applied sciences

MDPI

Article

Action Recognition Based on the Fusion of Graph Convolutional Networks with High Order Features

Jiuqing Dong [1,†], Yongbin Gao [1,*], Hyo Jong Lee [2], Heng Zhou [1], Yifan Yao [1], Zhijun Fang [1] and Bo Huang [1]

1 International Joint Research Lab of Intelligent Perception and Control, Shanghai University of Engineering Science, No. 333 Longteng Road, Shanghai 201620, China; jiuqingdong@sues.edu.cn (J.D.); hengzhou@sues.edu.cn (H.Z.); yaoyifan1995@gmail.com (Y.Y.); zjfang@sues.edu.cn (Z.F.); huangbosues@sues.edu.cn (B.H.)

2 Division of Computer Science and Engineering, CAIIT, Jeonbuk National University, Jeonju 54896, Korea; hlee@jbnu.ac.kr

* Correspondence: gaoyongbin@sues.edu.cn; Tel.: +86-182-0190-8363

† Current address: Shanghai University of Engineering Science, No. 333 Longteng Road, Songjiang District, Shanghai 201620, China.

Received: 23 December 2019; Accepted: 18 February 2020; Published: 21 February 2020

Abstract: Skeleton-based action recognition is a widely used task in action related research because of its clear features and the invariance of human appearances and illumination. Furthermore, it can also effectively improve the robustness of the action recognition. Graph convolutional networks have been implemented on those skeletal data to recognize actions. Recent studies have shown that the graph convolutional neural network works well in the action recognition task using spatial and temporal features of skeleton data. The prevalent methods to extract the spatial and temporal features purely rely on a deep network to learn from primitive 3D position. In this paper, we propose a novel action recognition method applying high-order spatial and temporal features from skeleton data, such as velocity features, acceleration features, and relative distance between 3D joints. Meanwhile, a method of multi-stream feature fusion is adopted to fuse these high-order features we proposed. Extensive experiments on Two large and challenging datasets, NTU-RGBD and NTU-RGBD-120, indicate that our model achieves the state-of-the-art performance.

Keywords: human action recognition; graph convolution; high-order feature; spatio-temporal feature; feature fusion

1. Introduct

Action recognition is a very important task in machine vision, and it can be applied to many scenes, such as automatic driving, security, human-computer interaction, and others. Therefore, in recent years, the task of analyzing the actions of people in videos has received more and more attention. The task of action recognition has many problems which are difficult to solve by using traditional methods, such as how to deal with occlusion, illumination changes, the positioning and recognition of human actions in a single frame, and extracting the relationships of frame-wise [1]. Recent approaches in depth-based human action recognition achieved outstanding performance and proved the effectiveness of 3D representation for the classification of action classes. Meanwhile, biological observation studies have also shown that even without appearance information, the locations of a few joints can effectively represent human action [2]. For identifying human action, skeleton-based human representation has attracted more and more attention for its high level of representation and robustness in regard to position and appearance changes. Recently, graph neural networks, which generalize convolutional neural networks to graphs of arbitrary structures, have been adopted in a number of applications

and have proved to be efficient for the processing of graph data [3–5]. Skeleton data also can be considered as graph structure data. Therefore, graph-based neural networks have been used for action recognition instead of the traditional CNN networks because of the successful performance. Some graph-based neural networks [6–10] are dedicated to learning both spatial and temporal features for action recognition. Meanwhile, they focus on capturing the hidden relationships among vertices in space. However, they all ignore the high-order information hidden in the skeleton data. For example, the velocity, acceleration, and relative distance information of each vertex can be extracted from the skeleton-based data. The values and directions of velocity are different for various actions. When a human is brushing his/her teeth, the hand should move up and down instead of moving back and forth. When pushing, the hand should move forward rather than backward. In a single frame, for different parts of the body, the acceleration is also varied. Additionally, there are some different actions with similar posture patterns but with different motion speeds. For example, the main difference between "grabbing another person's stuff" and "touching another person's pocket (stealing)" is the motion velocity. Therefore, taking advantage of this high-order information and extracting discriminative representations are necessary.

In this work, our main contributions are as follows:

1. We propose several high-order spatial and temporal features that are important for skeletal analysis: velocity, acceleration, and relative distance between 3D joints. Currently, the spatial features are extracted by a deep network through an adjacent matrix, while the relative distances between 3D joints are not considered in the network; we propose to use deep learning to extract the relative distances between 3D joints, which represent the postural changes of each action. Meanwhile, the widely used temporal features are extracted from the original 3D joints. The high-order motion features, such as velocity and acceleration of the joints, are nontrivial to be learned from the deep network. By explicitly calculating the high-level information as input, the deep network is able to learn higher level spatial and temporal features.
2. A multi-stream feature fusion is proposed to blend the high-order spatial and temporal features; thus, the accuracy of action recognition can be improved significantly. Our method is evaluated on the NTU-RGBD and NTU-RGBD-120 dataset, which achieves state-of-the-art performance on action detection.

2. Related Work

Recent years, NTU-RGBD [11] created a large-scale dataset for human action recognition in 2016. In 2019, NTU-RGBD has been enlarged, which is referred to NTU-RGBD-120 [12]. In addition, there are a lot of public data sets for action recognition, such as [13–19] datasets. The release of high-quality datasets have encouraged more researches on action recognition. These datasets are mainly divided into two categories, RGB-Video based and Skeleton-based. Most of the researches focus on the study of RGB video based and Skeleton-based action recognition.

2.1. RGB-Video Based Methods

In terms of video-based analysis methods, most studies consider video as a sequence of images, and then analyze the images frame by frame to learn spatial and dynamic features. Before the emergence of deep learning, the actions were identified and classified mainly by hand-designed features. [20,21] mainly introduce a method of eliminating background light flow. Their features are more focused on the description of human motion. Three hand-designed motion descriptors HOG(histogram of gradient), HOF(histogram of flow), MBH(motion boundary histograms) have been introduced, which play a very good role in the classification of motion. Since 2014, deep learning mothods have been applied to action recognition. Two-Stream Convolutional Neural Network [22] divides the convolutional neural networks into two parts, one for processing RGB images and one for processing optical flow images, which are ultimately combined and trained to extract

spatial-temporal action features. The important contribution is introduced the feature of optical flow into action recognition.

After the two-stream network [22], researchers have been trying to improve its performance, such as [23–25]. Du Tran proposed that C3D [26], for the first time, applied a 3D convolution kernel to detect action and capture the motion information on the time series. After that, the 3D convolutional-based methods became popular, prestigious methods; e.g., T3D [27].

2.2. Skeleton-Based Methods

Skeleton-based analysis benefits from the development of pose estimation algorithms and the application of depth cameras. The original skeleton data are usually estimated from RGB video by a pose estimation algorithm, or directly extracted by Kinetics cameras. In the analysis of the skeleton, how to deal with the relationship among vertices in the single frame and how to deal with the interframe relationship in the skeleton sequence are very important. Some researchers believe that a certain type of action is usually only associated with and characterized by the combinations of a subset of kinematic joints. For identifying an action, not all frames in a sequence have the same importance. In order to assign different weights to different vertices of different frames, attention mechanisms and recurrent neural networks are proposed, such as STA-LSTM proposed by Sijie Song et al. [28]. A spatial attention module adaptively allocates different attentions to different joints of the input skeleton within each frame, and a temporal attention module allocates different attention levels to different frames; e.g., Inwoong Lee et al. proposed TS-LSTM [29] and Spatio-temporal LSTMs [30]. Attention-based LSTM [28] and simple LSTM networks with part-based skeleton representation have been used in [31,32]. These methods either use complex LSTM models which have to be trained very carefully or use part-based representation with a simple LSTM model. Yan et al. proposed ST-GCN [6], which was the first graph-based neural network for action recognition. They believed that the spatial configuration of the joints and their temporal dynamics were significant for action recognition. Therefore, they constructed the spatial temporal graph, which is shown in the Figure 1. This model is formulated on top of a sequence of skeleton graphs, where each node corresponds to a joint of the human body. The edges in the single-frame skeleton are composed of physical connections of the human body, and the edges of the time dimension are composed of the connections between the corresponding joins.

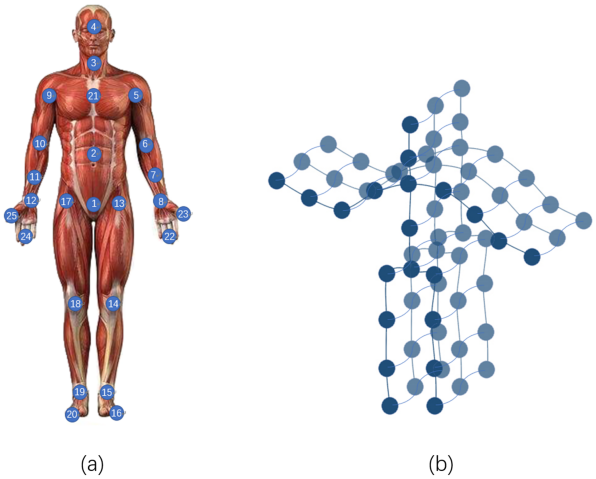

Figure 1. (**a**) The joint labeling of the NTU-RGBD and NTU-RGBD-120 datasets; the 21st node is defined as the gravity center of human. (**b**) The spatio-temporal graph used in ST-GCN [6].

Kalpit divided the skeleton graph into four subgraphs with joints shared across them and taught a recognition model using a part-based graph convolutional network [8]. AGC-LSTM [10] can not only

capture features in spatial configuration and temporal dynamics but also explore the co-occurrence relationship between spatial and temporal domains.

In the previous work for action recognition task based on skeleton, only the 3D coordinate information of the joints was utilized. Nevertheless, how to effectively extract discriminative spatial and temporal features is still a challenging problem. Therefore, in this work, we put more attention on the high-order information features. The features we proposed are efficient for action recognition, and the feature fusion method we used is easy to implement.

3. Proposed Graph Convolutional Network with High-Order Features

A graph is good for representing spatial and temporal information. We can transform a frame of the skeleton data to a topological map, which contains joint and edge subsets as shown in Figure 1. A graph neural network can model joint features and structure features simultaneously, which is good method for graph data learning. As the convolution of an image is performed by a convolution kernel with a regular shape, the graph convolution layer is applied on the graph data to generate a high-level feature. Our network model is based on the 2s-AGCN [7]. The overall pipeline of our model is shown in Figure 2, where AGCN is a multi-layer graph convolution network. The networks we proposed consist of five sub-networks. Each sub-network is used to extract a variety of spatial and temporal features. Joint-coordinates, bone, and relative distance are spatial features, and velocity and acceleration of joints and bones are temporal features.

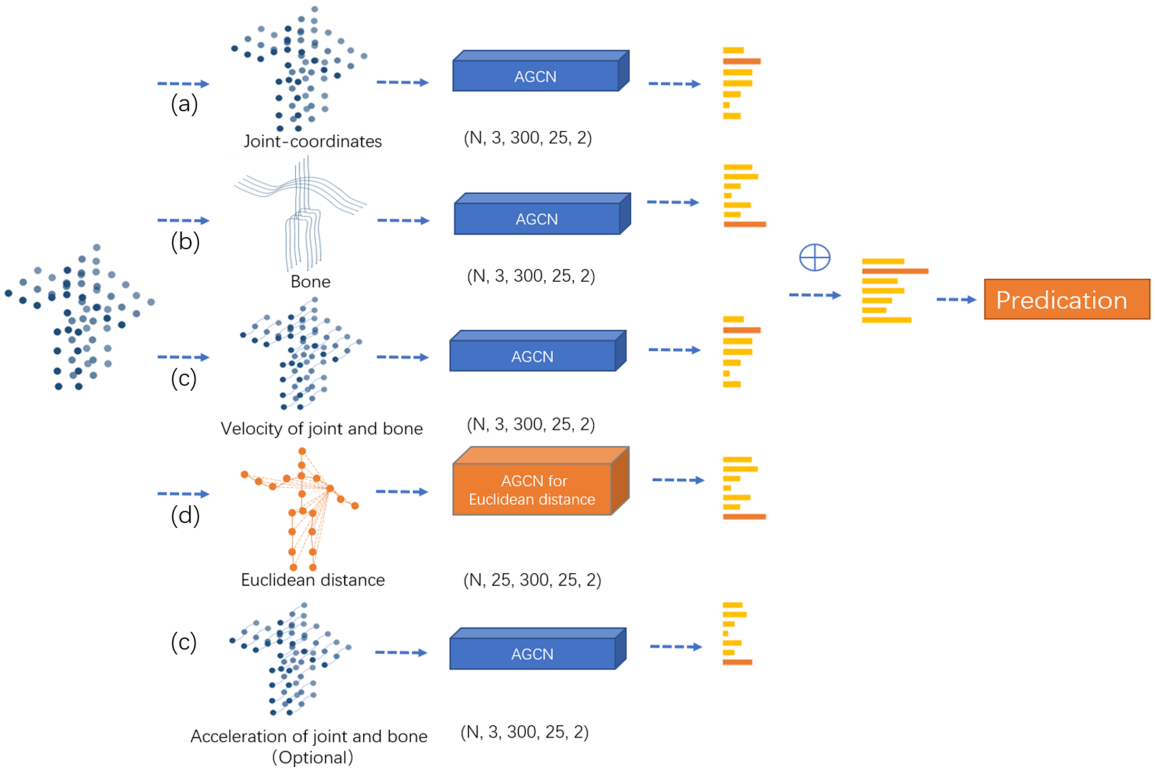

Figure 2. Illustration of the overall architecture of the MS-AGCN. The structure of the AGCN in blue is the same. The only difference between blue and orange is the number of input channels. The final score to obtain the prediction. The shape of input data is presented. (**a**) The joint feature, which is extracted from 3D coordinates of all joints. (**b**) The bone feature, which contains edge information. (**c**) The velocity feature and the acceleration feature, which are calculated from consecutive frames to obtain the temporal feature. (**d**) The relative distance feature of 3D joints; each joint contains relative distance information from others, and we only use one joint as an illustration in the figure.

3.1. Improved Graph Convolutional Network

The implementation of the graph convolution in the spatial domain is not straightforward. Concretely, the input of every layer in the network is actually a $C \times T \times N$ tensor, where C, T, and N are the number of channels, frames, and vertices, respectively. Furthermore, the edge importance matrix was proposed in ST-GCN [6], aiming to distinguish the importance of the edge of skeletons for different actions. The graph convolution operation is formulated as Equation (1) in [6]:

$$f_{out}^{n} = \sum_{s}^{S_v} W_s * (f_{out}^{n-1} * A_S) \odot M_k \tag{1}$$

where the matrix A is initial adjacency matrix proposed in [6], and S is the subset of matrix A, which is similar to the $N \times N$ adjacency matrix. W_s is the weight vector of the $C_{out}^{n} \times C_{out}^{n-1} \times 1 \times 1$ convolution operation, where $*$ denotes the matrix product. M is the edge importance matrix of $n * n$, which is dot multiplied by matrix A.

Equation (1) shows that the edge importance matrix M_k is dot multiplied to A_s. That means that if one of the elements in A_s is zero, it will always be zero, which is unreasonable. Thus, we change the computing method. We add another attention matrix M_{k1} and then multiply matrix M_k. In addition, we use the similarity matrix in 2S-AGCN [7] to estimate the similarity of two joints, and determine whether there is a connection between two vertices and how strong the connection is. Finally, Equation (1) is transformed into Equation (2):

$$f_{out}^{n} = \sum_{s}^{S_v} W_s * (f_{out}^{n-1} * (A_S \oplus M_{k1} \oplus S_k)) \odot M_k \tag{2}$$

where $\oplus$ denotes matrix addition. S_k is the similarity matrix proposed in 2s-AGCN [7]. M_{k1} is a new attention matrix we added.

For the temporal domain, since the number of neighbors for each vertex is fixed as two (corresponding joints in the two consecutive frames), it is straightforward to perform the graph convolution similar to the classical convolution operation. Concretely, we perform $K_t * 1$ convolution on the output feature map calculated above, where K_t is the kernel size of temporal convolution. Spatial convolution is combined with temporal domain convolution into a graph convolution module. The details are shown in Figure 3:

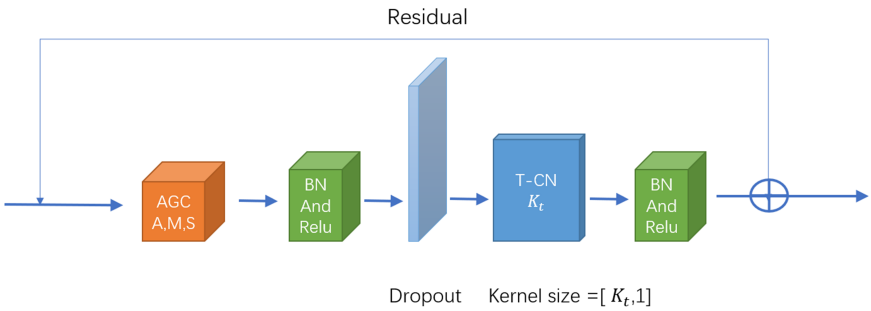

Figure 3. An AGCN block consists of spatial GCN(AGC), temporal GCN(T-CN), and other operations: batch normalization (BN), Relu, dropout, and the residual block. A, M, and S in AGC represent the adjacency matrix, edge importance matrix, and similarity matrix, respectively.

3.2. High-Order Spatial Features

For spatial features in a single frame, we propose combining the bone feature with the relative distance feature of 3D joint. From the Figure 2b,d we can directly get the information contained by these two features.

Bone feature: Shi et al. [7] argued that the coordinate information of the joints could not represent the action of the human body well. Therefore, they proposed the second-order information, which is referred to bone feature, as a feature to enhance the performance on action recognition. The bone feature is extracted from bone data, which includes the length and the direction. Each bone is a human physical connection between joints; Shi defined the person's center of gravity as the target joint; and all directions of the bone are centripetal. Each bone is connected to two joints. The distance from joint $j_1(x_1, y_1, z_1)$ to center of gravity is farther than $j_2(x_2, y_2, z_2)$. The vector representation of bone between j_1 and j_2 is $e_{j_1,j_2} = (x_1 - x_2, y_1 - y_2, z_1 - z_2)$. The direction is from j_1 to j_2.

The number of bones is always one less than the number of joints because each bone is connected to two joints. In order to keep the quantity consistent, we set the empty bone at the center of gravity. The input dimension of the bone network thereby can be the same as the joint network.

Relative distance feature of 3D Joints: We find that the feature extracted from relative distance between 3D joints is useful for skeleton data. For example, nodding requires only a head movement. The acceleration/velocity values of all vertices are zero, except for those of head-related joints. However, the relative distance from the head to the other joints must be changing at all frames and it can not be zero. In addition, we set the distance between the vertex and itself as zero, so the relative distance information of one vertex is 25-dimensional. For a single frame skeleton, we can use a 25×25 matrix to represent it. This matrix is a diagonal matrix, and the principal diagonal elements are zeros. The shape of relative distance information is $(N, 25, T, 25, 2)$, while the shape of other information is $(N, 3, T, 25, 2)$, where N denotes the batch-size we set and T denotes the length of one action sequence.

3.3. High-Order Temporal Features

For temporal features in a single frame, we propose the velocity feature and the acceleration feature. From the Figure 2c, we can directly get the information contained by these two features.

Velocity feature: Velocity features of an action are very crucial for action recognition. Learning velocity features can be relatively complemented with learning features of the joint and bone. For skeleton data, we calculate the motion velocity information of each vertex. The velocity of vertex v_1 is equal to the coordinate of v_1 in the next frame minus the current frame. We can obtain the velocity in three directions (x, y, z), which is helpful for analyzing the action. Velocities of different orientations correspond to different changes. Therefore, velocity analysis in each orientation of the vertex is effective for the final prediction. $j_1^t(x_1^t, y_1^t, z_1^t)$ denotes the coordinates of joint j_1 at t frame. $j_1^{t+1}(x_1^{t+1}, y_1^{t+1}, z_1^{t+1})$ denotes the coordinates of joint j_1 at $T + 1$ frame. The velocity of $v_1^t(v_{x1}^t, v_{y1}^t, v_{z1}^t)$ at t frame can be written as:

$$v_1^t(v_{x1}^t, v_{y1}^t, v_{z1}^t) = j_1^{t+1} - j_1^t = (x_1^{t+1} - x_1^t, y_1^{t+1} - y_1^t, z_1^{t+1} - z_1^t) \tag{3}$$

For all joints, Equation (3) is transformed into Equation (4):

$$v^t(v_x^t, v_y^t, v_z^t) = j^{t+1} - j^t = (x^{t+1} - x^t, y^{t+1} - y^t, z^{t+1} - z^t) \tag{4}$$

where v denotes the velocity of all joints in a single frame. Moreover, we calculate the velocity of the edge between the two joints, which is the velocity of the bone. The calculation method of velocity of the bone is the same as that of the joints. We use the 3D velocity of the bone as a feature and feed it into the network. More details of the training results and comparison experiments are provided in Section 4.

Acceleration feature: Acceleration is a physical quantity used to describe the change in velocity. Acceleration is helpful for analyzing action. In one skeleton sequence, the velocities of joints may have different changes. Some joints move at a constant velocity, while other joints accelerate. The acceleration of the joint is equal to the velocity of the current frame minus the corresponding joint of the previous frame. Its feature dimensions are also three-dimensional. Basically, that means that the

calculation method of acceleration information is the same as that of the velocity information. Therefore, the features extracted from velocity and acceleration information are similar, while the acceleration uses more frames to calculate the high-order motion. We can calculate acceleration information based on Equation (5) as follows:

$$a_1^t = v_1^{t+1} - v_1^t = (v_{x1}^{t+1} - v_{x1}^t, v_{y1}^{t+1} - v_{y1}^t, v_{z1}^{t+1} - v_{z1}^t) \tag{5}$$

For all joints, Equation (5) is transformed into Equation (6):

$$a^t = v^{t+1} - v^t = (v_x^{t+1} - v_x^t, v_y^{t+1} - v_y^t, v_z^{t+1} - v_z^t) \tag{6}$$

where a_1^t denotes the acceleration of joint j_1 at t frame. v_1^{t+1} and v_1^t denote the velocity of joint j_1 at $t+1$ and t frames, respectively, and a^t denotes the acceleration of all joints in t frame.

3.4. High-Order Features Fusion

Joint Feature: For both of NTU-RGBD and NTU-RGBD-120 datasets, the joint features are extracted from the 3D coordinates of the skeleton sequence. Joint features are fundamental and important features for the skeleton data. Joints coordinates contain abundant spatial and temporal information. Our baseline is a single stream of 3D joint. We also put the joint data into our neural networks to extract joint feature as shown in Figure 2a.

Features extracted only by 3D joints are not enough for action recognition. We propose several pieces of high-order information as input which is effective for action recognition. In front of the input layer, a batch normalization layer is added to normalize the input data. A global average pooling layer is added at the end of the network to pool feature maps of different samples to the same size. Both the input and output of the network are graph-structures data in the graph convolution. The last graph convolution layer generates a discriminative feature and puts it into the standard soft-max classifier. The final score is the weighted summation of the scores of five streams, which is used to predict the action label. We believe that the information contained in the joints, bones, and relative distance is the most fundamental and important. Therefore, these features should be set large weights. The velocity and acceleration information are auxiliary features that strengthen the temporal relationship. These features should be set small weights. The weighted summation method can be formulated as Equation (7):

$$S_f = S_a W_a + S_b W_b + S_c W_c + S_d W_d \tag{7}$$

where S_a, S_b, S_c, and S_d denote the score of joint, bone, joint and bone velocity, and relative distance, respectively. S_f denotes the final score. W_* denotes the weights of scores.

4. Experiments

4.1. Datasets

NTU-RGBD [11] contains 56,880 video clips of 60 actions. The samples were taken from 40 different people by using a Kinect v2 camera. The ages of subjects are between 10 and 35. They used three cameras simultaneously to capture three different horizontal views from the same action. For the camera position setting: the three cameras were at the same height but three different horizontal angles: $-45°$, $0°$, $+45°$ [11]. The dataset provides two methods to evaluate the performance of action classification: cross-subject and cross-view. The training set of cross-subject includes 40,320 samples, which consists of actions performed by 20 subjects. The testing set contains 16,560 samples, which consists of samples taken by another 20 subjects [11]. The cross-subject training set includes 37,920 samples taken by Cameras 2 and 3, and testing set contains 18,960 samples taken by Camera 1.

NTU-RGBD-120 [12] is an extension of NTU-RGBD, which is much larger and provides much more variation of environmental conditions, subjects, camera views, etc. It contains 114,480 video clips of 120 actions. The ages of subjects are between 10 and 57, and heights are between 1.3 m and 1.9 m. The dataset provides two criteria to evaluate the performance of action classification: cross-subject and cross-setup. The training set of cross-subject includes 63,026 samples, which consists of actions performed by 53 subjects. The testing set contains 50,919 samples taken by another 53 subjects [12]. The cross-setup training set includes 54,468 samples consisting of the samples with even collection setup IDs. Testing set contains 59,477 samples, which consists of samples with odd setup IDs. Different setup IDs correspond to changeable vertical heights of the cameras and their distances to the subjects.

4.2. Data Augmentation

During the experiment, we performed the data analysis and gathered statistics on the samples of incorrect recognition. Experiments show that the graph convolution is efficient for the large displacement. However, we also found that the fine-grained actions were more likely to predict incorrectly. Thus, we made a data augmentation for these action categories, which consists of 16 categories. They are drinking water, eating a meal/snack, brushing teeth, clapping, reading, writing, wearing a shoe, taking off a shoe, making a phone call, playing with the phone/tablet, typing on the keyboard, pointing to something with a finger, taking a selfie, sneezing, coughing, touching the head (headache), and touching the neck (neckache). Considering that the datasets were collected in-three-dimensions, and in order to maintain the relative position of the joints unchanged, we performed the rotation of the skeleton data with angles of $\pm 2^{\circ}$.

4.3. Training Detail

All experiments were conducted on the Pytorch deep learning framework. Stochastic gradient descent (SGD) with Nesterov momentum (0.9) was applied as the optimization strategy. The batch size was 64. Cross-entropy was selected as the loss function to backpropagate gradients. The weight decay was set to 0.0001. For both the NTU-RGBD [11] and NTU-RGBD-120 [12] datasets, there are at most two people in each sample of the dataset. If the number of bodies in the sample was less than two, we padded the second body with 0. The maximum number of frames in each sample is 300. For samples with less than 300 frames, we repeated the samples until it reached 300 frames. The learning rate was set as 0.1 and was divided by 10 at the 30th epoch and 40th epoch. The training process was ended at the 50th epoch.

4.4. Ablation Experiment

In Section 3, we add the joints feature, bones feature, joint-velocity feature, bone-velocity feature, and relative distance feature for action recognition. Since the acceleration feature is similar to the velocity feature, the accuracy after fusion is not significantly improved. The ablation studies of different features are shown in Tables 1 and 2, where J, B, JV, BV, and RD denote that features of joint, bone, joint-velocity, bone-velocity and relative-distance, respectively. Obviously, the multi-feature fusion method outperforms the single-feature-based methods on two benchmark evaluations.

Table 1. Comparisons of the validation accuracy with different input modalities on a cross-subject benchmark of the NTU-RGBD dataset.

Methods	Accuracy (%) (No Augmentation)	Accuracy (%) (Augmentation)
Joint	86.7	87.7
Bone	87.0	88.1
Joint-Velocity	86.1	86.8
Bone-Velocity	85.4	86.9
Relative-Distance	87.1	87.5
J+B+JV+BV+RD	**90.5**	**91.7**

Table 2. Comparisons of the validation accuracy with different input modalities on a cross-view benchmark of the NTU-RGBD dataset.

Methods	Accuracy (%) (No Augmentation)	Accuracy (%) (Augmentation)
Joint	93.0	93.8
Bone	93.4	94.3
Joint-Velocity	93.0	93.5
Bone-Velocity	92.7	93.4
Relative-Distance	93.2	94.0
J+B+JV+BV+RD	**95.8**	**96.8**

Tables 3 and 4 are the results on NTU-RGBD-120 dataset. The results also illustrate that the multi-feature fusion method is more effective. The recognition accuracy of our model in NTU-RGBD-120 is slightly lower than the accuracy of NTU-RGBD. The major reasons leading to this result were: (1) NTU-RGBD-120 adds some fine-grained object-related individual actions. For these actions, the body movements are not significant, and the sizes of the objects involved are relatively small; e.g., when "counting money" and "playing magic cube". (2) Some fine-grained hand/finger motions are added in NTU-RGBD-120. Most of the actions in the NTU-RGBD dataset have significant body and hand motions, while the NTU-RGBD-120 dataset contains some actions that have fine-grained hand and finger motions, such as "making an ok sign" and "snapping fingers". (3) The third limitation is the large number of action categories. When only a small set of classes is available, each can be very distinguishable by finding a simple motion pattern or even by the appearance of an interacted object. However, when the number of classes increases, similar motion patterns and interacted objects will be shared among different classes, which makes the action recognition much more challenging.

Table 3. Comparisons of the validation accuracy with different input modalities on cross-subject benchmark of NTU-RGBD-120 dataset.

Methods	Accuracy (%) (No Augmentation)
Joint	80.7
Bone	81.2
Joint-Velocity	78.5
Bone-Velocity	79.2
Relative-Distance	81.5
J+B+JV+BV+RD	**86.4**

Table 4. Comparisons of the validation accuracy with different input modalities on cross-setup benchmark of NTU-RGBD-120 dataset.

Methods	Accuracy (%) (No Augmentation)
Joint	84.3
Bone	84.5
Joint-Velocity	81.4
Bone-Velocity	82.3
Relative-Distance	84.5
J+B+JV+BV+RD	**89.2**

4.5. Comparison with the State-of-the-Art

We compare the final model with the state-of-the-art skeleton-based action recognition methods on NTU-RGBD dataset and NTU-RGBD-120 dataset. The results of the comparison are shown in Tables 5 and 6. The methods used for comparison include the handcraft-feature-based methods [33], RNN-based methods [28,29,34,35], CNN-based methods [36,37], and GCN-based methods [6–10]. From Table 5, we can see that our proposed method achieves the best performances of 96.8% and 91.7% in terms of two criteria on the NTU-RGBD dataset.

Since the NTU-RGBD-120 dataset was released in 2019, there are no related works on this dataset yet. Therefore, we only cite the result of relevant methods mentioned in the original paper of this dataset. As shown in the Table 6, our method is significantly better than the others.

Table 5. Comparisons of the validation accuracy with state-of-the-art methods on the NTU-RGBD dataset.

Methods	Cross-Subject (%)	Cross-View (%)
Lie Group(2014) [33]	50.1	82.8
Trust Gate ST-LSTM(2016) [29]	69.2	77.7
Two-stream RNN(2017) [34]	71.3	79.5
STA-LSTM(2017) [28]	73.4	81.2
VA-LSTM(2017) [35]	79.4	87.6
SR-TSL(2018) [37]	84.8	92.4
HCN(2018) [36]	86.5	91.1
ST-GCN (2018) [6]	81.5	88.3
AS-GCN(2018) [9]	86.8	94.2
PB-GCN (2018) [8]	87.5	93.2
2s-AGCN(2019) [7]	88.5	95.1
AGC-LSTM(2019) [10]	89.2	95.0
ours	**91.7**	**96.8**

Table 6. The results of different methods, which are designed for 3D human activity analysis, using the cross-subject and cross-setup evaluation criteria on the NTU RGB+D 120 dataset.

Methods	Cross-Subject (%)	Cross-Setup (%)
ST-LSTM(2016) [29]	55.7	57.9
Internal Feature Fusion(2017) [38]	58.2	60.9
GCA-LSTM(2017) [30]	58.3	59.2
Multi-Task Learning Network(2017) [39]	58.4	57.9
FSNet(2018) [40]	59.9	62.4
Skeleton Visualization (Single Stream)(2017) [41]	60.3	63.2
Two-Stream Attention LSTM(2018) [38]	61.2	63.3
Multi-Task CNN with RotClips(2018) [42]	62.2	61.8
Body Pose Evolution Map(2018) [43]	64.6	66.9
ours	**86.4**	**89.4**

5. Conclusions

In this work, we propose several spatial and temporal features which are more effective for skeleton-based action recognition. By blending these high-order features, the deep network highlights the spatial changes and temporal changes of the 3D joints, which are crucial for action recognition. It is worth mentioning that the multi-feature fusion method outperforms the single-feature-based method. For each high-order feature added, the accuracy of the final result is improved by about 1%. On the cross-subject and cross-view evaluation criteria of the NTU-RGBD dataset, blending high-order features can improve the accuracy by 3.8% and 2.8%, respectively. What is more, for the cross-subject and cross-setup evaluation criteria of the NTU-RGBD-120 dataset, blending high-order features can improve the accuracy by 5.7% and 4.9%, respectively. The results prove the efficiency of the high-order features and indicate that the performance of our model is the state-of-the-art. In future work, we will add visual information to solve the problems caused by object-related individual actions, and prepare to add some part-based features to solve the problem of fine-grained actions.

6. Patents

Using the method we proposed in this article, we published an invention patent. There is some information about our invention patent. More details can be searched for publication number (CN110427834A) from the official website of the State Intellectual Property Office of China.

China Patent: Jiuqing Dong, Yongbin Gao, Yifan Yao, Jia Gu, and Fangzheng Tian. Behavior recognition system and method based on skeleton data [P]. CN110427834A,2019-11-08.

Author Contributions: Conceptualization, J.D., Y.G. and H.J.L.; methodology, J.D. and B.H.; software, J.D. and H.Z.; validation, J.D., Y.Y. and H.Z.; formal analysis, B.H.; investigation, Y.G.; resources, Z.F.; data curation, J.D.; writing—original draft preparation, J.D.; writing—review and editing, H.J.L. and Y.G.; visualization, H.Z.; supervision, Y.G.; project administration, Z.F.; funding acquisition, Y.G. and Z.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the Youth Program of National Natural Science Foundation of China (Grand No.:61802253), the National Natural Science Foundation of China (Grand No.:61831018, 61772328). In part by the Chenguang Talented Program of Shanghai under Grand 17Cg59. In part by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (GR2019R1D1A3A03103736).

Acknowledgments: We thank LEE from Jeonbuk National University for his great help. We also thank anonymous reviewers for their careful reading and insightful comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dai, R.; Gao, Y.; Fang, Z.; Jiang, X.; Wang, A.; Zhang, J.; Zhong, C. Unsupervised learning of depth estimation based on attention model and global pose optimization. *Signal Process. Image Commun.* **2019**, *78*, 284–292. [CrossRef]
2. Johansson, G. Visual perception of biological motion and a model for its analysis. *Percept. Psychophys.* **1973**, *14*, 201–211. [CrossRef]
3. Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems 29*; Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; pp. 3844–3852. Available online: http://papers.nips.cc/paper/6081-convolutional-neural-networks-on-graphs-with-fast-localized-spectral-filtering (accessed on 20 February 2020).
4. Bruna, J.; Zaremba, W.; Szlam, A.; LeCun, Y. Spectral networks and locally connected networks on graphs. *arXiv* **2013**, arXiv:1312.6203.
5. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
6. Yan, S.; Xiong, Y.; Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
7. Shi, L.; Zhang, Y.; Cheng, J.; Lu, H. Two-Stream Adaptive Graph Convolutional Networks for Skeleton-Based Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 12026–12035.
8. Thakkar, K.; Narayanan, P. Part-based graph convolutional network for action recognition. *arXiv* **2018**, arXiv:1809.04983.
9. Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; Tian, Q. Actional-Structural Graph Convolutional Networks for Skeleton-based Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3595–3603.
10. Si, C.; Chen, W.; Wang, W.; Wang, L.; Tan, T. An Attention Enhanced Graph Convolutional LSTM Network for Skeleton-Based Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 1227–1236.
11. Shahroudy, A.; Liu, J.; Ng, T.T.; Wang, G. NTU RGB+ D: A large scale dataset for 3D human activity analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
12. Liu, J.; Shahroudy, A.; Perez, M.L.; Wang, G.; Duan, L.Y.; Chichung, A.K. NTU RGB+ D 120: A Large-Scale Benchmark for 3D Human Activity Understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**. [CrossRef] [PubMed]
13. Li, W.; Zhang, Z.; Liu, Z. 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, San Francisco, CA, USA, 13–18 June 2010; pp. 9–14.
14. Sung, J.; Ponce, C.; Selman, B.; Saxena, A. Human activity detection from RGBD images. In Proceedings of the Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 7–11 August 2011.
15. Koppula, H.S.; Gupta, R.; Saxena, A. Learning human activities and object affordances from rgb-d videos. *Int. J. Robot. Res.* **2013**, *32*, 951–970. [CrossRef]
16. Bloom, V.; Makris, D.; Argyriou, V. G3D: A gaming action dataset and real time action recognition evaluation framework. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 7–12.
17. Liu, C.; Hu, Y.; Li, Y.; Song, S.; Liu, J. PKU-MMD: A large scale benchmark for continuous multi-modal human action understanding. *arXiv* **2017**, arXiv:1703.07475.
18. Jhuang, H.; Garrote, H.; Poggio, E.; Serre, T.; Hmdb, T. A large video database for human motion recognition. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; Volume 4, p. 6.

19. Soomro, K.; Zamir, A.R.; Shah, M. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv* **2012**, arXiv:1212.0402.
20. Wang, H.; Kläser, A.; Schmid, C.; Liu, C.L. Dense trajectories and motion boundary descriptors for action recognition. *Int. J. Comput. Vis.* **2013**, *103*, 60–79. [CrossRef]
21. Wang, H.; Schmid, C. Action recognition with improved trajectories. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 3551–3558.
22. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. Advances in Neural Information Processing Systems (NIPS). 2014. Available online: http://papers.nips.cc/paper/5353-two-stream-convolutional (accessed on 20 February 2020).
23. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*; Springer: Berlin, Germany, 2016; pp. 20–36.
24. Lan, Z.; Zhu, Y.; Hauptmann, A.G.; Newsam, S. Deep local video feature for action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 1–7.
25. Zhou, B.; Andonian, A.; Oliva, A.; Torralba, A. Temporal relational reasoning in videos. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 803–818.
26. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3D convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
27. Diba, A.; Fayyaz, M.; Sharma, V.; Karami, A.H.; Arzani, M.M.; Yousefzadeh, R.; Van Gool, L. Temporal 3D convnets: New architecture and transfer learning for video classification. *arXiv* **2017**, arXiv:1711.08200.
28. Song, S.; Lan, C.; Xing, J.; Zeng, W.; Liu, J. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
29. Liu, J.; Shahroudy, A.; Xu, D.; Wang, G. Spatio-temporal lstm with trust gates for 3D human action recognition. In *European Conference on Computer Vision*; Springer: Berlin, Germany, 2016; pp. 816–833.
30. Liu, J.; Wang, G.; Hu, P.; Duan, L.Y.; Kot, A.C. Global context-aware attention LSTM networks for 3D action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1647–1656.
31. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1110–1118.
32. Tao, L.; Vidal, R. Moving poselets: A discriminative and interpretable skeletal motion representation for action recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2015; pp. 61–69.
33. Vemulapalli, R.; Arrate, F.; Chellappa, R. Human action recognition by representing 3D skeletons as points in a lie group. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 588–595.
34. Wang, H.; Wang, L. Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 499–508.
35. Zhang, P.; Lan, C.; Xing, J.; Zeng, W.; Xue, J.; Zheng, N. View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In Proceedings of the IEEE International Conference on Computer Vision, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2126.
36. Li, C.; Zhong, Q.; Xie, D.; Pu, S. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. *arXiv* **2018**, arXiv:1804.06055.
37. Si, C.; Jing, Y.; Wang, W.; Wang, L.; Tan, T. Skeleton-based action recognition with spatial reasoning and temporal stack learning. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 103–118.
38. Liu, J.; Shahroudy, A.; Xu, D.; Kot, A.C.; Wang, G. Skeleton-based action recognition using spatio-temporal LSTM network with trust gates. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 3007–3021. [CrossRef]

39. Ke, Q.; Bennamoun, M.; An, S.; Sohel, F.; Boussaid, F. A new representation of skeleton sequences for 3D action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3288–3297.
40. Liu, J.; Shahroudy, A.; Wang, G.; Duan, L.Y.; Chichung, A.K. Skeleton-based online action prediction using scale selection network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**. [CrossRef]
41. Liu, M.; Liu, H.; Chen, C. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognit.* **2017**, *68*, 346–362. [CrossRef]
42. Ke, Q.; Bennamoun, M.; An, S.; Sohel, F.; Boussaid, F. Learning clip representations for skeleton-based 3D action recognition. *IEEE Trans. Image Process.* **2018**, *27*, 2842–2855. [CrossRef] [PubMed]
43. Liu, M.; Yuan, J. Recognizing human actions as the evolution of pose estimation maps. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1159–1168.