Joint Motion Similarity (JMS)-Based Human Action Recognition using Kinect

Jiawei Li, Jianxin Chen, Linhui Sun
Nanjing University of Posts and Telecommunications
Nanjing, China
Email: lijiaweiny@icloud.com; chenjx@njupt.edu.cn; sunlh@njupt.edu.cn

***Abstract*—Human action recognition has been a research topic due to challenges such as viewpoint variation and self-occlusion. Recently the cost-effective 3D sensor like Kinect makes it possible to combat these problems. In this paper, we propose a human action recognition method according to the similarity property of the joint motion, divide the human body into several clusters, and use feature extraction and classification method for each clusters. Then each class is weighted and the maximal weighted distance is obtained for action recognition. Experimental results show that the proposed approach not only achieves better action recognition accuracy than current methods, but also significantly reduces the computation cost.**

I. INTRODUCTION

Human action recognition has a wide of applications, such as surveillance systems, security, video games and robotics. There are two procedures for the action recognition. One is extracting features from the video sequence, while the other is learning and recognizing actions from the action sequences. But in these procedures, due to the intra-class variations, the differences in viewpoints, the self-occlusion and the variant rate of actions, there are still challenges for human action recognition.

Human action recognition began in the early of 1980s, and the previous research is mainly based on the traditional RGB video camera [1]. However, the RGB video camera is sensitive to the changing of brightness, viewpoints, and self-occlusion. These disadvantages result in the motion captured from RGB video cameras losing a large volume of information, which decreases the action recognition accuracy.

Recently, with the introduction of cost-effective depth sensors such as Mircosoft Kinect and ASUS Xtion PRO, the depth images can be obtained much easier than before. The location of each joint in 3D (dimension) space could be obtained from these depth images, which is helpful to overcome the above challenges for action recognition [2]. However, there are still some problems such as intra-class variations in the motion. For example, when someone is waving, the action of waving is judged mainly depending on the motion of arms. While the movement of parts such as feet or head are insignificant during the action of waving. It is the same with other actions. These factors indicate that if the training samples include the whole parts of body, some insignificant parts making no contribution to action recognition will decrease the accuracy of the action recognition.

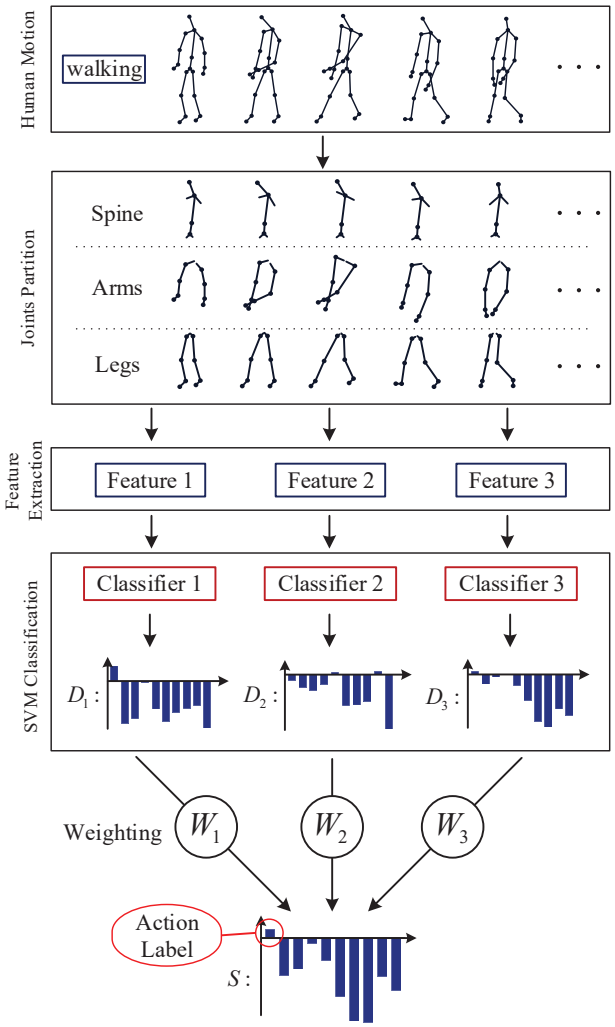


Fig. 1. Sketch map of the proposed approach. A instance of action *walking* is shown at the top of the figure, from the UTKinect dataset. As in section III, the human body is divided into $M = 3$ clusters. Each clusters is extracted feature independly. The distance between sample point and hyperplane is $D_m \in R^{1\times 10}$ is computed, and added with weight W_m. As the action *walk* has the largest distance in the S (it is highlighted by the red circle), the action is recognized as *walk*.

978-1-5090-2896-2/16/$31.00 ©2016 IEEE

In this paper, by analyzing the joint motion similarity in 3D space during action, we propose a new action recognition method as in Fig. 1 for the action "walking", which might improve the action recognition accuracy while keep low computation cost.

Our contributions consist of three aspects:

1) We analyze the joint motion during action using the Kinect sensor, and found their similarity feature.

2) Based on the similarity feature of joint motion, we propose a new action recognition approach by dividing the body joints into several clusters. Each cluster is extracted feature and classified independently, and the effect of insignificant parts are largely eliminated by the process of weighting.

3) Plenty of experiments have been performed over three datasets to verify the efficiency of our approach in terms of recognition accuracy and computation cost.

The remainder of this paper is organized as follows. In section II, we provide a brief review of the related work. Section III discusses the representation of the motion pattern, and the joint motion similarity feature. Section IV propose our action recognition approach by dividing the human body into several clusters according to the feature of joint motion similarity. Experimental results are given to verify the efficiency of the proposed approach in section V. And Section VI draws the conclusion.

II. RELATED WORK

There have been some approaches for human action recognition in the past few years. Generally for action recognition, tracking 3D location of body joint can obtain better performance. Previously, a particular motion capture system has been used to track the 3D positions of labels attached to the human body, and such equipment is expensive [4]. Recently, the introduction of cost-effective depth sensors like Kinect makes it easier to obtain the 3D location of joints. But using such sensors for human action recognition, some challenges are appearing due to the viewpoint variations, intra-class variations and inter-class similarity.

The viewpoint problem is that for the same action the viewpoints might be different. To combat this problem, the joint relation or joint angle is used to represent the skeleton. In [16], skeletons were rotated to keep the view invariant. In [11], some joints were chosen as the most informative joints, and the body was represented with these joints. As the variation of joint angle are used, the viewpoint changing does not affect the recognition accuracy. In [22], Georgios used the skeletal quad as the descriptor which encoded the geometric relation of joint quadruples. The rotation around local coordinate was normalized so that it was view-invariant.

The second challenge is to overcome the intra-class variations. As individuals perform the same action in different motion patterns, it will be hard to recognize two instances of the same action with different motion patterns. In [5], Jiang Wang et. al. proposed an action let ensemble model to describe the interaction of joints, and they used Fourier Temporal Pyramid to remove the noise of depth data. In [9], Vemulapalli et. al. modeled the 3D geometric relationship between various body parts with a special Euclidean group $SE(3)$ and the skeleton was represented in Lie group as $SE(3) \times ... \times SE(3)$. In [21], Xiaodong Yang et. al. described the difference of each joint in temporal and spatial domains, and applied PCA to find the Eigen joints. In [7], Wang et. al. grouped the estimated joints into five body parts to represent the human body, which is robust to the joint estimation and intra-class variations. In [6], Mikel et. al. proposed an approach based on a Maximum Average Correlation Height filter, which might cancel intra-class variation by synthesizing a single Action MACH filter for a given action class. However, most of these works concentrate on representation methods to eliminate the interference of intra-class variations. In our work, we attempt to improve the accuracy of action recognition by dividing the human body into clusters according to the similarity of joint motion, while decreasing the computation cost. In the next section, we will analyze the joint motion similarity.

III. MOTION PATTERN REPRESENTATION

To represent the motion pattern of each joint, here we use joint angles between two adjoining bones. Fig. 2 depicts the human skeleton and related joint angles. In this model, there are twenty joints, nineteen body parts and eighteen joint angles. These joint angles are named as those in Fig. 2.

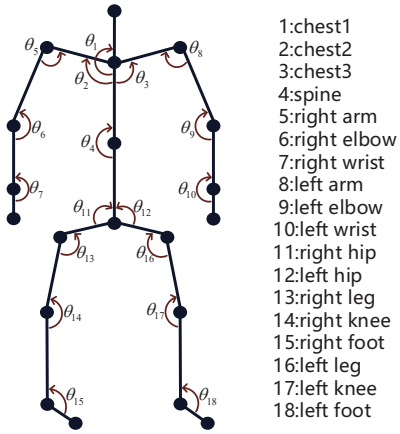


Fig. 2. Human Skeleton and Joint Angles

A. DTW Distance and Joint Variance

DTW has proven effective to achieve the optimal alignment between two time-dependent sequences. Assume $X = (x_1, x_2, \cdots, x_N)$ and $Y = (y_1, y_2, \cdots, y_M)$ to be two angle sequences, and N and M denote their lengths. The cost matrix $C \in \mathbb{R}^{N \times M}$ for them can be obtained according to their Euclidean distance.

From the cost matrix, we define the accumulated cost matrix O as

$$\begin{cases} O(n,1) = \sum_{k=1}^{n} C(k,1) \\ O(1,m) = \sum_{k=1}^{m} C(1,k) \\ O(n,m) = \min\{O(n-1,m-1), O(n-1,m), \\ \qquad\qquad O(n,m-1)\} + c(x_n, y_m) \end{cases}, \quad (1)$$

where $1 \leq n \leq N$ and $1 \leq m \leq M$. Then, the optimal warping path $P = (p_1, \cdots, p_L)$ is

$$p_{l-1} = \begin{cases} (1, m-1), & if\ n = 1 \\ (n-1, m), & if\ m = 1 \\ argmin\{O(n-1, m-1), & \\ O(n-1, m), O(n, m-1)\}, & otherwise \end{cases} . \quad (2)$$

Here $p_l = (n, m)$ is the element in the optimal warping path and L is the path length. The distance can be achieved by summing elements of optimal alignment P in the cost matrix C as follows

$$d_n^r = \sum_{i=1}^{T} \sum_{j=1}^{L} C(p_j). \quad (3)$$

Here n is denotes the action, r denotes the related joint, and T is the number of action instance.

For an action instance, let the angle sequence $\Theta = \{\theta_1, \theta_2, \cdots, \theta_T\}$, then the angle variance is

$$\sigma_n^r = \sum_{i=1}^{T} Var(\Theta_i). \quad (4)$$

B. Joint Motion Similarity

According to the definitions of variance matrix and DTW distance matrix, Fig. 3a and Fig. 3b depict them on the UTKinect dataset [16]. The motion pattern of ten actions is represented with 14 joint angles. The other four joints, like *right wrist, left wrist, right foot and left foot*(with the index of 7, 10, 15, 18) are omitted here due to the measuring error from the Kinect sensor.

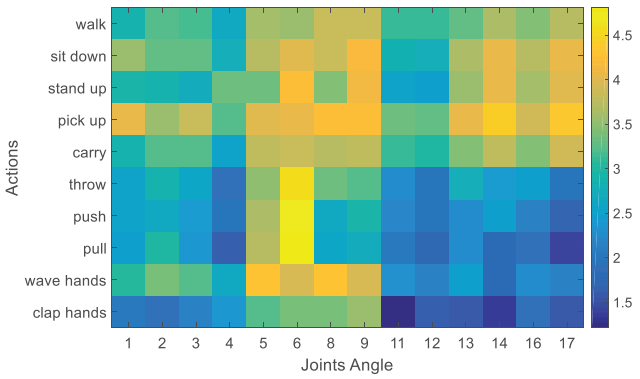
From them, we can find that some joints have similar motion patterns. For example, the joint angle *right arm*(index 5 in horizontal axis) and *right elbow* (index 6 in horizontal axis) have similar motion patterns for actions such as walk, pick up, carry, and claps. While for *left arm* (index 8 in horizontal axis) and *left elbow* (index 9 in horizontal axis), the motion patterns are similar under the actions such as pull, push, pick up and walk. Right hip and left hip nearly have the similar motion pattern under most of actions. Especially, the motion patterns are substantially similar for the joints of lower limb (index 11-17 in horizontal axis) under all actions. But for *right wrist, left wrist, right foot and left foot*, motion patterns have weakly similar under most actions.

In next section, we will design an action recognition scheme using such feature of joint motion similarity during actions.

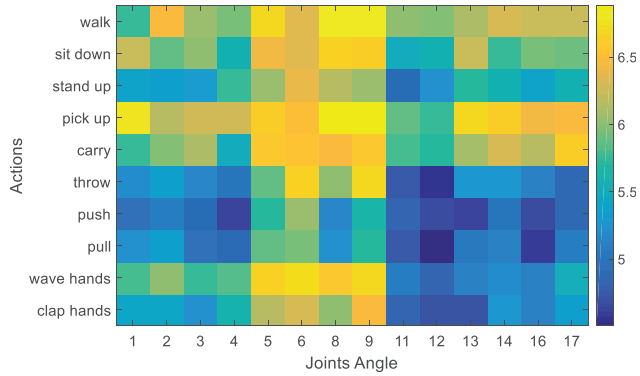
IV. Algorithm Description

A. Overview of Algorithm

The proposed algorithm is based on the joint motion similarity (JMS). Fig. 1 depicts the action recognition procedure for the instance of walking. In this algorithm, all joints in human body are divided into several clusters, and the joints belonging to the same cluster have similar motion pattern as we discussed in above section. Then for each cluster, features are extracted and mapped into Lie group respectively. DTW and Fourier Temporal Pyramid representation are used to deal



(a)



(b)

Fig. 3. (a) Variance Matrix ; (b) DTW Distance Matrix

with the rate variation and combat the high frequency noise. Then each cluster is classified by one-vs-all SVM. The output class is weighted and the maximal weighted distance is chosen for action recognition.

B. Feature Extraction

Following [9], human skeletons in 3D space are represented as points in a Lie group [15]. Assume a Euclidean group denoted by a 4 by 4 matrix as following

$$\begin{bmatrix} R & \vec{d} \\ 1 & 1 \end{bmatrix} \in SE(3), \quad (5)$$

where $R \in \mathcal{R}^{3\times3}$ is a rotation matrix, and $\vec{d} \in \mathcal{R}^3$ is a position vector. Let the group $SE(3)$, and its associated Lie algebra is $\mathfrak{se}(3)$, which is the tangent plane to $SE(3)$ at identity element I_4. The element in $\mathfrak{se}(3)$ is a vector in $\mathcal{R}^6$. Define the logarithm map from $SE(3)$ to $\mathfrak{se}(3)$ as

$$\log : SE(3) \rightarrow \mathfrak{se}(3). \quad (6)$$

Fig. 2 depicts a body skeleton with 20 joints and 19 body parts. It is possible to divide into several clusters according to the joints or parts. Then we will extract feature from each cluster. Assume s_m and s_n be two parts in one cluster, and their starting and end points are $s_{m1}, s_{n1} \in R^3$ and $s_{m2}, s_{n2} \in$

R^3 in the local coordinate. According to above definitions, we have

$$\begin{bmatrix} s^n_{m1,t} & s^n_{m2,t} \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} R_{mn,t} & \vec{d_{mn,t}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & l_m \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}, \quad (7)$$

where l_m is the length of skeleton s_m. $R_{mn,t}$ and $d_{mn,t}$ are the rotation matrix and translation matrix at time t. Features between s_m and s_n are represented as

$$g_{n,m}(t) = \begin{bmatrix} R_{nm,t} & \vec{d_{nm,t}} \\ 0 & 1 \end{bmatrix} \in SE(3), \quad (8)$$

$$g_{m,n}(t) = \begin{bmatrix} R_{mn,t} & \vec{d_{mn,t}} \\ 0 & 1 \end{bmatrix} \in SE(3). \quad (9)$$

By direct product, the skeleton S at time t can be obtained as

$$S(t) = [g_{1,2}(t) \times g_{2,1}(t) \times \cdots \times g_{N-1,N}(t) \times g_{N,N-1}(t)], \quad (10)$$

where N is the number of parts in one cluster. As the data belonging to $SE(3)$ space cannot be used for the common classifier such as SVM, we need to map $SE(3) \times \cdots \times SE(3)$ to $\mathfrak{se}(3) \times \cdots \times \mathfrak{se}(3)$. Then features in Euclidean Group are mapped to Lie algebra by

$$\begin{aligned} F(t) = [&\log(g_{1,2}(t)) \times \log(g_{2,1}(t)) \times \cdots \\ &\times \log(g_{N-1,N}(t)) \times \log(g_{N,N-1}(t))], \end{aligned} \quad (11)$$

which is a vector and will be used for classification.

C. Action Classification

As each instance has different frames, we use DTW to normalize action sequences [12]. Let the first instance of each action as the standard sequence, the other instances are warped to have the same length as that of the standard sequence. Since the depth map from Kinect sensor is easily affected by the noise, the representation of Fourier Temporal Pyramid [13] is used here.

After that we perform action classification by one-vs-all SVM [14] for each cluster respectively. One-vs-all SVM classifier is the combination of N binary classifiers, where N is the number of actions. The output of one-vs-all SVM is a distance vector from the hyperplane, which is denotes as $d = (d_1, .., d_N)$. Then the classification result is

$$\gamma = \arg\max_{1 \le n \le N} d_n. \quad (12)$$

In our approach, if the body is divided into M clusters, the feature of human body can be denoted as a distance matrix as

$$D = \begin{bmatrix} d_{1,1} & \cdots & d_{1,N} \\ \vdots & \ddots & \vdots \\ d_{M,1} & \cdots & d_{M,N} \end{bmatrix}. \quad (13)$$

Now for this distance matrix, it is necessary to add weight before action recognition.

D. Weight Computation

For the test action i, the distance vector of class n is D_n, which is denoted $D_n = (d_{1,n}, \cdots, d_{M,n})$. If i is equal to n, it is denoted as positive sample. Otherwise, it denotes the negative sample.

To compute the weight, here we choose the logistic regression method. Define a sigmoid function as

$$\sigma(z) = \frac{1}{1 + e^{-z}}, \quad (14)$$

where $z = w_0 + w_1 x_1 + \cdots + w_M x_M$ and M is the number of classes. And a cost function is defined by

$$f(w) = \frac{1}{T} \sum_{i=1}^{T} [-y_i \log(\sigma(z_i)) - (1 - y_i) \log(1 - \sigma(z_i))], \quad (15)$$

where T is the number of instances. Then the cost function gradient is

$$\nabla f(w) = \frac{1}{T} \sum_{i=1}^{T} (\sigma(z_i) - y_i) x_i. \quad (16)$$

Algorithm 1 describes the weight computation procedure. In this algorithm, the weight w is updated by $w = w + \lambda \nabla f(w)$ during each iteration. Here λ is the step length. The algorithm works till the gradient is less than the tolerance or the iteration times approximate to the threshold.

Algorithm 1 Weight Computation

Require: Cost function $f(w)$, gradient function $\nabla f(w)$, maximum number of iterations $maxiter$, tolerance ε.
Ensure: Weight matrix w
  Initialization: $w = [0 \ \cdots \ 0], k = 0$
  while $k < maxiter$ **do**
    Compute gradient $\nabla f(w)$;
    if $\nabla f(w) < \varepsilon$ **then**
      break
    end if
    Find λ which minimize $f(w + \lambda \nabla f(w))$
    $w = w + \lambda \nabla f(w)$;
    $k = k + 1$;
  end while
  return w

Fig. 4a depicts the distribution of one instance of the action *walking* on the UTKinect dataset. It has three dimensions as the body is divided into three classes. According to such definition, Fig. 4b depicts the distribution of weighted data.

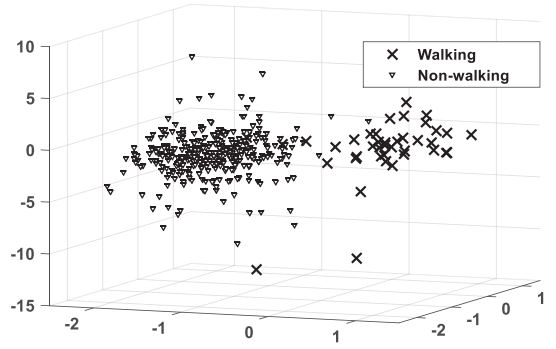
E. Action Recognition

After we obtain the distance matrix $D \in \mathcal{R}^{M \times N}$ and the weight $W \in \mathcal{R}^{N \times M}$, the weighted distance S is denoted as

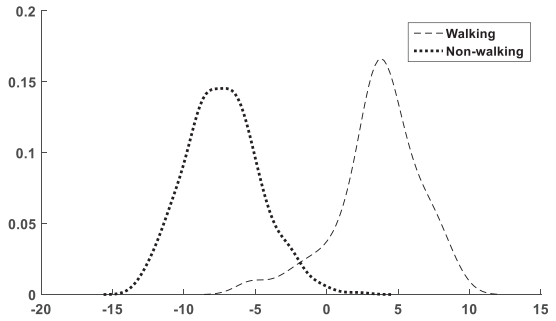
$$S_n = \sum_{i=1}^{M} d_{n,i} w_{n,i}. \quad (17)$$

Then we have the label for action recognition

$$label = \arg\max_{1 \le n \le N} S_n. \quad (18)$$



(a) original distribution



(b) weighted distribution

Fig. 4. (a) The original distribution of the action *walking*, on UTKinect dataset. These points have three dimension; (b) The weighted one-dimension points.

V. PERFORMANCE ANALYSIS

A. Dataset

To show the performance of our proposed approach, three datasets are used: UTKinect [16] dataset, MSR Actions3D [20] dataset, and the dataset collected by ourselves.

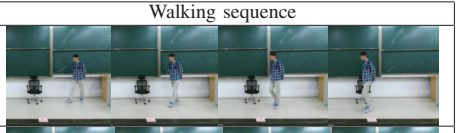
(1) UTKinect dataset [16]: This dataset was built using one Kinect, and ten subjects are tested. Each subject performs ten actions twice such as *walk, sit down, stand up, pick up, carry, throw, push, pull, wave hands and clap hands*. In this dataset, there are 199 action sequences. For each action, the locations of 20 joints are recorded combined with the RGB and depth information. The frame rate is set 30 Hz.

(2) MSR-Action3D dataset [20]: In this dataset, the video was recorded with a depth sensor like Kinect. Ten subjects are tested and each subject performs 20 actions two or three times, e.g. *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side-boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pick up and throw*. In this dataset there are 557 samples.

(3) The dataset collected by ourselves: In this dataset, ten subjects are captured and each subject performs ten indoor actions twice, e.g. *walk, pick up, read book, sit down, stand up, jump, clap, make phone call, throw and drink*. The locations of joints in 3D space are used for action recognition.

As in TABLE. I, we perform the same actions with different motion patterns. For example, someone walks with arms sagging naturally, while someone walks with hands crossing on the chest. In addition, action sequences have different frames. These constraints increase the difficulty of action recognition. Then it is possible to highlight the capability of our approach to combat the problem of intra-class variations.

TABLE I
SAMPLE IMAGES OF FOUR SUBJECTS PERFORMING THE ACTION WALKING

Subject No.	Walking sequence
Subject A	
Subject B	
Subject C	
Subject D	

B. Experimental Results

Here we will evaluate the performance of our approach by dividing the human body into different clusters and comparing it with the state-of-the-art action recognition methods. In our experiments, a half of the dataset is used to train the classifier. One tenth is used to train the weight, and two fifth is used to evaluate the performance. When computing the weight, we set the iteration threshold as 400. As MSR3D dataset has two experimental setting, for the first setting, we follow the method of [20] and divide the 20 actions into three subsets (AS1, AS2, AS3), each consisting of 8 actions. For the second setting, the classification are perform over the whole dataset. We perform our experiments on a computer with intel i7 CPU of 3.50 GHz. The recognition accuracy is average of all actions in the dataset. The elapsed time is the computation cost for the feature extraction.

1) Different Division Schemes: According to analysis in III, here we compare different division fashions by dividing human joints into different clusters, e.g. one cluster, two clusters (JMS-2), three clusters (JMS-3), and five clusters (JMS-5). In each cluster, the joints have similar motion patterns. The detailed division fashions are:

a) One cluster [9]: the entire body;

b) Two clusters (JMS-2): *upper body* and *legs*;

c) Three clusters (JMS-3): *arms*, *legs* and *spine*;

d) Five clusters (JMS-5): *left arm*, *right arm*, *left leg*, *right leg* and *spine*.

Table II and Table III list the performance of our approach over three datasets. In these tables, the division of one cluster

TABLE II
RECOGNITION RATE ON UTKINECT AND OURS DATASET

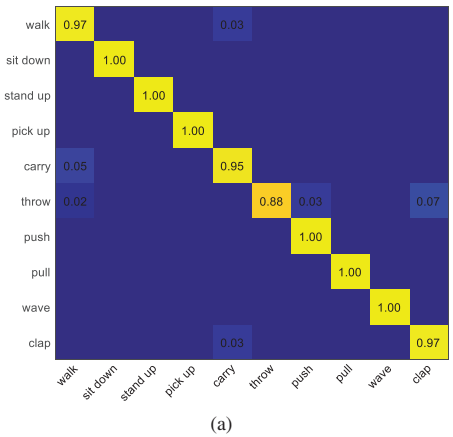
UTKinect dataset [16]				
Division scheme	one cluster	JMS-2	JMS-3	JMS-5
Recognition rate	97.19%	97.67%	97.65%	94.47%
Elapsed time / second	4533	2347	1618	1022
Our dataset				
Division scheme	one cluster	JMS-2	JMS-3	JMS-5
Recognition rate	87.71%	91.69%	91.85%	88.44%
Elapsed time / second	5260	2729	1690	1176

TABLE III
RECOGNITION RATE ON MSR3D DATASET

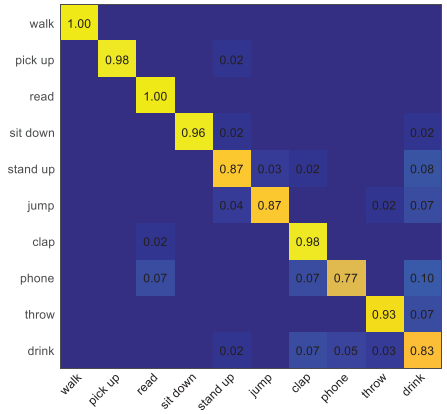
Setting one				
Dataset	one cluster	JMS-2	JMS-3	JMS-5
AS1	94.34%	94.96%	94.94%	88.67%
AS2	83.87%	87.90%	74.85%	68.84%
AS3	97.43%	98.90%	97.94%	93.42%
Average	91.88%	93.92%	89.24%	83.64%
Time	22850	11056	7306	4839
Setting two				
Partition scheme	One cluster	JMS-2	JMS-3	JMS-5
Recognition rate	89.48%	89.48%	89.94	86.64

corresponds to the approach in [9]. From these results, we can find that for the UTKinect dataset, JMS-2 works a little better than one cluster in term of recognition accuracy. But for the computation complexity, the cost of JMS-2 is nearly half of one cluster. When we divide into three clusters (JSM-3), the computation cost is nearly one third of one cluster, and the recognition accuracy is a little better. But when five clusters are divided, the recognition accuracy also decreases. However, the computation cost decreases significantly. On the dataset collected by ourself, our approach outperforms that in [9], not only in terms of recognition accuracy, but also the computation cost. Even for the case of five clusters (JMS-5), the recognition accuracy is a little better.

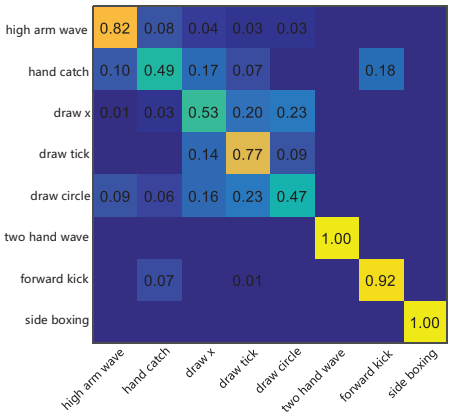
Table III lists the experimental results on MSR3D dataset. For the first setting, JSM-2 achieves better recognition accuracy on all three subsets AS(1), AS(2) and AS(3), while the computation cost reduces half. But when we divide the body joints into three clusters (JSM-3), the average recognition accuracy is a little worse than that of one cluster. This is due to the low recognition accuracy on the subset AS(2). But the computation cost reduces significantly. For the second setting, the recognition accuracy of JMS-2 is equal to that of one cluster. While JMS-3 works a little better than one cluster, and only costs one third time. Since the first setting and second setting have the same procedure of extracting features, they have the same computation cost. Therein we do not compare here.



(a)



(b)



(c)

Fig. 5. Confusion matrices: (a) UTKinect dataset; (b) Ours dataset; (c)MSR3D dataset AS1

From the above results, we note that the computation cost decreases significantly when the body joints are divided into more clusters. This is mainly due to the reducing of the feature dimension. For instance with UTKinect dataset, the skeletal representation contains 342 pairs of relative features in [9]. But in our approach, the features decrease rapidly. For example,

TABLE IV
COMPARISON WITH THE STATE-OF-ART RESULTS

A: UTKinect dataset [16]	
Histograms of 3D joints [16]	90.92%
Random forests [17]	87.9%
Relative pairs in lie group [9]	97.08%
Histogram of direction vectors [19]	91.96%
JMS-2	**97.67%**
JMS-3	97.65%
JMS-5	94.47%
B: MSR3D dataset (setting one) [20]	
Histograms of 3D joints [16]	78.97%
EigenJoints [21]	83.8%
Relative pairs in lie group [9]	92.46%
Joint angles similar and HOG2 [18]	**94.84%**
JMS-2	93.92%
JMS-3	89.24%
JMS-5	83.64%
C: MSR3D dataset (setting two) [20]	
Actionlet [5]	88.2%
JMS-2	89.48%
JMS-3	**89.94%**
JMS-5	86.64%

JMS-2, JMS-3, and JMS-5, have 166, 102 and 54 pairs of relative feature respectively. Therein it is possible to reduce the computation cost significantly with more clusters being divided.

From the results, we found that the performance of our approach is better than the method in [9]. This is due to that the interference of insignificant body part is eliminated by the weight. Fig. 5a depicts the confusion matrix of JMS-3 scheme for UTKinect dataset. From it, we can find that most of actions have good recognition rate, such as *sit down, stand up, and wave*. The insignificant parts do not interfere the classification. The action *walk* and *carry* are partly confused since these motions are similar. The action of *carrying* contains the motion of *walking*. For the action of *throw*, our approach performs weakly because this action has rather different motion pattern for each subject.

Fig. 5b depicts the confusion matrix of JMS-3 on our dataset. Some actions have a relatively poor recognition accuracy, such as *jump, make phone call and drink*. We note that some joints are too close while the Kinect sensor cannot overcome the problem of self-occlusion. In addition, these actions are more complicate and have a relatively different motion pattern from other actions. Fig. 5c depicts the confusion matrix of JMS-3 on the second subset of MSR3D dataset. We can find that the misclassification mainly occurs over the highly similar actions such as *hand catch, draw X, draw tick, draw circle* which only contain hands in the motions.

2) Comparing with other Methods: Table. IV lists the results of various human action recognition algorithms based on the 3D joint location over the UTKinect dataset and MSR3D dataset. From it, we note that our approach works well on these datasets and achieves the best results on the UTKinect and MSR3D dataset with the second setting. We have already compared the computation cost with the method in [9] in section V-B1. And results indicate that our method works much better while achieving the similar recognition accuracy. But for the method in [18], it works a little better than our approach as it uses more features to realise the classification.

VI. CONCLUSION

In this paper, we propose an action recognition approach by dividing body joints into clusters according to the feature of motion similarity. The joints belonging to the same cluster have the similar motion. The feature extraction is performed on each cluster. After that, the action of each cluster is classified, and before action recognition a logistic regression is used to compute the weight for each class to combat the interference of insignificant part. The approach is evaluated over three motion datasets including the one built by ourselves, which contains intra-class variations and self-occlusions from the Kinect sensor. Experimental results show that our approach can achieve better action recognition accuracy with rather lower computation complexity compared with the state-of-the-art action recognition approaches.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (No. 61322104), National Science and Technology Innovation Training Program (No. SZDG2015002), Top-notch Academic Programs Project of Jiangsu Higher Education Institutions (No. PPZY2015A034) and Natural Science Foundation of Jiangsu Province (No. BK20140891).

REFERENCES

[1] Aggarwal J K, Ryoo M S. Human activity analysis: A review[J]. ACM Computing Surveys (CSUR), 2011, 43(3): 16.
[2] Aggarwal J K, Xia L. Human activity recognition from 3d data: A review[J]. Pattern Recognition Letters, 2014, 48: 70-80.
[3] Shotton J, Sharp T, Kipman A, et al. Real-time human pose recognition in parts from single depth images[J]. Communications of the ACM, 2013, 56(1): 116-124.
[4] Campbell L W, Bobick A E. Recognition of human body motion using phase space constraints[C]//Computer Vision, 1995. Proceedings., Fifth International Conference on. IEEE, 1995: 624-630.
[5] Wang J, Liu Z, Wu Y, et al. Learning actionlet ensemble for 3D human action recognition[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2014, 36(5): 914-927.
[6] Rodriguez M D, Ahmed J, Shah M. Action MACH: A spatio-temporal maximum average correlation height filter for action recognition[C]//Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008: 1-8.
[7] Wang C, Wang Y, Yuille A L. An approach to pose-based action recognition[C]//Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013: 915-922.
[8] Lu G, Zhou Y, Li X, et al. Efficient action recognition via local position offset of 3D skeletal body joints[J]. Multimedia Tools and Applications, 2015: 1-16.
[9] Vemulapalli R, Arrate F, Chellappa R. Human action recognition by representing 3d skeletons as points in a lie group[C]//Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. IEEE, 2014: 588-595.
[10] Muller M. Information retrieval for music and motion[M]. Heidelberg: Springer, 2007.
[11] Ofli F, Chaudhry R, Kurillo G, et al. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition[J]. Journal of Visual Communication and Image Representation, 2014, 25(1): 24-38.

[12] Veeraraghavan A, Srivastava A, Roy-Chowdhury A K, et al. Rate-invariant recognition of humans and their activities[J]. Image Processing, IEEE Transactions on, 2009, 18(6): 1326-1339.
[13] Wang J, Liu Z, Wu Y, et al. Mining actionlet ensemble for action recognition with depth cameras[C]//Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012: 1290-1297.
[14] Bishop C M. Pattern recognition and machine learning[M]. springer, 2006.
[15] Murray R M, Li Z, Sastry S S, et al. A mathematical introduction to robotic manipulation[M]. CRC press, 1994.
[16] Xia L, Chen C C, Aggarwal J K. View invariant human action recognition using histograms of 3d joints[C]//Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on. IEEE, 2012: 20-27.
[17] Zhu Y, Chen W, Guo G. Fusing spatiotemporal features and joints for 3d action recognition[C]//Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on. IEEE, 2013: 486-491.
[18] Ohn-Bar E, Trivedi M. Joint angles similarities and HOG2 for action recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2013: 465-470.
[19] Chrungoo A, Manimaran S S, Ravindran B. Activity Recognition for Natural Human Robot Interaction[M]//Social Robotics. Springer International Publishing, 2014: 84-94.
[20] Li W, Zhang Z, Liu Z. Action recognition based on a bag of 3d points[C]//Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on. IEEE, 2010: 9-14.
[21] Yang X, Tian Y L. Effective 3d action recognition using eigenjoints[J]. Journal of Visual Communication and Image Representation, 2014, 25(1): 2-11.
[22] Evangelidis G D, Singh G, Horaud R. Continuous gesture recognition from articulated poses[C]//Computer Vision-ECCV 2014 Workshops. Springer International Publishing, 2014: 595-607.