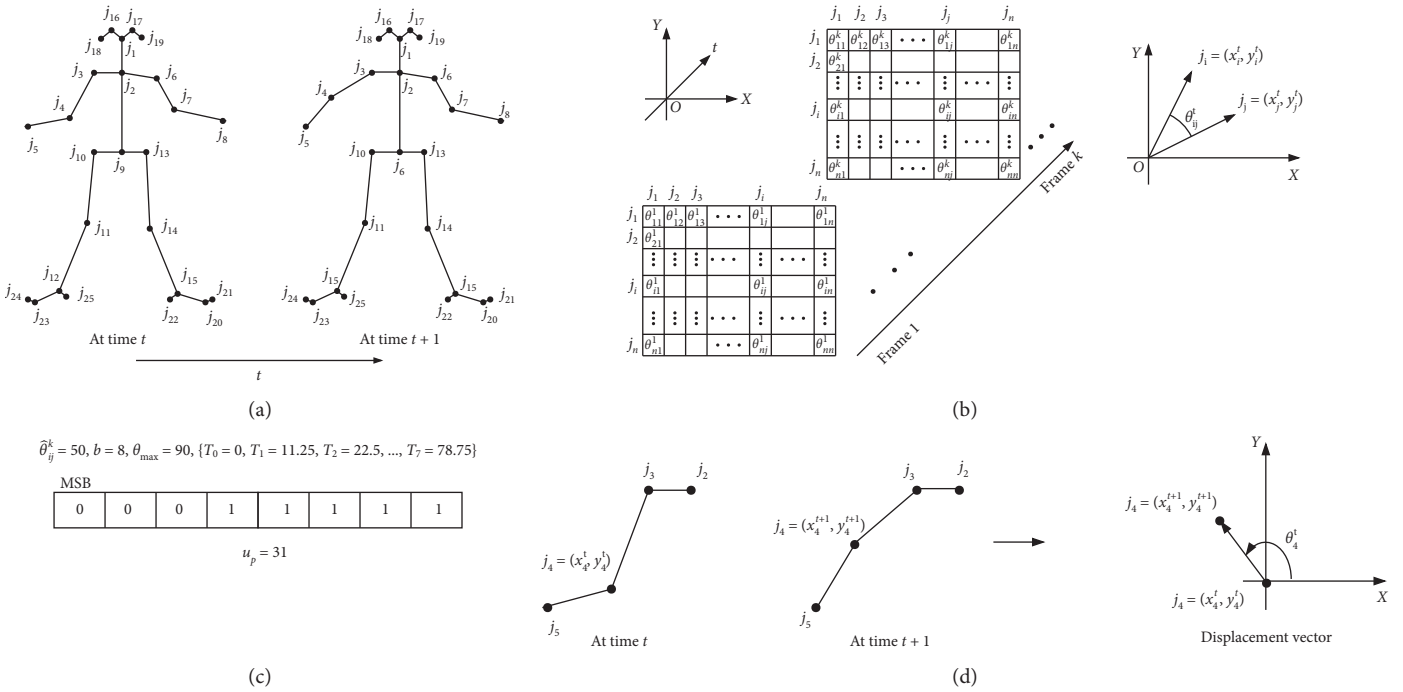

Figure 2: Structural feature extraction and temporal feature extraction from joint locations. (a) An example of a skeletal joint model. (b) Computing the angle information for structural feature extraction using (3). (c) An example of angle binning with $b = 8$ and $\theta_{\max} = (\pi/2)$. (d) An example of computing the displacement angle for temporal feature extraction.

$$\hat{\theta}_{ij}^k = \frac{1}{N}\sum_{k=1}^{N}\left|\theta_{ij}^k\right|, \tag{4}$$

$$f(x, y) = \begin{cases} 1, & \text{if } x > y, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The temporal variation feature captures the dynamics of individual joint by tracking them through the frames. This process is explained in the following section.

3.3. Temporal Feature Extraction. The temporal feature extraction looks at the change in the joint location for a joint j_i from the frame t to $t + 1$. We consider the location of the joint j_i in two successive frames to find the relative position of the joint. This is effectively the tracking of the joint location. A histogram of 2D displacement orientation of joint location in the X–Y plane is constructed to capture the temporal dynamics. The vector representing the joint displacement is computed for the sequence of video frames. For each displacement vector of joints, we obtain the orientation pair consisting of orientation angle and the magnitude represented by (θ_i, ρ_i). The orientation angles θ_i, $i = 1, 2, \ldots, n_s$, for all the joints are used as the temporal features. The θ_i^t for joint j_i at time instance $t + 1$ is computed using

$$\theta_i^t = \arctan\left(\frac{\Delta y}{\Delta x}\right). \tag{6}$$

The displacement vector v_t of joint j_i at time instance $t + 1$ is calculated using

$$v_t = \begin{bmatrix} \Delta x_i \\ \Delta y_i \end{bmatrix} = \begin{bmatrix} x_i^{t+1} - x_i^t \\ y_i^{t+1} - y_i^t \end{bmatrix}. \tag{7}$$

The feature vector $\overrightarrow{f_i}$ for every joint location j_i is given by

$$\overrightarrow{f_i} = \left[\theta_i^1, \theta_i^2, \ldots, \theta_i^t, \ldots, \theta_i^N \right]. \quad (8)$$

A k-bin histogram is created for every joint j_i from the feature vector $\overrightarrow{f_i}$. This is concatenated to form a temporal feature vector $\overrightarrow{f}$ representing an action. It is clear that the joint locations are sparse when compared to the traditional optical flow-based methods. Thus, the feature extraction process is computationally more efficient.

3.4. Score-Level Fusion Using Neural Network. We combine the structural features and the temporal features at the score level. For every sample j, the classifier i assigns a score ranging from −inf to +inf. The score is the signed distance of the observation j to the decision boundary. A positive score indicates that the sample j belongs to class i. A negative score gives the distance of j from decision boundary. The score-level fusion is performed using a neural network. The neural network assigns significance scores to the classifiers based on structural and temporal features. The structural features are less discriminative for describing actions having similar body part movements such as *walk*, *run*, and *jogging*. The optimal fusion of temporal and structural features would help in better recognition.

To generalize the classifier fusion, we consider a multiclass classification problem with c classes and n classifiers. In our case, we have used scores from two SVM classifiers for fusion. The class prediction score for a sample j from i^{th} classifier is

$$\overrightarrow{x_{ij}} = \left[x_{ij}^{(1)}, x_{ij}^{(2)}, \ldots, x_{ij}^{(c)} \right], \quad (9)$$

where each $x_{ij}^{(t)}$ is a prediction score corresponding to the class t. The input to the neural network for the sample j is given by

$$\overrightarrow{v_j} = \left[\overrightarrow{x_{1j}, x_{2j}}, \ldots, \overrightarrow{x_{ncj}} \right]^T, \quad (10)$$

where $\overrightarrow{v_j} \in \mathbb{R}^{nc}$.

The predicted label at the output layer of the neural network is given by $\overrightarrow{y_j'} \in \mathbb{R}^c$. To get the optimal fusion score, we need to solve the objective function given in (11) for the N training samples in the action recognition dataset.

$$\text{minimize} \sum_{j=1}^{N} \left\| \overrightarrow{y_j} - \overrightarrow{y_j'} \right\|, \quad (11)$$

where $\overrightarrow{y_j}$ is the actual label at the output layer for the sample j.

For a neuron k, in the hidden layer t, the output θ_k of the neuron is given by

$$\theta_k = \sigma\left(\overrightarrow{w_k}^{(t-1)} \overrightarrow{v_j} \right), \quad (12)$$

where

$$\overrightarrow{w_k}^{(t-1)} = \left[w_{1k}^{(t-1)}, w_{2k}^{(t-1)}, \ldots, w_{nck}^{(t-1)} \right]. \quad (13)$$

represents the synaptic weights from the previous layer to the neuron k, and $\sigma(.)$ is the sigmoid activation function. For a neuron p at the output layer, the predicted label o_p is given by

$$o_{pj} = S\left(\overrightarrow{w_p}^l \overrightarrow{v_j}^l \right), \quad (14)$$

where

$$\overrightarrow{w_p}^l = \left[w_{1p}^{(l)}, w_{2p}^{(l)}, \ldots, w_{ncp}^{(l)} \right], \quad (15)$$

represents the synaptic weights from the last hidden layer l to the output neuron p. $\overrightarrow{v_j}^l$ is the input from last hidden layer. $S(.)$ is the softmax function. The output of this layer, $\overrightarrow{y_j'}$, for a sample j, is given by

$$\overrightarrow{y_j'} = \left[o_{1j}, o_{2j}, \ldots, o_{pj}, \ldots, o_{cj} \right]. \quad (16)$$

The neural network uses the backpropagation algorithm to learn the network parameters. An example of neural network architecture used in the proposed model is shown in Figure 3.

4. Experiments and Results

To demonstrate the performance of the proposed model, we carried out experiments on three publicly available datasets, namely, KTH [35], UTKinect [36], and MSR Action3D dataset [37]. The KTH dataset requires explicit pose estimation. However, UTKinect dataset contains the pose information captured using Kinect sensors. The source code of our implementation is available at https://github.com/muralikrishnasn/HARJointDynamics.git.

4.1. Datasets. The KTH dataset contains six action types performed by 25 subjects under four different conditions. The skeletal joint information is not included in the dataset unlike other datasets used in the experiment. The UTKinect dataset is acquired using a Kinect sensor. The dataset contains skeletal joint information for 10 types of actions performed by 10 subjects repeated twice per action. The MSR Action3D dataset contains skeleton data for 20 action types, performed by 10 subjects, where each action is performed 2 to 3 times. The dataset contains 20 joint locations per frame captured using a sensor similar to Kinect device.

4.2. Experimental Setup and Results. In our experiments, OpenPose library [33] is used to estimate the pose for the KTH dataset. A pretrained network with BODY_25 model is used in our experiments. The parameters of the experiment have been set as described in [35]. The deep neural network to detect the joints is executed on a Tesla P100 GPU. The Support Vector Machine (SVM) classifiers are used to extract the structural and temporal features. The predicted scores from these SVM classifiers are combined using a neural network. We used radial basis function kernel in the SVM classifiers. A simple feed-forward network with sigmoid function at the hidden layers and softmax output neurons is used to solve (11). In the experiments, the neural network has been trained with 50 epochs. A plot of epochs

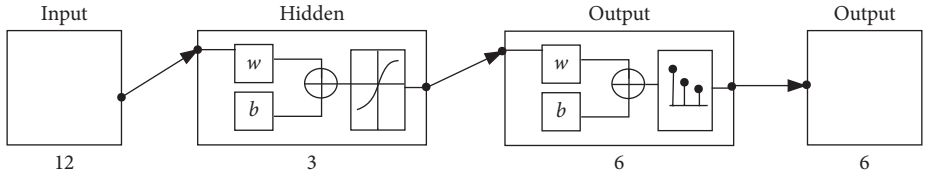

Figure 3: An example of neural network architecture for the KTH dataset with 12 input neurons, 3 hidden layers, and 6 output neurons.

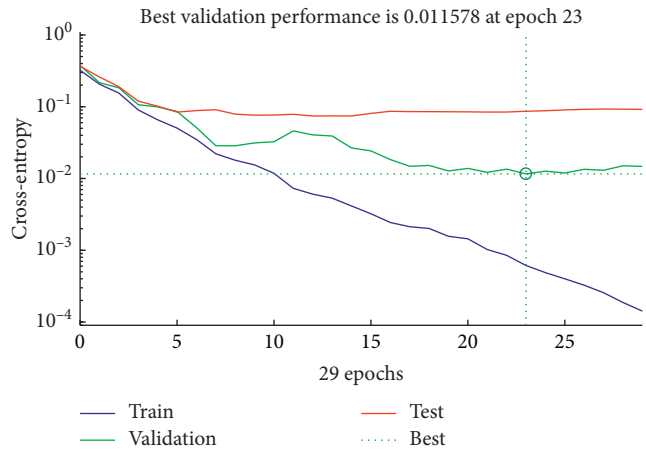

Figure 4: A plot of epochs vs cross-entropy for the neural network.

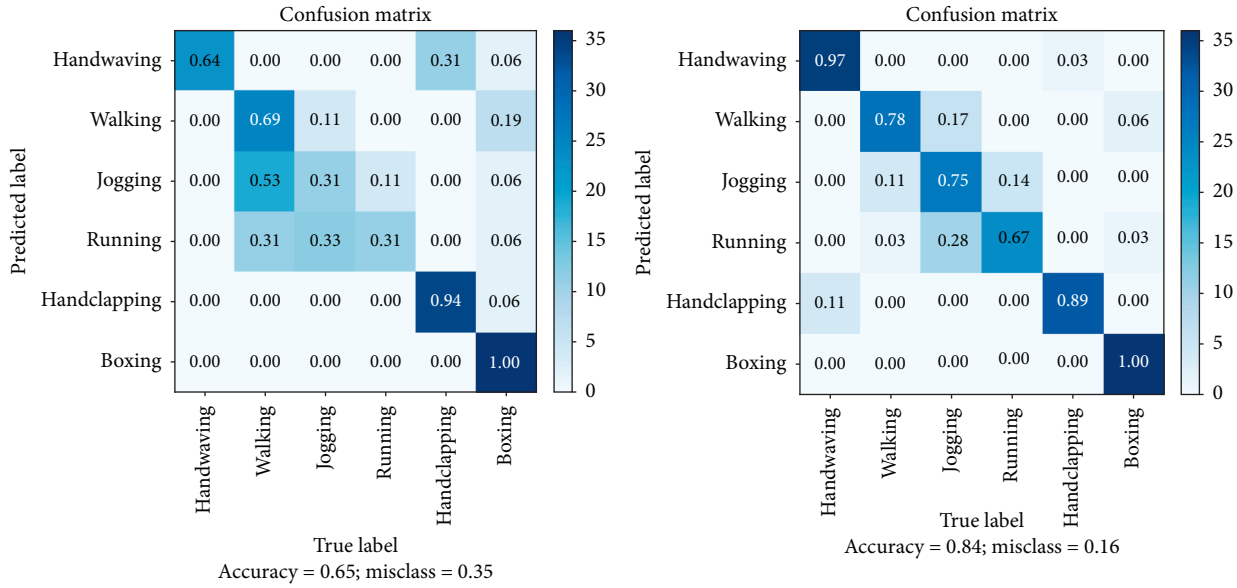

Figure 5: Continued.

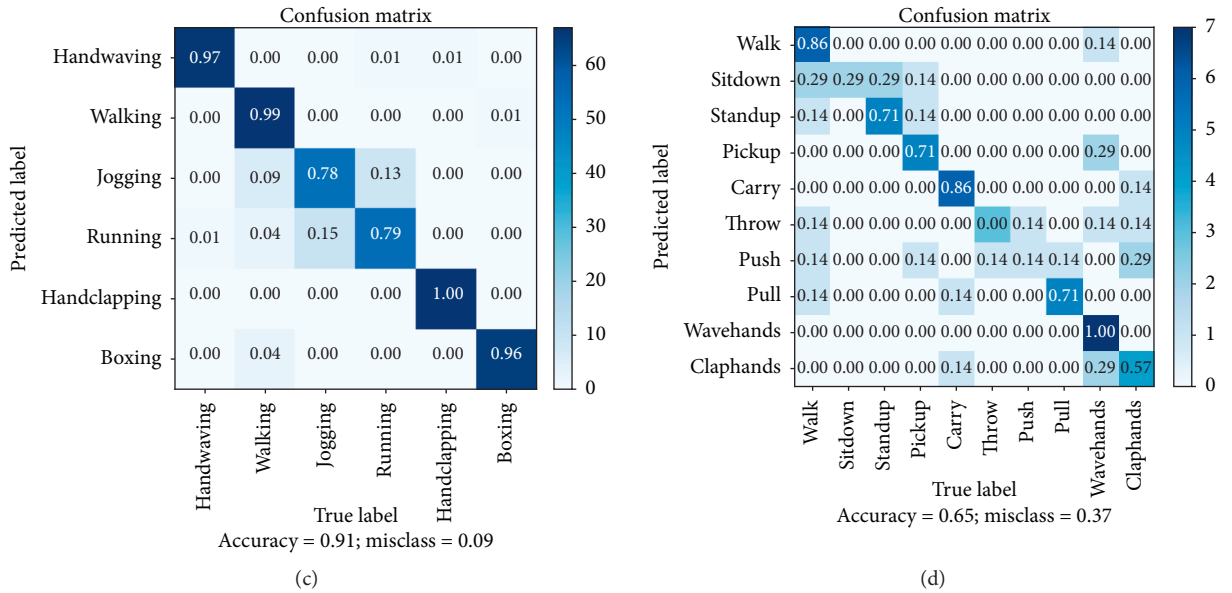

FIGURE 5: (a) Confusion matrix for structural features. Dataset: KTH, #bits used for encoding $b = 8$, total number of joints considered: 25, 2D location of joints. (b) Confusion matrix for temporal features. Dataset: KTH, #bins $k = 5$, features considered: orientation angle only. (c) Confusion matrix for the score-level fusion using neural network. Dataset: KTH, number of hidden layers = 3, epochs = 20. (d) Confusion matrix for structural features. Dataset: UTKinect, #bits used for encoding $b = 8$.

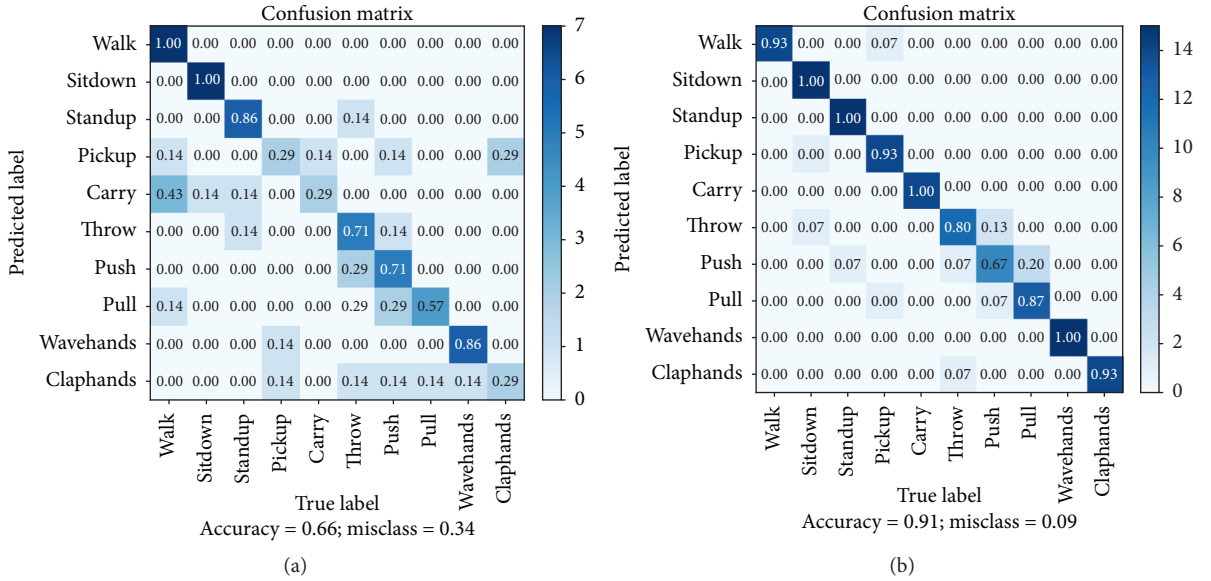

FIGURE 6: (a) Confusion matrix for temporal features. Dataset: UTKinect, #bins $k = 5$, features considered: orientation angle only. (b) Confusion matrix for the score-level fusion using neural network. Dataset: UTKinect, number of hidden layers = 3, epochs = 50.

versus cross-entropy is shown in Figure 4. The results of the experiments are shown in Figures 5(a)–5(c), summarizing the confusion matrices for structural features, temporal features, and the score-level fusion, respectively. It can be seen that the misclassifications are between highly similar actions like running and jogging. The proposed model has achieved an accuracy of $\approx 90.3\%$ on the KTH dataset.

We have conducted experiments on UTKinect dataset in a similar manner to that shown in [36, 44]. The confusion matrix considering the structural features is presented in Figure 5(d). The results for temporal features and score-level fusion using neural network are shown in Figures 6(a) and 6(b). The accuracy of the proposed

TABLE 1: Experimental results for the MSR Action3D dataset.

Dataset	Cross-subject test
AS1	89.6
AS2	83.2
AS3	98.2
Overall	**90.33**

Table 2: Experimental results of the proposed method vs other techniques for human action recognition.

KTH dataset [35]	
STIP, Schüldt [38]	73.6%
Efficient motion features [39]	87.3%
Proposed method	**90.30%**
UTKinect dataset [36]	
Histogram of 3D joints [40]	90.92%
Random forest [41]	87.90%
Proposed method	**91.30%**
MSR Action3D dataset	
Histogram of 3D joints [40]	78.97%
Eigen joints [42]	82.30%
Joint angle similarities [43]	83.53%
Proposed method	**90.33%**

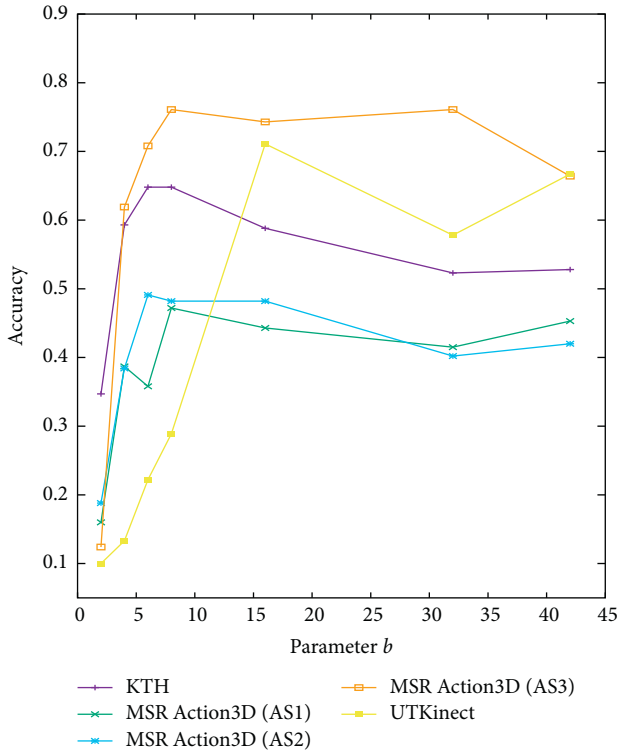

Figure 7: Accuracy of action recognition using only joint angle feature with respect to the quantization parameter b.

method on the UTKinect dataset is ≈ 91.3 with a deviation of ± 1.5.

The experiment on MSR Action3D dataset has been conducted using *cross-subject test* as described in [37] unlike the leave-one-subject-out cross-validation (LOOCV) method given in [40]. The actions are grouped into three subsets: AS1, AS2, and AS3. The AS1 and AS2 have less interclass variations, whereas AS3 contains complex actions. The obtained results are listed in Table 1. A summary of the results from all the three datasets is reported in Table 2. From Table 2, it is observed that the proposed method outperforms the existing methods for

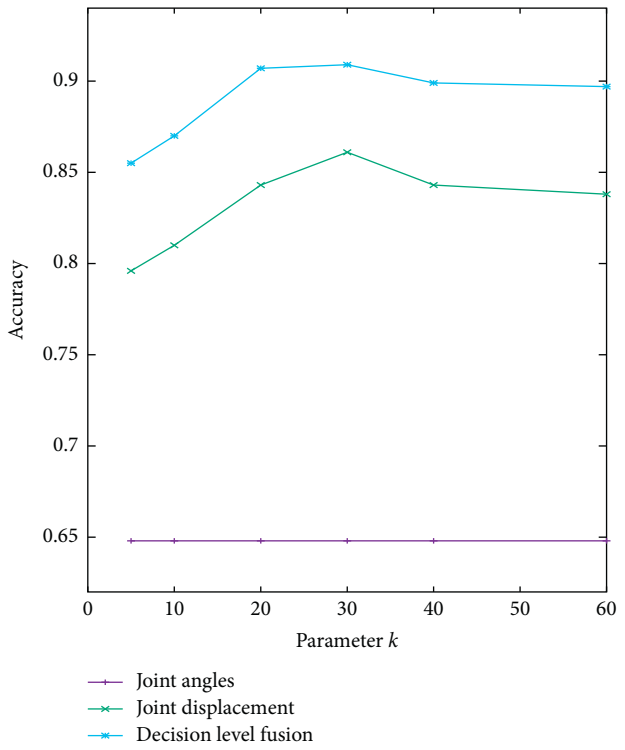

Figure 8: Accuracy of action recognition by varying number of bins k in joint displacement feature for KTH dataset.

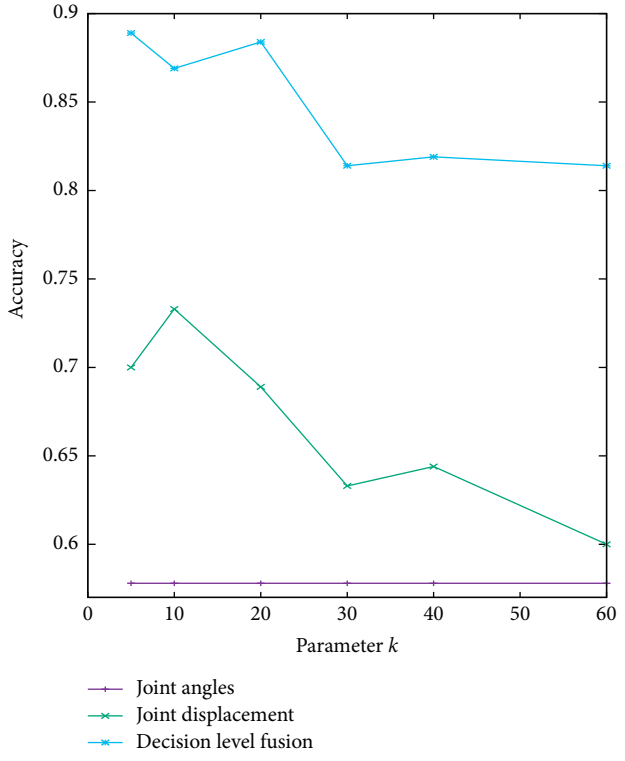

Figure 9: Accuracy of action recognition by varying number of bins k in joint displacement feature for UTKinect dataset.

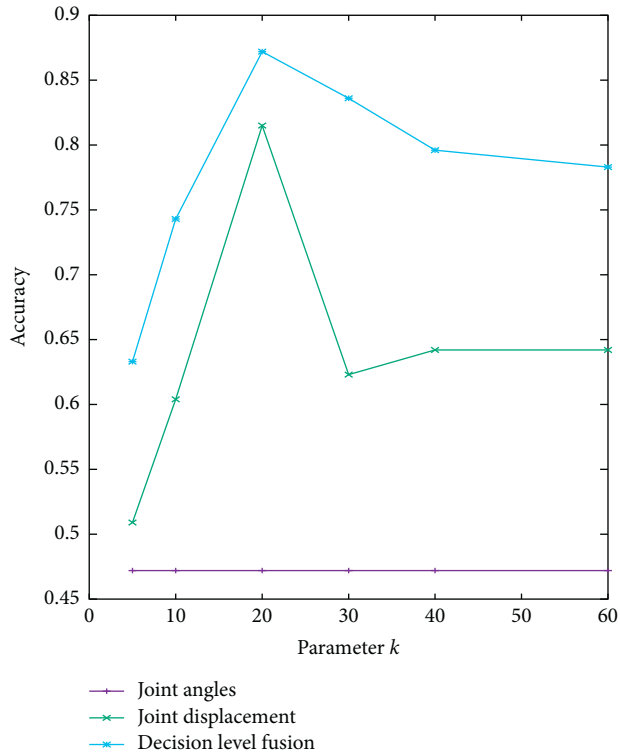

Figure 10: Accuracy of action recognition by varying number of bins k in joint displacement feature for MSR Action3D (AS1) dataset.

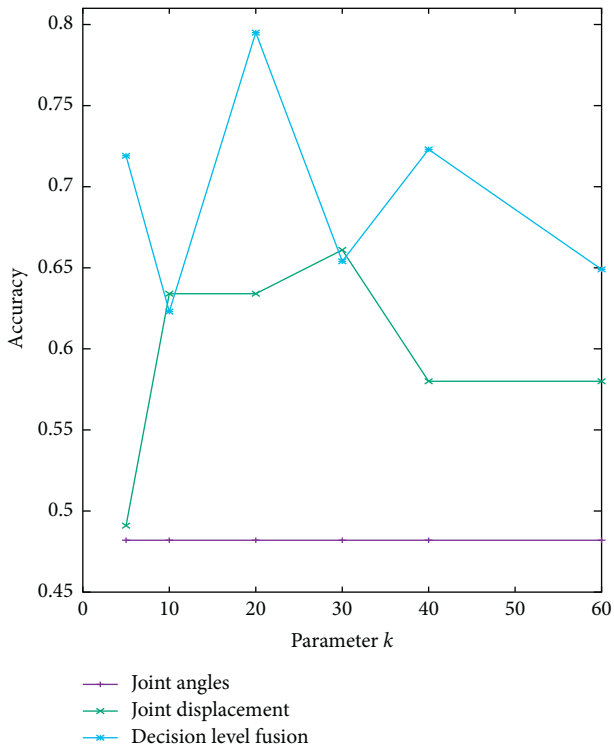

Figure 11: Accuracy of action recognition by varying number of bins k in joint displacement feature for MSR Action3D (AS2) dataset.

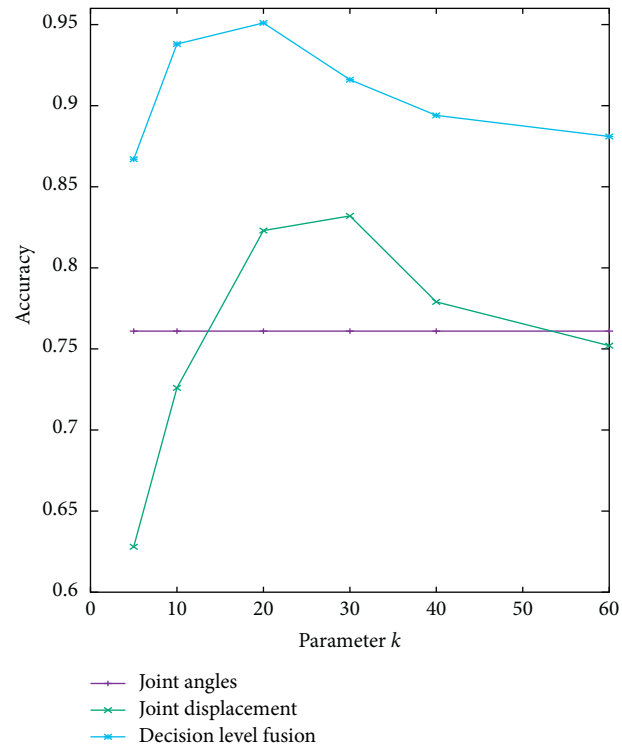

Figure 12: Accuracy of action recognition by varying number of bins k in joint displacement feature for MSR Action3D (AS3) dataset.

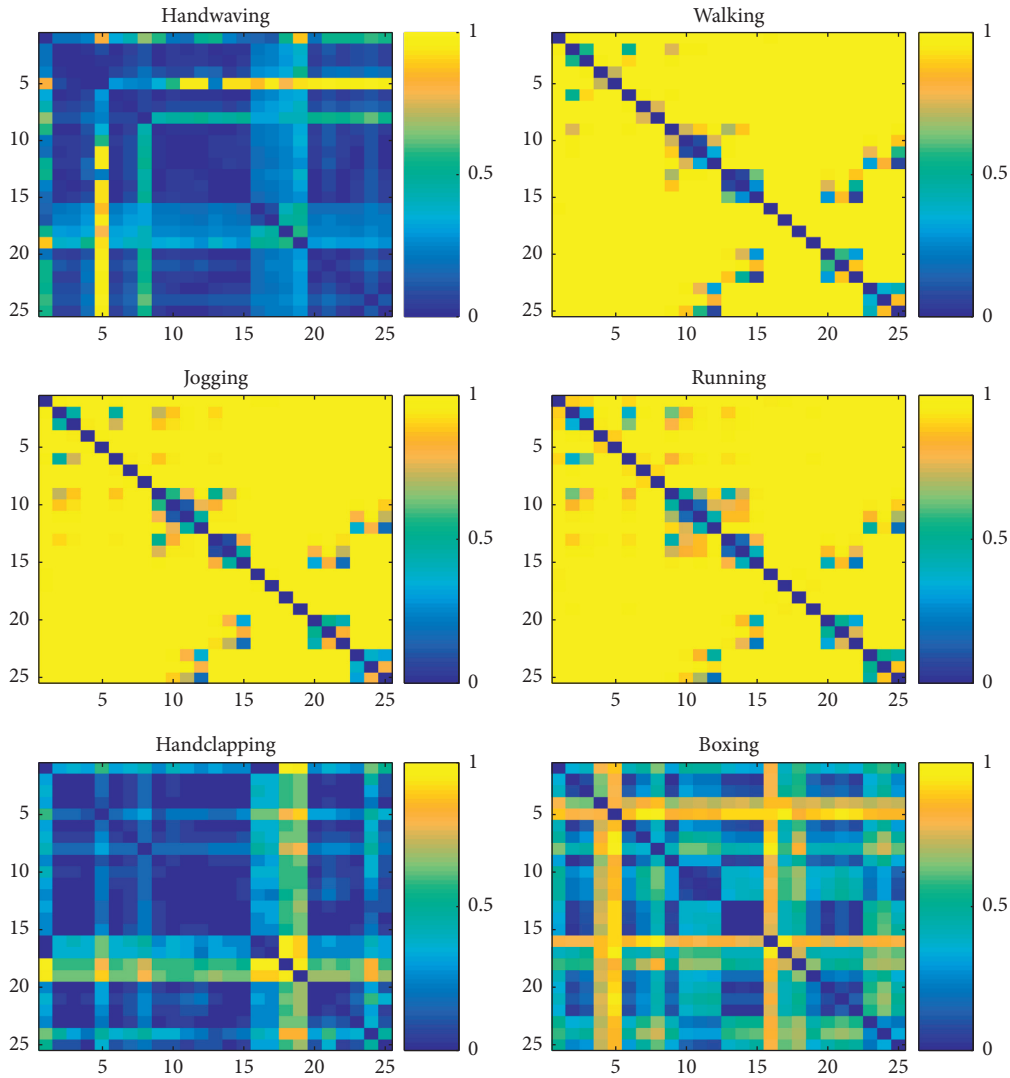

Figure 13: Visualization of dominant joints with respect to angular movement for KTH dataset.

human action recognition. We used similar classifier settings for the other datasets in the experiment. The experimental results show that the proposed method outperforms some of the state-of-the-art techniques for all the three datasets considered in the experiments. For the MSR Action3D dataset, our method gives an accuracy of $\approx 90.33\%$ with a deviation of ± 2.5, which is better than the listed methods in Table 2 by more than $\approx 5\%$. However, the fusion of classifiers shows better performance than the single classifier.

4.3. Influence of Quantization Parameter b and Histogram Bins k on Accuracy. The performance of SVM classifier-1 shown in Figure 1 is analyzed by varying the quantization parameter b. The number of bits b used in quantization versus accuracy is plotted in Figure 7. It is observed that the parameter has no influence on the results beyond $b = 8$ for KTH and MSR Action3D datasets. However, the optimal value of b for UTKinect dataset is 16. This is due to the variations in the range of data values for the location coordinates.

A plot of number of bins k in joint displacement feature versus the accuracy is shown in Figures 8–12. The displacement vectors provide complementary information to joint angles. Most of the pose estimation algorithms fail to detect the joints that are hidden due to occlusion or self-occlusion. Normally, the pose estimation algorithms result in a zero value for such joint locations. These hidden joint locations act as noise and may degrade the performance of the action recognition algorithm.

4.4. Analysis of Most Significant Joints. In KTH dataset, the *hand-waving* action is mainly due the movement of joints $j3$ to $j8$. The other joints do not contribute to the action. The most important joints involved in an action are depicted in Figure 13. It can be observed that actions *walking, running, and jogging* have similar characteristics in terms of angular movements. This is very useful in identifying any outliers while detecting abnormalities in actions. (Dominant joints with respect to angular movement for other datasets are included in Figures 14–17).

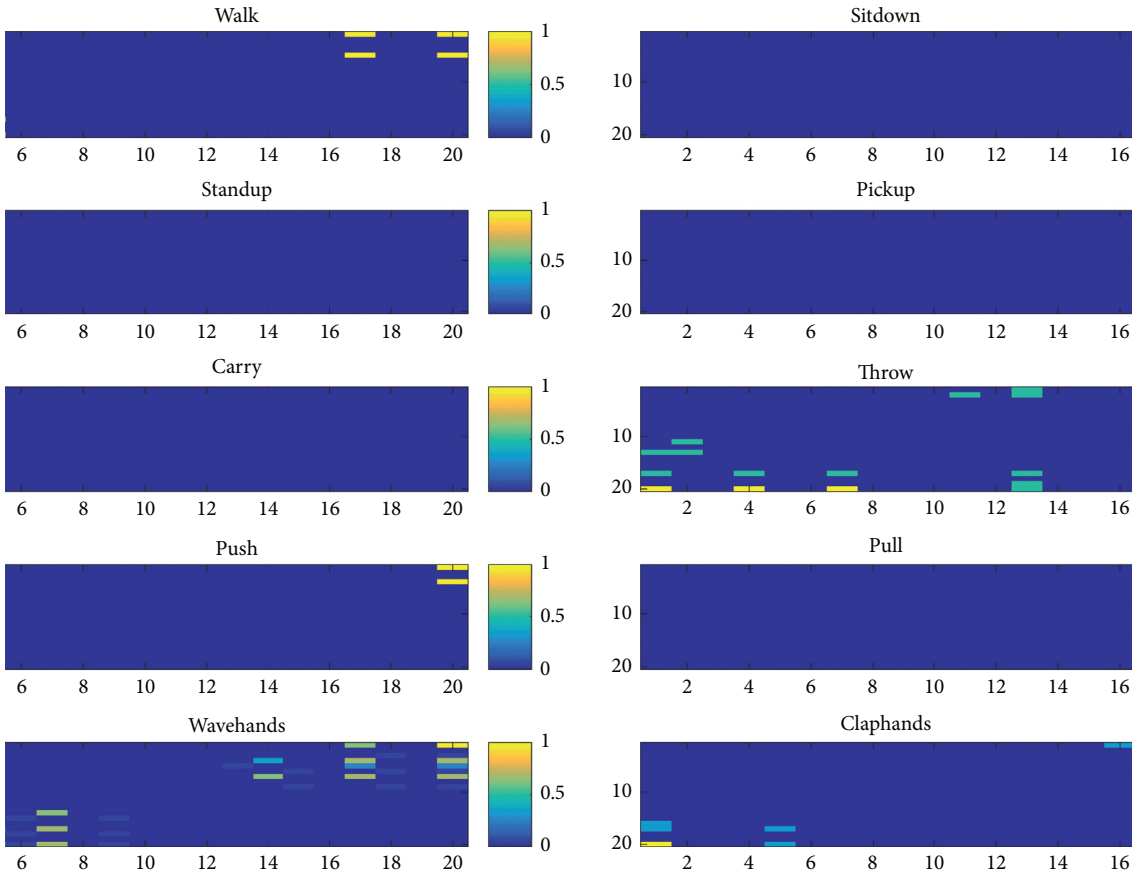

Figure 14: Visualization of dominant joints with respect to angular movement for UTKinect dataset.

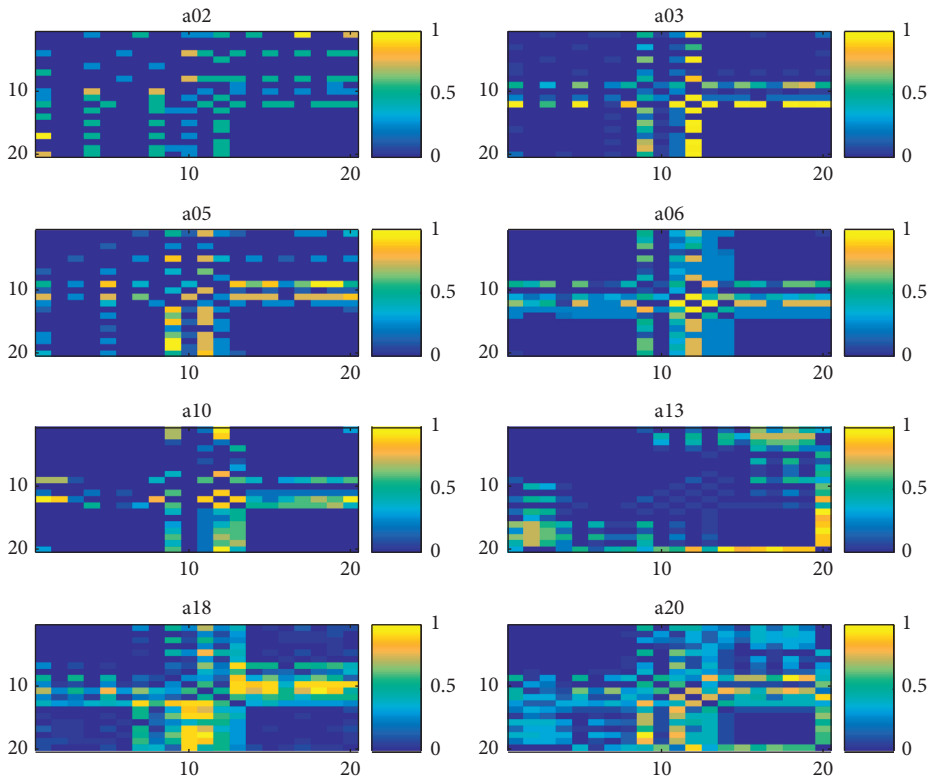

Figure 15: Visualization of dominant joints with respect to angular movement for MSR Action3D (AS1) dataset.

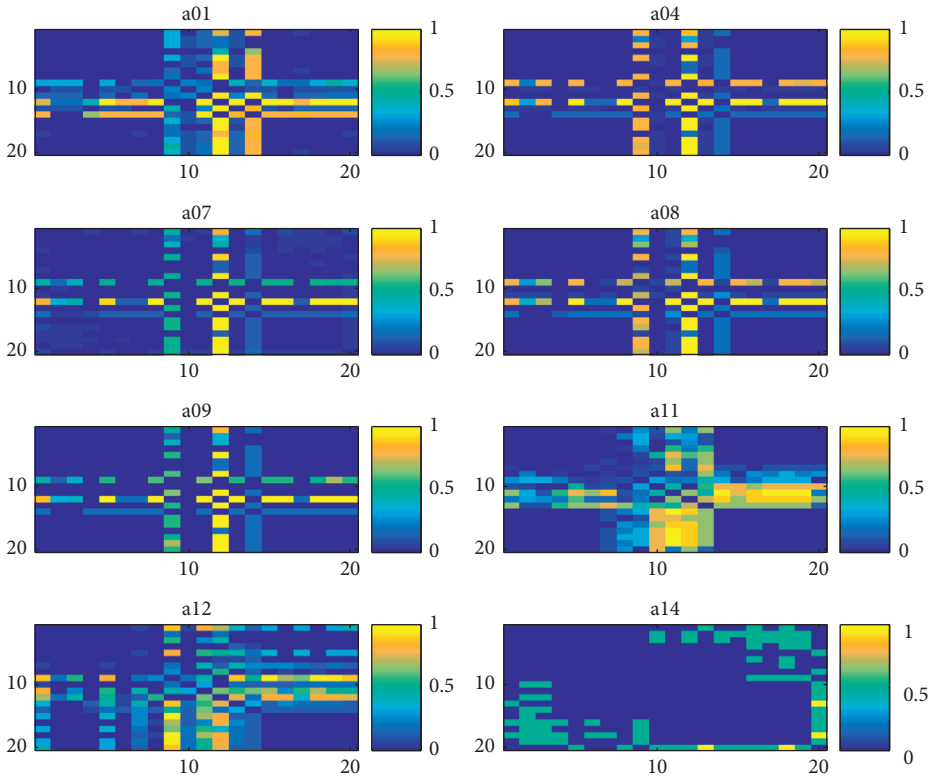

Figure 16: Visualization of dominant joints with respect to angular movement for MSR Action3D (AS2) dataset.

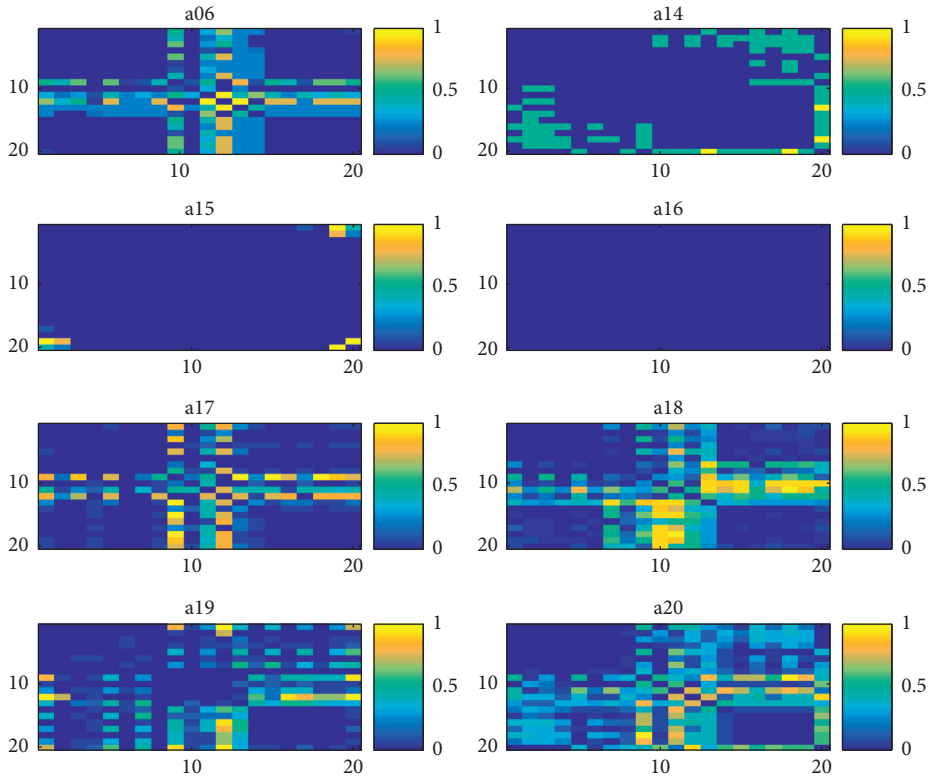

Figure 17: Visualization of dominant joints with respect to angular movement for MSR Action3D (AS3) dataset.

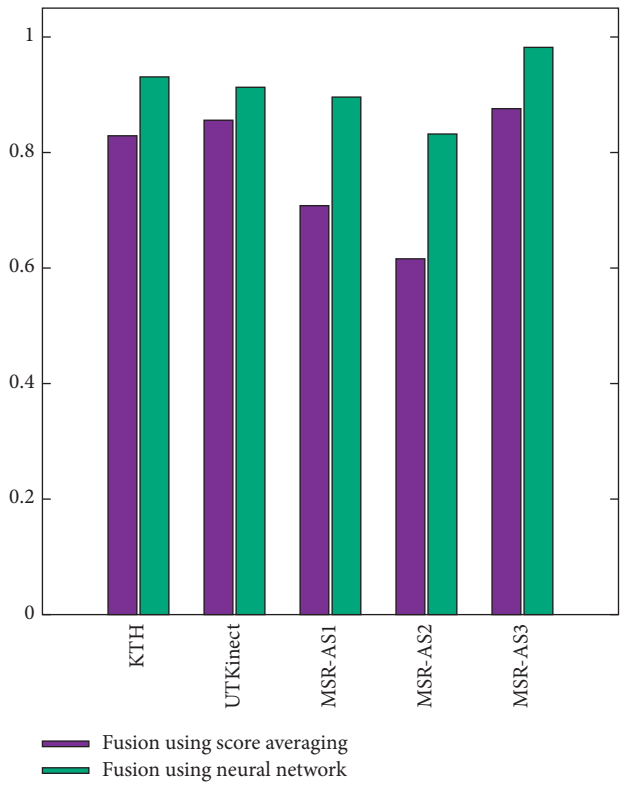

Figure 18: A comparison of decision level fusion using neural network and score averaging [45].

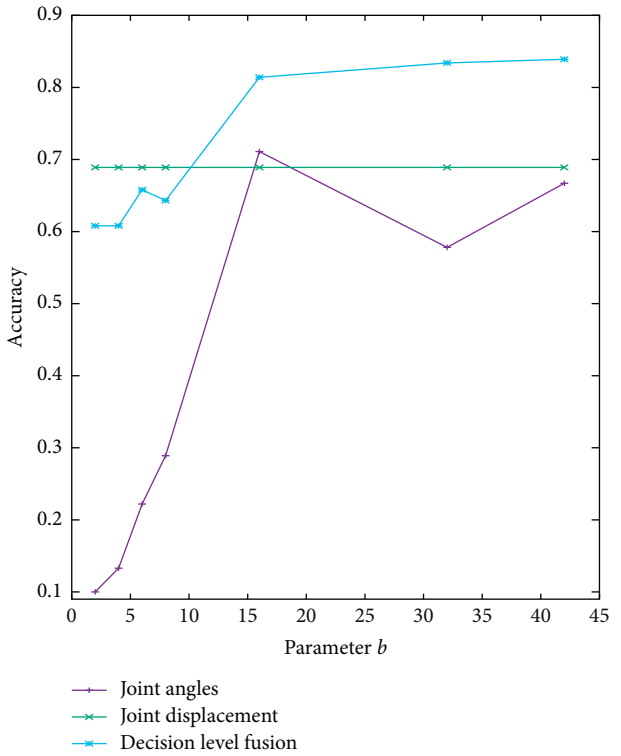

Figure 20: A graph demonstrating improvement in accuracy using decision level fusion for the UTKinect dataset.

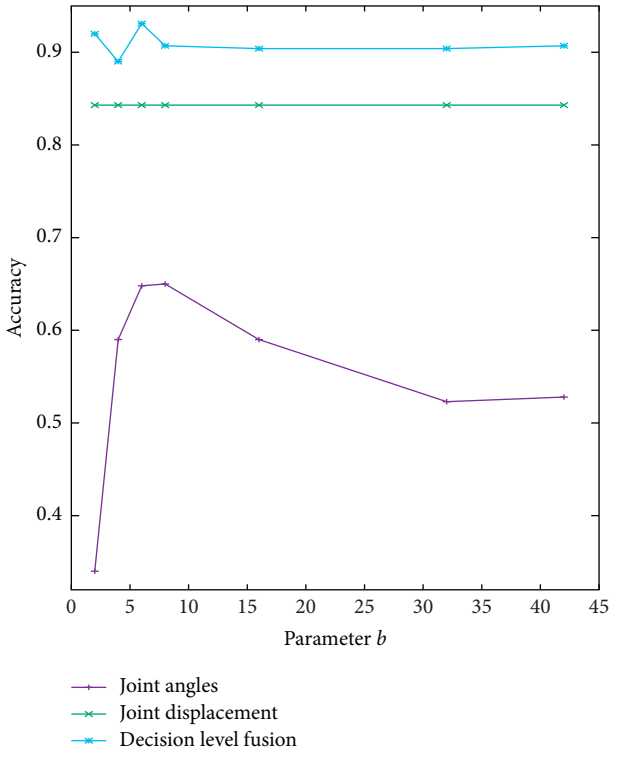

Figure 19: A graph demonstrating improvement in accuracy using decision level fusion for the KTH dataset.

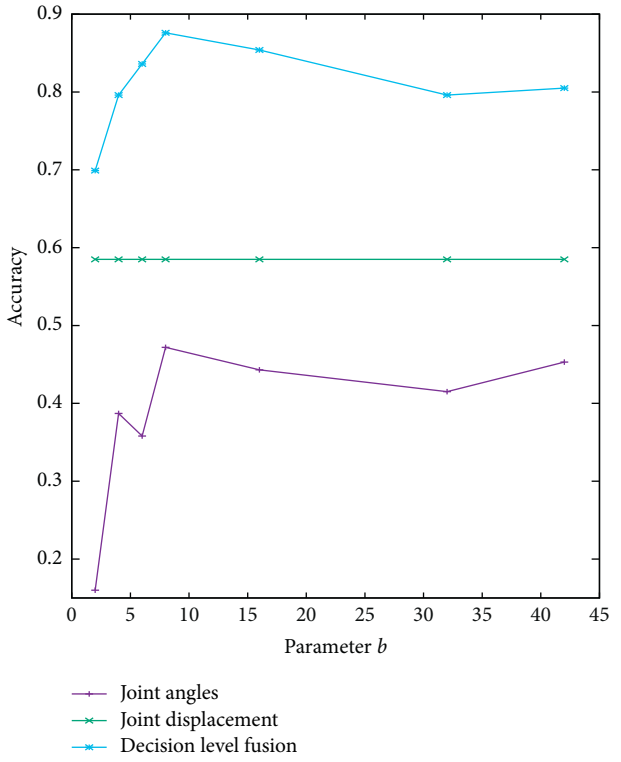

Figure 21: A graph demonstrating improvement in accuracy using decision level fusion for the MSR Action3D (AS1) dataset.

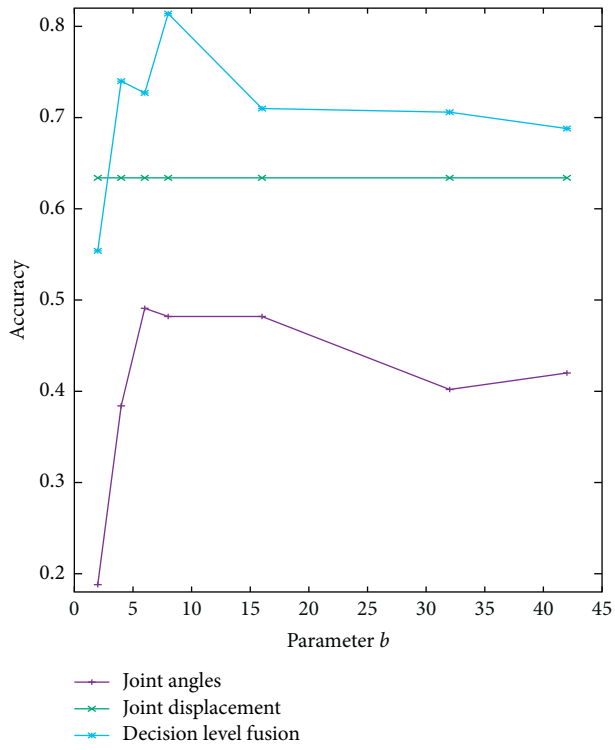

Figure 22: A graph demonstrating improvement in accuracy using decision level fusion for the MSR Action3D (AS2) dataset.

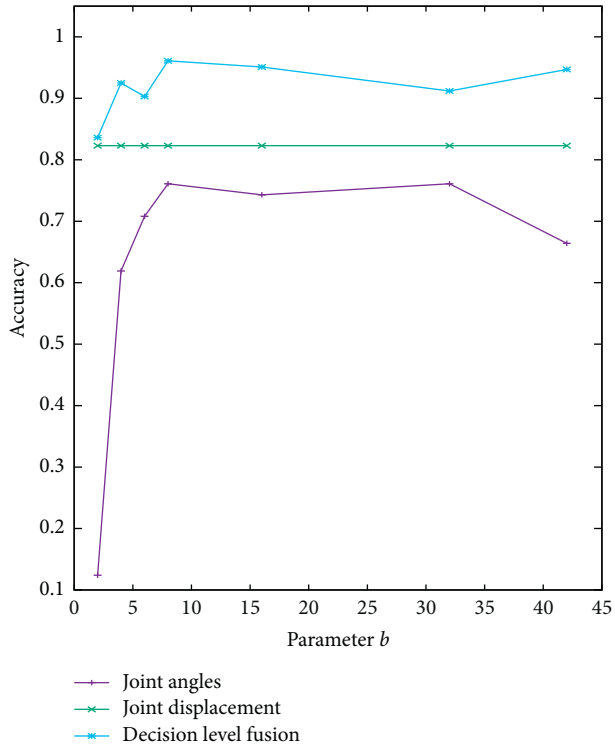

Figure 23: A graph demonstrating improvement in accuracy using decision level fusion for the MSR Action3D (AS3) dataset.

Table 3: Correlation analysis of the classifier output to find the classifier agreement for the SVM classifiers shown in Figure 1.

Dataset	Correlation coefficients of classifier output
KTH [35]	0.6308
UTKinect [36]	0.2938
MSR Action3D [37]	
AS1	0.4059
AS2	0.5619
AS3	0.5478
Average	0.48804

The accuracy of the proposed system has been analyzed using two types of combiners: a trainable combiner using a neural network and a fixed combiner using score averaging [45]. This is shown in Figure 18. The neural network is a better combiner as it is able to find the optimal weights for the fusion, whereas score averaging works as a fixed combiner with equal importance to both classifiers showing lower accuracy. The neural network-based fusion enhances the performance in terms of accuracy. It can be seen from Figures 19–23 that the fusion technique results in better performance.

The correlation analysis is performed on the output of two SVM classifiers. The result is listed in Table 3. The analysis shows that the average correlation is less than 0.5. This indicates that the classifiers moderately agree on the classification. Consequently, the fusion of these scores leads to improvement in the overall accuracy of the system.

5. Conclusions

We have developed a method for human action recognition based on skeletal joints. The proposed method extracts structural and temporal features. The structural variations are captured using joint angle, and the temporal variations are represented using joint displacement vector. The proposed approach is found to be simple as it uses single-view 2D joint locations and yet outperforms some of the state-of-the-art techniques. Also, we showed that, in the absence of Kinect sensor, pose estimation algorithm can be used as a preliminary step. The proposed method shows promising results for action recognition tasks when temporal features and structural features are fused at the score level. Thus, the proposed method is suitable for robust human action recognition tasks.

Data Availability

The references of the datasets used in the experiment are provided in the reference list.

Conflicts of Interest

The authors have no conflicts of interest regarding the publication of this paper.

References

[1] M. Bucolo, A. Buscarino, C. Famoso, L. Fortuna, and M. Frasca, "Control of imperfect dynamical systems," *Nonlinear Dynamics*, vol. 98, no. 4, pp. 2989–2999, 2019.

[2] J. K. Aggarwal and Q. Cai, "Human motion analysis: a review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428–440, 1999.

[3] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: a review," *ACM Computing Surveys*, vol. 43, no. 3, pp. 1–43, 2011.

[4] J. Ben-Arie, Z. Zhiqian Wang, P. Pandit, and S. Rajaram, "Human activity recognition using multidimensional indexing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1091–1104, 2002.

[5] V. Kellokumpu, G. Zhao, and M. Pietikäinen, "Recognition of human actions using texture descriptors," *Machine Vision and Applications*, vol. 22, no. 5, pp. 767–780, 2011.

[6] C. Gao, Y. Shao, and Y. Guo, "Human action recognition using motion energy template," *Optical Engineering*, vol. 54, no. 6, pp. 1–10, 2015.

[7] W. Xu, Z. Miao, X.-P. Zhang, and Y. Tian, "A hierarchical spatio-temporal model for human activity recognition," *IEEE Transactions on Multimedia*, vol. 19, no. 7, pp. 1494–1509, 2017.

[8] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[9] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: a review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[10] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[11] P. Wang, Z. Li, Y. Hou, and W. Li, "Action recognition based on joint trajectory maps using convolutional neural networks," in *Proceedings of the 24th ACM International Conference on Multimedia, MM '16*, pp. 102–106, Association for Computing Machinery, New York, NY, USA, October 2016.

[12] C. Li, Y. Hou, P. Wang, and W. Li, "Joint distance maps based action recognition with convolutional neural networks," *IEEE Signal Processing Letters*, vol. 24, no. 5, pp. 624–628, 2017.

[13] Y. Hou, Z. Li, P. Wang, and W. Li, "Skeleton optical spectra-based action recognition using convolutional neural networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 3, pp. 807–811, 2018.

[14] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, "RGB-D-based human motion recognition with deep learning: a survey," *Computer Vision and Image Understanding*, vol. 171, pp. 118–139, 2018.

[15] H. Rahmani, A. Mian, and M. Shah, "Learning a deep model for human action recognition from novel viewpoints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 667–681, 2018.

[16] C. Li, Y. Hou, P. Wang, and W. Li, "Multiview-based 3-D action recognition using deep networks," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 1, pp. 95–104, 2019.

[17] R. Xiao, Y. Hou, Z. Guo, C. Li, P. Wang, and W. Li, "Self-attention guided deep features for action recognition," in *Proceedings of the 2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1060–1065, Shanghai, China, July 2019.

[18] C. Li, Y. Hou, W. Li, and P. Wang, "Learning attentive dynamic maps (adms) for understanding human actions," *Journal of Visual Communication and Image Representation*, vol. 65, Article ID 102640, 2019.

[19] C. Tang, W. Li, P. Wang, and L. Wang, "Online human action recognition based on incremental learning of weighted covariance descriptors," *Information Sciences*, vol. 467, pp. 219–237, 2018.

[20] A. Edison and C. V. Jiji, "Automated video analysis for action recognition using descriptors derived from optical acceleration," *Signal, Image and Video Processing*, vol. 13, no. 5, pp. 915–922, 2019.

[21] B. Fernando, E. Gavves, M. J. Oramas, A. Ghodrati, and T. Tuytelaars, "Rank pooling for action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 773–787, 2017.

[22] P. Wang, L. Liu, C. Shen, and H. T. Shen, "Order-aware convolutional pooling for video based action recognition," *Pattern Recognition*, vol. 91, pp. 357–365, 2019.

[23] J. Hu, W. Zheng, L. Ma, G. Wang, J. Lai, and J. Zhang, "Early action prediction by soft regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 11, pp. 2568–2586, 2019.

[24] X. Yang and Y. L. Tian, "Action recognition using super sparse coding vector with spatio-temporal awareness," in *Computer Vision–ECCV 2014*, pp. 727–741, Springer International Publishing, Berlin, Germany, 2014.

[25] J.-F. Hu, W.-S. Zheng, J. Lai, S. Gong, and T. Xiang, "Exemplar-based recognition of human-object interactions," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 647–660, 2016.

[26] J. Zhu, B. Wang, X. Yang, W. Zhang, and Z. Tu, "Action recognition with actons," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 3559–3566, Sydney, Australia, December 2013.

[27] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 3551–3558, Sydney, Australia, December 2013.

[28] W. Hao and Z. Zhang, "Spatiotemporal distilled dense-connectivity network for video action recognition," *Pattern Recognition*, vol. 92, pp. 13–24, 2019.

[29] M. Ramanathan, W.-Y. Yau, and E. K. Teoh, "Human action recognition with video data: research and evaluation challenges," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 5, pp. 650–663, 2014.

[30] D. Gowsikhaa, S. Abirami, and R. Baskaran, "Automated human behavior analysis from surveillance videos: a survey," *Artificial Intelligence Review*, vol. 42, no. 4, pp. 747–765, 2014.

[31] Y. Fu, *Human Activity Recognition and Prediction*, Springer, Berlin, Germany, 2016.

[32] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using part affinity fields," 2018, https://arxiv.org/abs/1812.08008.

[33] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 1302–1310, Honolulu, HI, USA, July 2017.

[34] MSCOCO keypoint evaluation metric," http://cocodataset.org/#keypoints-eval.

[35] C. Schüldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," in *Proceedings of the 17th*

International Conference on Pattern Recognition, 2004. ICPR 2004, vol. 3, pp. 32–36, Cambridge, UK, September 2004.

[36] Y. Zhu, W. Chen, and G. Guo, "Fusing spatiotemporal features and joints for 3D action recognition," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 486–491, Portland, OR, USA, June 2013.

[37] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 9–14, San Francisco, CA, USA, June 2010.

[38] C. Schüldt, B. Caputo, C. Sch, and L. Barbara, "Recognizing human actions: a local SVM approach recognizing human actions," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004 (ICPR 2004)*, vol. 3, pp. 3–7, Cambridge, UK, September 2004.

[39] M. Yang, F. Lv, W. Xu, K. Yu, and Y. Gong, "Human action detection by boosting efficient motion features," in *Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pp. 522–529, Kyoto, Japan, September 2009.

[40] L. Xia, C. C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 20–27, IEEE, Providence, RI, USA, June 2012.

[41] Y. Zhu, W. Chen, and G. Guo, "Fusing spatiotemporal features and joints for 3D action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Portland, OR, USA, June 2013.

[42] X. Yang and Y. Tian, "Effective 3d action recognition using eigenjoints," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 2–11, 2014.

[43] E. Ohn-Bar and M. M. Trivedi, "Joint angles similarities and hog2 for action recognition," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW'13*, pp. 465–470, IEEE Computer Society, Washington, DC, USA, June 2013.

[44] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 588–595, Columbus, OH, USA, June 2014.

[45] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, John Wiley & Sons, Hoboken, NJ, USA, 2004.