# KEY JOINTS SELECTION AND SPATIOTEMPORAL MINING FOR SKELETON-BASED ACTION RECOGNITION

*Zhikai Wang*[1]*,Chongyang Zhang*[1,2*]*,Wu Luo*[1]*, and Weiyao Lin*[1]

[1]School of Electronic Information and Electrical Engineering,
Shanghai Jiao Tong University, Shanghai 200240, China
[2]Shanghai Key Lab of Digital Media Processing and Transmission, Shanghai 200240, China
[*]Corresponding email: sunny_zhang@sjtu.edu.cn

## ABSTRACT

Trajectories and spatiotemporal attention model have been successfully used in skeleton-based action recognition. Most existing methods focus more attention on temporal structure mining. However, only a few local joints and their position features (e.g., critical position changes of hand, head, leg etc.) are responsible for the action label. In this work, we introduce a novel action recognition framework using Key Joints Selection and Spatiotemporal Mining, which can identify both key joints and their position & velocity histogram as well as trajectory features for action classification. First, histogram of human joints position and velocity are developed to enhance the spatiotemporal structure representation of existing trajectory-based methods. Second, the key joints are selected according to their information gains, and then their position & velocity histograms are weighted and composed with trajectory features to form one richer representation for final action classification. Experiments on two widely-tested benchmark datasets show that by combining the strength of both richer features and key joints selecting, our method can achieve state-of-the-art or competitive performance compared with existing results using sophisticated models such as deep learning, with advantages regarding the recognition accuracy and robustness.

***Index Terms—*** Action recognition, key joints, position & velocity histograms, spatiotemporal mining, skeleton

## 1. INTRODUCTION

Action recognition has attracted much attention due to its importance in many applications. Thanks to the development of commodity RGB-D cameras, skeleton-based action recognition has drawn considerable attention in the computer vision community recently [1, 2]. Although the recent advances in deep convolutional networks (ConvNets) have brought some improvements on action recognition [3], it remains a difficult challenge due to the problem that they require a large number of labeled videos for training [4], while most available datasets, especially the skeleton-based 3D action datasets, are relatively small. Thus, traditional handcrafted feature based methods are still useful for 3D action recognition.

In recent years, many learning-based methods have been proposed for skeleton-based action recognition. Three categories of approaches are often used: spatial modeling, temporal modeling, and spatiotemporal modeling. The modeling in the spatial domain is mainly driven by the fact that an action is usually only characterized by the interactions or combinations of a subset of skeleton joints [5]. In HBRNN [6], skeletons are decomposed into five parts and a hierarchical recurrent neural network is built to model the relationship among these parts. Similarly, in [7] a part-aware model is proposed to construct the relationship between body parts. In S-MIJ [8], the most informative joints are selected simply based on measures such as mean or variance of joint angle trajectories. On the temporal domain, temporal pyramid matching [9], and dynamic time warping [10] or segmentation [11] are the common methods for temporal modeling. In [12], short-term and long-term temporal models are combined to form a multi-model framework. Many efforts on spatiotemporal modeling are also proposed: In [13], LSTM model is extended to spatiotemporal domain to analyze skeletons, spatiotemporal vector of locally max pooled features are developed in [14], and spatiotemporal attention parts are selected in [2].

Good features are crucial to reliable action recognition. Although features developed from existing works have shown big improvements in many domains, most of the mentioned methods pay more attention to temporal trajectory features while largely ignore key local parts' spatial patterns: position and velocity distribution of the key body parts. Without this type patterns, they have limitations in precisely differentiating the ambiguity among fine-grained action classes due to the subtle inter-class trajectory differences. For example, the actions of Horizontal-arm-wave and High-arm-wave have similar hands trajectory. Conversely, the hands height histograms of these two actions have notable differences (Fig. 1), which can be used to distinguish them more easily. In another case, actions with inverse part activities, such as pull and push, are easy to be confused due to the similar trajectory and

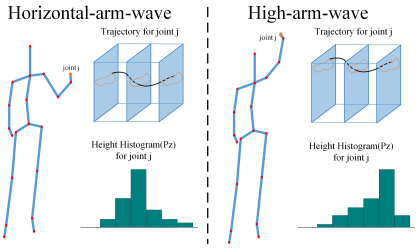


**Fig. 1**. Examples which have similar trajectories but discriminative height histograms

position histogram features. Thus, velocity histogram, which includes the motion direction feature, can be adopted to distinguish this type of actions. With the above motivations, we propose a novel action descriptor, which focus on two key issues for improving the recognition performance: (1) deriving richer features, position & velocity histograms are developed and combined with trajectory patterns, to better represent fine-grained actions, (2) selecting the key joints, enhancing their weights to better leverage the unequal contribution of different joints in different stages for action recognition.

## 2. PROPOSED METHOD

In this section, we introduce how the proposed method is established. As illustrated in Fig. 2, the procedure of proposed method includes three main steps: firstly, position & velocity histogram and trajectory are abstracted from the skeleton-based action sequence simultaneously, and then the trajectories are scaled into S (S=10 for example) temporal length averagely; secondly, key joints are selected using information gain, and then spatiotemporal weights for selected joints are calculated to obtain weighted histogram features; at last, the histogram concatenated with scaled trajectory is encoded into fisher vector, which is taken as the final feature for the action classifier.

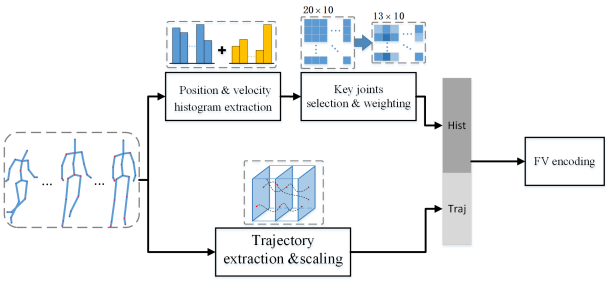


**Fig. 2**. The pipeline of the proposed framework. Firstly, two features are extracted and processed from skeleton-based action sequences: 1) position & velocity histogram extraction and then key joints selection & weighting, and 2) trajectory extraction and scaling . Secondly, the two type features are concatenated and then encoded into fisher vector as final feature for classification.

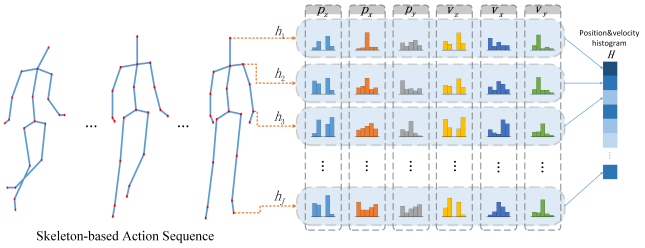


**Fig. 3**. The construction of histogram feature. For each joint, there are six dimensions of histograms: the x,y,z dimensions of position and velocity, respectively. The six histograms for every joints are then concatenated to form one full position & velocity histogram for each action sequence.

### 2.1. Constructing position & velocity histogram

Trajectories have been successfully used in skeleton-based action recognition as a traditional feature. However, the trajectory itself cannot reflect the statistic spatial characteristic of action which can be sometimes discriminative for action recognition. For example, in actions such as waving hands in different heights (Fig. 1) the trajectories of the key joint (hand) are similar. Obviously, the position feature, hand's height histograms, is more distinctive than trajectories to differentiate these two actions. In one word, the spatial location distributions of key joints in temporal domain, histogram of position and velocity, can be exploited further to enhance the spatiotemporal representation of existing trajectory-based methods.

As shown in Fig. 3, each skeleton-based action sample is a sequence of 3D pose frames. For one joint $j$ (Totally $J$ joints in all) in a sample with $F$ frames, its 3D temporal position and velocity features are formulated as the following:

$$\begin{aligned} \boldsymbol{p}_j &= [\boldsymbol{p}_{j,1}^T, \dots \boldsymbol{p}_{j,F}^T]^T \in \mathbb{R}^{3F} \\ \boldsymbol{v}_j &= [\boldsymbol{v}_{j,1}^T, \dots \boldsymbol{v}_{j,F}^T]^T \in \mathbb{R}^{3F} \end{aligned} \tag{1}$$

The joint $j$'s descriptor is denoted as: $\boldsymbol{x}_j = [\boldsymbol{p}_j^T, \boldsymbol{v}_j^T]$, and each component of $\boldsymbol{p}_j$ and $\boldsymbol{v}_j$, has three channels to denote the value of x,y,z dimensions of position and velocity, respectively. The six parts are mapped into six histograms for each joint. In Fig. 3, they are denoted as $P_z, P_x, P_y, V_z, V_x, V_y$, respectively, and the concatenation of them forms the histogram $\boldsymbol{h}_j$ of $\boldsymbol{x}_j$. All the $\boldsymbol{h}_j$ of $J$ joints are then concatenated as $\boldsymbol{H} = \{\boldsymbol{h}_j\}_{j=1}^J$ to describe the joints' position and velocity distribution in one action sequence.

### 2.2. Key joints selection and weighting

Selecting the key joints, enhancing their weights to better leverage the unequal contribution of different joints in different stages, is another contribution of our work. Here we will discuss the realization of key joints selection and weighting in our proposed method.

*2.2.1. Key Joints selection by Information Gain*

In most actions , only a few joints are responsible for the action recognition, so we propose to only preserve the most informative joints for classification. Due to information gain means the gain of information entropy brought by the introducing of one feature [15], it is also applied successfully in many machine learning algorithms, such as decision tree in [16]. In this work, we adopt information gain as the quantitative criteria to evaluate the classification contribution of each joint as following.

Information entropy of using trajectory $T$ only is calculated firstly:

$$\mathrm{Ent}(D,T) = -\sum_{k=1}^{n} p_k \log_2(p_k) \quad (p_k = \frac{TP_k}{\mathrm{m_k}}) \tag{2}$$

Here $n$ is the number of categories, $TP_k$ denotes the number of true positives for class $k$, $m_k$ means the number of samples of class $k$, and $p_k$ is to present the test data's "purity" [16].

When one joint's histogram feature $\boldsymbol{h}_j$ is introduced for classification, the entropy can be recalculated as following:

$$\mathrm{Ent}(D,T,\boldsymbol{h}_j) = -\sum_{k=1}^{n} p_k^{'} \log_2(p_k^{'}) \quad (p_k^{'} = \frac{TP_k^{'}}{\mathrm{m_k}}) \tag{3}$$

Here $TP_k^{'}$ is the new value of predicted $TP_k$ with the introducing of $\boldsymbol{h}_j$. Then, the information gain introduced by $\boldsymbol{h}_j$ can be obtained using the following equation:

$$\mathrm{Gain}(D,T,\boldsymbol{h}_j) = Ent(D,T) - Ent(D,T,\boldsymbol{h}_j) \tag{4}$$

With the introducing of $\boldsymbol{h}_j$ ($j = 1,\ldots,J$) one by one, the information gain for each joint can be obtained. Thus, with one appropriate threshold, the joints which has a large information gain (greater than the threshold) can be selected to group one compact features. The performance comparison for different key joints selection is illustrated in Table 1.

**Table 1**. Performance comparison for different key joints selection schemes

| Threshold | Key joints | Accuracy |
|---|---|---|
| $-\infty$ | 1 to 20 | 92% |
| **0.01** | **2,4,5,7,8,9,11,12,13,15,16,17,18** | **96%** |
| 0.02 | 2,4,8,9,12,16,17,18 | 94% |
| 0.03 | 2,16 | 92% |
| $\infty$ | $\emptyset$ | 88% |

*2.2.2. Spatiotemporal weighting for selected joints*

Spatiotemporal weighting is widely used in sequence learning which can better leverage the unequal contribution of different joints in different stages for action recognition [2]. Similarly, here a method of spatiotemporal weighting for skeleton-based action recognition is established. For the convenience of computing and expression, each action sequence with different frames $F$ is divided into $S$ stages averagely (each stages has $\lceil F/S \rceil$ frames). Since each joint may have one different weight in each stage, one weight matrix for all joints in all stages can be formulated as:

$$\begin{pmatrix} w_{1,1} & \cdots & w_{1,S} \\ \vdots & \ddots & \vdots \\ w_{J,1} & \cdots & w_{J,S} \end{pmatrix} \tag{5}$$

Here $S$, $J$ denotes the total number of stages and joints, respectively, and $w_{j,s}$ is the weight for joint $j$ in stage $s$. Due to the scale-invariant feature of Mahalanobis distance, it is adopted to calculate the distance between two patterns, and then this distance is used as the dependent variable to obtain the weights. Here we give the calculating procedure for the pattern of velocity as one sample. Firstly, the velocity matrix for sample D is given as:

$$\begin{pmatrix} \overline{\boldsymbol{v}}_{1,1} & \cdots & \overline{\boldsymbol{v}}_{1,S} \\ \vdots & \ddots & \vdots \\ \overline{\boldsymbol{v}}_{J,1} & \cdots & \overline{\boldsymbol{v}}_{J,S} \end{pmatrix} \tag{6}$$

It is noteworthy that each element $\overline{\boldsymbol{v}}_{j,s}$ here is the average velocity of all $N$ frames in stage $s$ of joint $j$:

$$\overline{\boldsymbol{v}}_{j,s} = \frac{1}{N}\sum_{f=1}^{N} \boldsymbol{v}_{j,s,f} \tag{7}$$

From velocity matrix we will obtain 1) the velocity of the stage $s$, joint $j$ ($\overline{\boldsymbol{v}}_{j,s}$), 2) the velocity set of all joints in the stage $s$ ($V_s = \{\overline{\boldsymbol{v}}_{1,s},\ldots,\overline{\boldsymbol{v}}_{J,s}\}$), and 3) the velocity set of joint $j$ in all stages ($V_j = \{\overline{\boldsymbol{v}}_{j,1},\ldots,\overline{\boldsymbol{v}}_{j,S}\}$) likewise. If we define the expectation of $V_s$,$V_j$ as $\mu_s$,$\mu_j$ and the covariance as $\sigma_s$,$\sigma_j$, the Mahalanobis distance $d_s^M$, $d_j^M$ can be calculated as:

$$\begin{aligned} d_s^M &= \sqrt{(\overline{\boldsymbol{v}}_{j,s}-\mu_s)^T \sigma_s^{-1} (\overline{\boldsymbol{v}}_{j,s}-\mu_s)} \\ d_j^M &= \sqrt{(\overline{\boldsymbol{v}}_{j,s}-\mu_j)^T \sigma_j^{-1} (\overline{\boldsymbol{v}}_{j,s}-\mu_j)} \end{aligned} \tag{8}$$

Considering the two distances play same effect to measure the difference of velocity patterns, the following equation is taken as the weighting function, where $d_s^M$ multiply $d_j^M$ is taken as the dependent variable, and C is taken as a constant bias.

$$w_{j,s} = \begin{cases} \log(d_s^M \cdot d_j^M) + C \ (w_{j,s} > 1) \\ 1 \ (w_{j,s} \le 1) \end{cases} \tag{9}$$

The max-min scaling is adopted for weight normalization.

For the weighting operation using weight matrix, it can be implemented by multiplying the vote numbers with $w_{j,s}$ for each joint in each stage.

An Example of Spatiotemporal Weight Matrix for ten different actions is shown in Fig. 4. From Fig. 4, it can be found that the weights of different joints and stages vary largely for different actions, while for the samples with same category, the weight are of high similarity. That's to say, the spatiotemporal weighting for the joints is necessary.

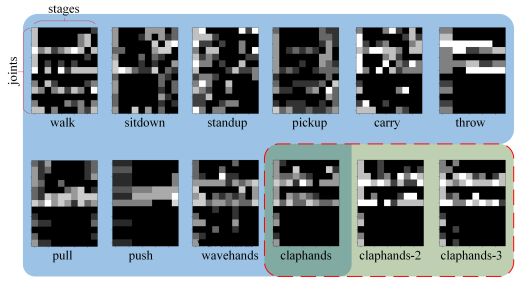


**Fig. 4**. An Example of Spatiotemporal Weight Matrix for ten different actions. Each matrix is 13 (selected 13 key joints) by 10 (stages), and the higher luminance means the larger weight. The weights of different joints and stages vary largely for different actions, while for the samples with same category (the right-bottom three matrices, which are marked by red box), the weight are of high similarity.

### 2.3. Fisher Vector Encoding

Fisher vector coding [17] is a great help to better describe the skeleton trajectory with a richer combination of features for enhancing the dimensions of its features. Before encoding, we need to declare two parameters T and K. The T descriptors are defined as X = $\{x_t$, t = 1 ... T$\}$, which are T joints histogram selected by information gain. The parameter of Gaussian distribution number K is adjustable. The best value given by our experiment is K=T. The reason might be that T is the number of key joints, and these joints's feature fulfill the i.i.d. conditions to some extent. The fisher vector encoding can improve the overall performance for about 1%.

## 3. EXPERIMENT

The experiments of our proposed method on two widely-tested benchmark datasets (UTKinect Action dataset [18] and MSRAction3D dataset [19]) are shown in Table 2 and Table 3. Three baseline methods are included in the comparison based on the UTKinect Action dataset in Table 2. It shows that position & velocity histograms do help improve the performance of our method. As we can see, the performance is also improved by 2% after key joints selection and spatial-temporal weighting is adopted. In addition, there is a slight improvement by encoding the concatenation of histogram and trajectory into fisher vector.

Compared with existing results using sophisticated models such as deep learning [6, 13], it shows that by combining the strength of both richer features and key joints selecting, our method can achieve state-of-the-art or competitive performance with advantages regarding the recognition accuracy and robustness. In MSRAction3D dataset, the proposed method outperforms all existing schemes [2, 6, 13, 20, 21, 22]; It also gets nearly the state-of-the-art performance (98.0 vs.98.2) in UTKinect Action dataset. Although SCK+DCK [21] gets higher accuracy than proposed method in UTKinect

**Table 2**. Comparison of Results on UTKinect (%)

| method | accuracy |
|---|---|
| Trajectory(traj) | 91.00% |
| traj+H | 95.00% |
| traj+H+key joints & weighting | 97.00% |
| **H+traj+key joints & weighting+FV** | **98.00%** |
| ST-LSTM[13] | 97.00% |
| Lie Group[20] | 97.10% |
| Graph-based[21] | 97.40% |
| ST-NBNN[2] | 98.00% |
| SCK+DCK [22] | **98.20%** |
| Ours | 98.00% |

**Table 3**. Comparison of Results on MSRAction3D (%)

| method | AS1 | AS2 | AS3 | Average |
|---|---|---|---|---|
| Lie Group [20] | **0.954** | 0.839 | **0.982** | 0.925 |
| SCK+DCK[22] | - | - | - | 0.94 |
| HBRNN[6] | 0.933 | 0.946 | 0.955 | 0.945 |
| ST-LSTM[13] | - | - | - | 0.948 |
| GRAPH-Based[21] | 0.936 | 0.955 | 0.951 | 0.948 |
| ST-NBNN[2] | 0.915 | 0.956 | 0.973 | 0.948 |
| Ours | 0.945 | **0.96** | 0.973 | **0.959** |

Action dataset, however, its average performance drops largely in MSRAction3D dataset. In other words, the proposed method achieves the best robustness compared to the existing algorithms [2, 6, 13, 20, 21, 22]. With the observation and experiments result, we can give one reasonable explain why the proposed method perform better in MSRAction3D is that: the samples in MSRAction3D have more inter-class difference in key joints' position changes, and thus the proposed framework, which combined position histogram and trajectory features, can get higher recognition accuracy.

## 4. CONCLUSION

In this paper, a novel skeleton-based action recognition method is proposed, in which position and velocity histograms are developed and concatenated with traditional trajectory feature to enhance the representation ability for 3D actions. Key joints selection and spatiotemporal weighting is also utilized to increase the performance further. In one word, our proposed method can discover critical joints and their spatiotemporal patterns for skeleton-based action recognition. Despite using only a non-deep-learning model, the proposed method works surprisingly well on two benchmark datasets and achieves competitive results compared with state-of-the-arts using sophisticated end-to-end networks.

## 5. ACKNOWLEDGEMENT

This work was partly funded by the National Key Research and Development Program (2017YFB1002401), NSFC(No.61571297,61521062, No.61420106008), and STCSM(18DZ2270700).

## 6. REFERENCES


[1] A. Aly Halim, C. Dartigues-Pallez, F. Precioso, M. Riveill, A. Benslimane, and S. Ghoneim, "Human action recognition based on 3d skeleton part-based pose estimation and temporal multi-resolution analysis," in *International Conference on Image Proceeding(ICIP)*. IEEE, 2016, pp. 3041–3045.

[2] J. Weng, C. Weng, and J. Yuan, "Spatio-temporal naive-bayes nearest-neighbor for skeleton-based action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017, vol. 00, pp. 445–454.

[3] W.-Y Lin, K. Lu, B. Sheng, J.-X Wu, B.-B Ni, X. Liu, and H.-K Xiong, "Action recognition with coarse-to-fine deep feature integration and asynchronous fusion," in *AAAI Conference on Artificial Intelligence*, 2018.

[4] L.-M Wang, Y. Qiao, and X.-O Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Internaltional Conference on Computer Vision and Pattern Recogintion*. IEEE, 2015, pp. 4305–4314.

[5] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[6] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1119–1118.

[7] G. Chéron, I. Laptev, and C. Schmidy, "P-cnn: Pose-based cnn features for action recognition," in *International Comference on Computer Vision*. IEEE, 2015, pp. 3218–3226.

[8] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (smij): A new representation for human skeletal action recognition," in *Journal of Visual Communication & Image Representation*, 2014, vol. 25(1), pp. 24–38.

[9] X.-D Yang and Y.-L Tian, "Super normal vector for activity recognition using depth sequences," in *Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 804–811.

[10] S. Sempena, N. U. Maulidevi, and P. R. Aryan, "Human action recognition using dynamic time warping," in *International Conference on Electrical Engineering and Informatics*. IEEE, 2011, pp. 1–5.

[11] L.-M Wang, Y.-J Xiong, Z. Wang, Y. Qiao, D.-H Lin, X.-O Tang, and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Acm Transactions on Information Systems*, 2016, vol. 22(1), pp. 20–36.

[12] Y. Tian, L. Jing, Y. Ye, X. Yang, and Y. Tian, "3d convolutional neural network with multi-model framework for action recognition," in *International Conference on Image Proceeding(ICIP)*. IEEE, 2017.

[13] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal lstm with trust gates for 3d human action recognition," in *European Conference on Computer Vision*. Springer, 2016, pp. 816–833.

[14] I.C. Duta, B. Ionescu, K. Aizawa, and N. Sebe, "Spatio-temporal vector of locally max pooled features for action recognition in videos," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2017, pp. 3205–3214.

[15] E.T. Jaynes, "Information theory and statistical mechanics ii," in *Physical Review*, 1957, vol. 108, pp. 171–190.

[16] J.R. Quinlan, "C4.5: Programs for machine learning," in *Morgan Kaufmann*, 1993.

[17] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector theory and practice," in *International Journal of Computer Vision*, Dec 2013, vol. 105, pp. 222–245.

[18] L. Xia, C.-C Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2012, pp. 20–27.

[19] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern RecognitionWorkshops*. IEEE, 2010, pp. 9–14.

[20] R. Vemulapalli, F. Arrate, and R. Chellappa, "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*. IEEE, 2014, pp. 588–595.

[21] P. Wang, C. Yuan, W. Hu, B. Li, and Y. Zhang, "Graph based skeleton motion representation and similarity measurement for action recognition," in *European Conference on Computer Vision*. Springer, 2016, pp. 370–385.

[22] P. Koniusz, A. Cherian, and F. Porikli, "representations via kernel linearization for action recognition from 3d," in *arXiv preprint arXiv:1604.00239*, 2016.