Signal Processing: *Image Communication* 33 (2015) 29–40

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Informative joints based human action recognition using skeleton contexts

Min Jiang [a,*], Jun Kong [a,b], George Bebis [c], Hongtao Huo [d]

[a] *Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, China*
[b] *College of Electrical Engineering, Xinjiang University, Urumqi 830047, China*
[c] *Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, United States*
[d] *Department of Information Security Engineering, People's Public Security University of China, Beijing 100038, China*

ARTICLE INFO

Article history:
Received 15 August 2014
Received in revised form
7 February 2015
Accepted 10 February 2015
Available online 18 February 2015

Keywords:
Action recognition
Skeleton contexts
Informative joints
Affinity propagation
CRFs
Kinect

ABSTRACT

The launching of Microsoft Kinect with skeleton tracking technique opens up new potentials for skeleton based human action recognition. However, the 3D human skeletons, generated via skeleton tracking from the depth map sequences, are generally very noisy and unreliable. In this paper, we introduce a robust informative joints based human action recognition method. Inspired by the instinct of the human vision system, we analyze the mean contributions of human joints for each action class via differential entropy of the joint locations. There is significant difference between most of the actions, and the contribution ratio is highly in accordance with common sense. We present a novel approach named skeleton context to measure similarity between postures and exploit it for action recognition. The similarity is calculated by extracting the multi-scale pairwise position distribution for each informative joint. Then feature sets are evaluated in a bag-of-words scheme using a linear CRFs. We report experimental results and validate the method on two public action dataset. Experiments results have shown that the proposed approach is discriminative for similar human action recognition and well adapted to the intra-class variation.

1. Introduction

Automatic human action recognition has been a highly active research area in computer vision [1–5]. The main goal of human action recognition is to segment the human target from uncontrolled background and analyze the motion sequences to interpret the meaning of the action automatically. This research can be widely applied in various domains, such as public security surveillance, virtual reality, and computer games. So far, most of the research work mainly focuses on action recognition of intensity video sequence captured by RGB cameras. However, intensity images are sensitive to illumination. Observed from different viewpoints, the same action presents very different resulting images. Self-occlusion makes this problem even worse. Especially in the case of clutter background, segmentation of the human body is a very challenging task. Human action involves dynamic spatial and temporal information. Even if the human body is segmented accurately, the difficulty occurs due to the complexity of human actions. Moreover, human actions are closely influenced by different culture, personal character and emotion shift. How to reveal latent intra-class spatial-temporal law of each action with considerable intra-class variability and inter-class overlap becomes the primary issue for action recognition.

Depth map has drawn much interest for human action recognition [6–19]. Depth map records the distance from the surface of the object to the camera. With the depth information, human body can be detected and segmented robustly. In particular, depth based human skeleton tracking technology [6,20] achieves outstanding precision and

* Corresponding author.
E-mail address: minjiang@jiangnan.edu.cn (M. Jiang).

http://dx.doi.org/10.1016/j.image.2015.02.004
0923-5965/

stimulates the researches of human action recognition using skeleton information.

Our focus in this paper is to establish a robust scheme for human action recognition based on estimated skeletons only (Fig. 1). Specifically, we evaluate the contribution of all the joints and construct informative joint set for each action class to eliminate the disturbance of unrelated joints. To make our representation robust against variation of human body size and orientation, we retargeted the skeletons to a standard skeleton and normalize the skeletons by translation, rotation and scaling. Similarities between postures are evaluated by skeleton contexts, a binned pairwise spacial distribution of informative joints. To improve the robustness, we propose using multi-scale bins. We perform the quantization with AP [21] (Affinity Propagation) method to cluster the feature vectors into n (n is determined by preferences) posture vocabularies. Encoded sequential features are trained upon linear CRFs. Experiments show that the recognition performance achieves high precision on two public action databases: MSRAction3D [13] and UTKinect [16]. It is robust to intra-variations in viewpoints, performance styles and individual body sizes. It also has good quality to distinguish inter-similarity between different action classes.

This paper is organized as follows: Section 2 briefly reviews related work in action recognition over depth map. Section 3 discusses and analyzes the informative joints based feature extraction using skeleton contexts. In Section 4, we use linear CRFs to classify the action samples with the proposed representation. Section 5 presents experimental results and discussions of the proposed approaches. Finally, conclusions are drawn in Section 6.

2. Related work

With the development of depth sensors, especially the launching of Microsoft Kinect, there has been an upsurge

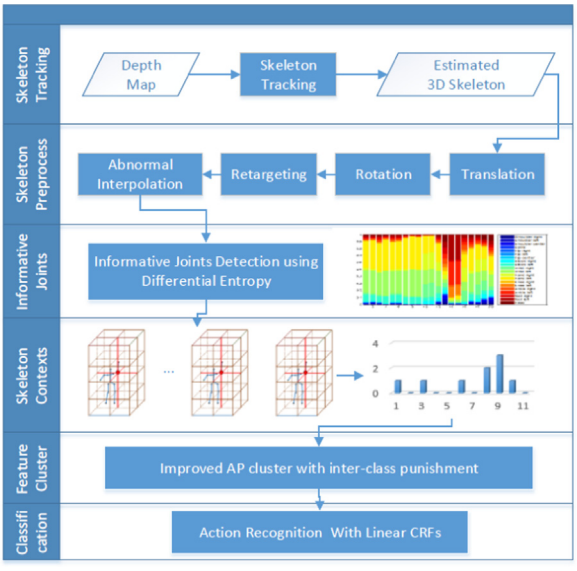

Fig. 1. Overview of the feature extraction and action classification scheme proposed in this paper.

of research on human recognition over depth map. Human action recognition using depth maps may be divided into two categories: algorithms using depth maps directly, and algorithms using estimated skeletons from depth maps.

In the first category, Lu et al. [11] introduce STIP's counterpart into depth video (called DSTIP). They describe the local 3D depth bin around the DSTIPs with a novel depth cuboid similarity feature (DCSF). DCSF features are clustered using K-means algorithm. Depth sequences are represented as a bag-of-codewords and classified by SVM with histogram intersection kernel. Oreifej et al. [12] describe the depth sequence using a histogram capturing the distribution of the surface normal orientation in the 4D space of time, depth, and spatial coordinates. They trained the Histogram of Oriented 4D Normals (HON4D) using SVM with polynomial kernel. Li et al. [13] employ an action graph to model explicitly the dynamics of the actions and a bag of 3D points to characterize a set of salient postures that correspond to the nodes in the action graph. To reduce the computation cost, they project the depth map onto the three orthogonal Cartesian planes and further sample a specified number of points at equal distance along the contours of the projections. This sampling scheme is view dependant and leads to a poor accuracy of 74.7% in Cross Subject Test on MSRAction3D dataset. Dubey et al. [14] proposed to use depth camera and 3D-MHIs to model the a 3D human shape in space–time. The 3D-MHI approach combines the MHIs (Motion History Image) which encodes a range of times in a single frame, with two additional channels, fDMHIs (forward depth MHIs) and bDMHIs (backward depth MHIs). Experiments report a high precision of 97% upon RGB-D data comparing a lower precision of 87% upon traditional RGB data. Note that these experiments simply aim to detect falls from other actions. These algorithms are used to recognize activities without dependence on skeleton tracking.

In the second category, Ohn-Bar et al. [15] characterize actions using pairwise affinities between view-invariant joint angle features over the performance of an action. Using cosine distance function, this skeleton based method arrives at a precision of 83.53% on MSRAction3D dataset. Lu et al. [16] present a compact representation of postures with histograms of 3D joint locations (HOJ3D) and train the features by discrete hidden Markov models (HMMs). Ofli et al. [17] sort the joints by the highly interpretable measures such as the mean or variance of joint angle trajectories and automatically select a few most informative skeletal joints. The experiments demonstrate that the sequence of the most informative joints (SMIJ) reveals significant discrimination for most human actions. But it is insensitive to discriminate different planar motions around the same joint. This limitation leads to a low classification rate on MSRAction3D dataset. Evangelidis et al. [18] encode the relative position of joint quadruples. This short, view-invariant descriptor is then represented by Fisher vectors and trained with a Gaussian mixture model. Sung et al. [22] use a two-layered maximum entropy Markov model (MEMM) to classify combined features of skeletal features, skeletal HOG image features, and skeletal HOG depth features. Lin et al. [23] reported high recognition accuracy (precision/recall of 97.7/97.2). They

proposed a method called Action Trait Code with a set of velocity types derived by the averages velocity of each body part. Yang et al. [24] designed a new action feature descriptor named EigenJoints which combine action information including static posture, motion property, and overall dynamics. Devanne et al. [25] propose representing human actions using spatio-temporal motion trajectories. The evolution of the 3D position of all joints within frames of action sequence forms the motion channel of the trajectory.

Some researches utilize the advantages of both the depth maps and the estimated skeletons. Wang et al. [26] propose a new feature called LOP feature which describes the "depth appearance" of each 3D joint. LOP feature is able to capture the relations between the human body parts and the environmental objects in the interaction. Ohn-Bar et al. [15] use a linear SVM to train both view-invariant joint angles feature and HOG2 feature extracted from color and depth images. To the best of our knowledge, this method gives the best precision of 94.84% on MSRAction3D dataset.

Based on the developing skeleton tracking technologies [6,20], it is much faster and more accurate now to reconstruct a 3D human body model or track human parts/joints comparing previous works over the intensive image. Johansson [27] proved that humans can recognize actions merely from the posture of a few moving light displays (MLD) attached to the human body. In other words, skeleton data has provided sufficient information for action recognition task. So far, reported skeleton-based methods alone generally perform lower precision than depth map based methods or methods using combined features, for example, on the public MSRAction3D dataset. The major cause is the noisy data introduced by the failure of skeleton tracking. This raises the question: how to construct a robust framework for action recognition with flawed skeleton tracking technology?

3. Action representation: skeleton contexts of informative joints

3.1. Skeleton preprocess

A continuous evolution of a series of human postures composes a human action. To make the action invariant to the absolute body position and the initial body orientation, we transform the coordinate system of all the postures. Inspired by Lu et al.'s work [16], we translate the skeleton along vector $-\overrightarrow{OH_C}$ (O is the origin (0,0) of coordinates system, H_C is the coordinate of the hip center). With the new origin of hip center, we rotate the skeleton and align the horizontal axis x with the direction of a vector $\overrightarrow{H_LH_R}$($\overrightarrow{H_LH_R}$ is a vector from left hip H_L to right hip H_R). We define the z-axis as the vector, which passes through the new origin o and is perpendicular to the new ground plane. Now the skeleton is independent of the viewpoints (Fig. 2).

Then we retarget all the posture samples to a common size to eliminate the bad influence of varied skeleton size. As a multi-rigid system, the movement of each joint pulls the adjacent joints as well. Let us suppose that the hip center is fixed. Thus the movement spreads from the hip center to the terminal joints. Based on this idea, we transform the structure of a skeleton to a root tree (Fig. 3) by designating the hip center as the root. By keeping the original adjacent relationship, the joints close to the hip center become fathers of their neighbor joints who are further away from the root. The terminal joints, such as head, hands and feet, become leaves. We choose a normal 3D skeleton $\text{Skel}^* = \{J_i^*\}_{i=1}^N$ as a standard skeleton model and calculate the length of each body segment. J_i^* is the ith joint. By keeping the direction of each segment vector, we scale the skeleton of the motion sample Skel according to the length of the segments in Skel*. All the joints (except for the hip) are moved $\overrightarrow{d_i}$ to a new position. The moving

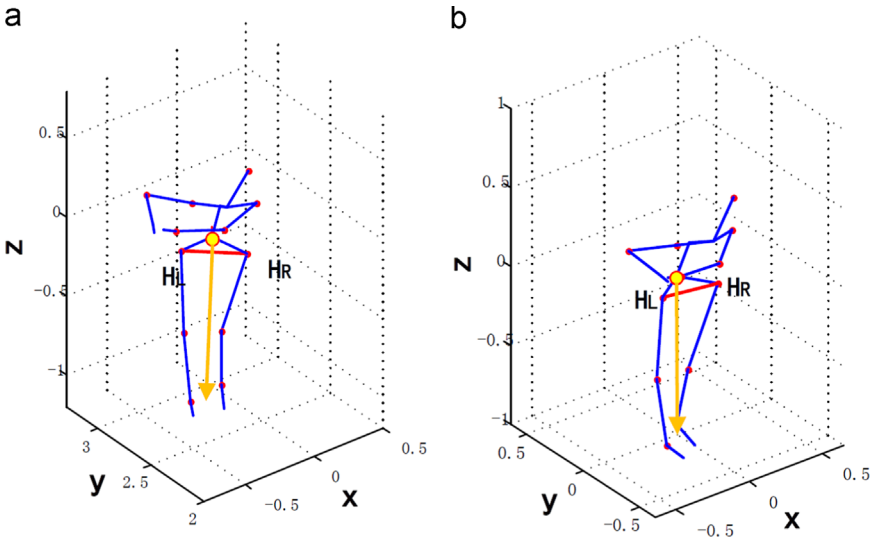

Fig. 2. Transformation for the coordinate system of the skeleton. (a) The original coordinate system for the skeleton. (b) The new coordinate system for the skeleton.

vector $\overrightarrow{d_i}$ is defined as

$$\overrightarrow{d_i} = \sum_{a \in A_i} \overrightarrow{\Delta d_a} + \overrightarrow{\Delta d_i} \tag{1}$$

where A_i is the set of ancestors of joint i in the tree, and $\overrightarrow{\Delta d_i}$ is the moving vector of joint i. We define $\overrightarrow{\Delta d_i}$ as follows:

$$\overrightarrow{\Delta d_i} = \overrightarrow{J_f - J_i} + \frac{\overrightarrow{J_i - J_f}}{|J_f - J_i|}|J_f^* - J_i^*| \tag{2}$$

where J_i is the 3D coordinates of joint i in posture sample Skel, and J_i^* is the coordinates of joint i in the standard skeleton model *Skel**. J_f denotes the 3D coordinates for the father of joint i.

Moreover, although Shotton et al. [6]'s method has achieved state-of-the-art accuracy, it deduces the best candidate only based on a single depth map. Without considering the temporal information, some abnormal skeletons are produced. For example, in the MSRAction3D dataset, some seriously self-obscured postures such as squat are severely corrupted. To address this problem, we evaluate the joint angles based on joint angle limit constraints, which are adopted from the biomechanics community. We only keep those posture samples that obey the limit constraints whose joint angles satisfying $\theta^l \le \theta \le \theta^u$, where θ^l and θ^u are the lower and upper bounds for 3D human pose respectively. For the rest that violate the constraints, we assume that all the joints move in an approximate straight trajectory from time $t - n_1\Delta t$ to $t + n_2\Delta t$, where $n_1, n_2 \in N$, t is the moment when the abnormal posture produced. Δt is the interval between 2 sampling. For the default sampling rate of 30 fps, $\Delta t = \frac{1}{30}$ s. Thus, we apply Eq. (3) to estimate the possible skeleton configuration Skel_t' of the abnormal posture at time t:

$$\text{Skel}_t' = \frac{\beta_1 \text{Skel}_{t-n_1\Delta t} + \beta_2 \text{Skel}_{t+n_2\Delta t}}{2} \tag{3}$$

where $\text{Skel}_{t-n_1\Delta t}$ and $\text{Skel}_{t+n_2\Delta t}$ are the closest normal postures performed before and after time t respectively. Parameters β_1 and β_2 control the posture timing variations. In our experiments, $\beta_1 = n_2/(n_1+n_2)$ and $\beta_2 = n_1/(n_1+n_2)$.

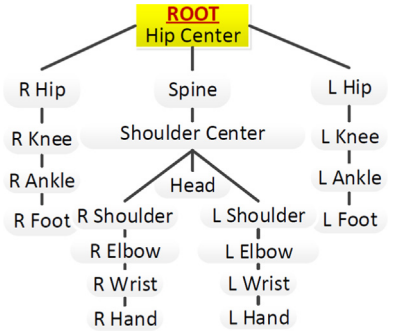

Fig. 3. Joint root tree with the root of hip center.

3.2. Informative joint vector

Instinctively, human tends to pay more attention to the moving targets and be blind to other static parts. That is actually one of the biggest secrets of magic. Motivated by this human nature, we try to find these "informative joints" for each action. While performing an action, not all of the joints are engaged. Among the 20 joints, some can be considered as 'redundant'. These joints are either too close, like hand and wrist, or relatively non-moving, like spin center. Moreover, when we go into more details of the joints and the related actions, we find that the ranges of contribution value produced by different joints are significantly different. In order to reveal this relationship, we perform an analysis based on the variation of moving distance of each joint during an action instance. One common way to evaluate information of a continuous signal is the differential entropy, which extends the idea of Shannon entropy. We take the idea of differential entropy to compare the contribution produced by each joint in each action.

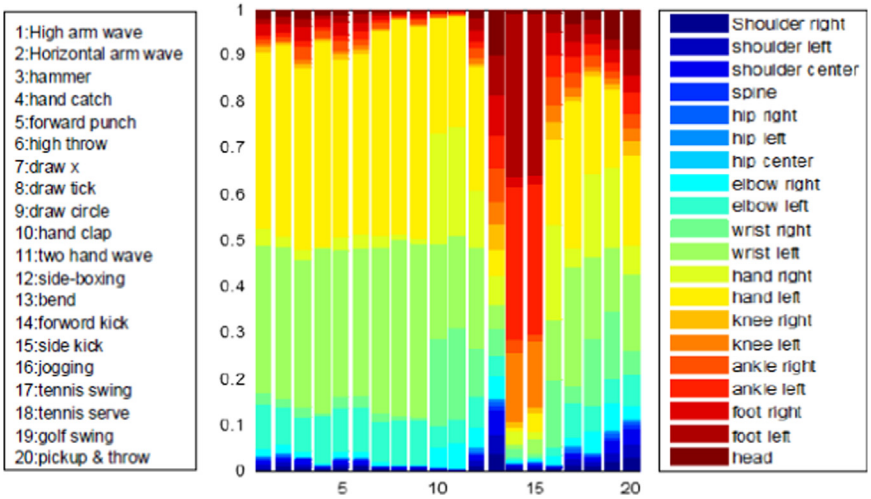

Fig. 4. Stacked histogram for the joint contribution of 20 different actions in the MSRAction3D dataset. Each bar corresponds to distinct action and shows the relative contribution each joint makes to the action.

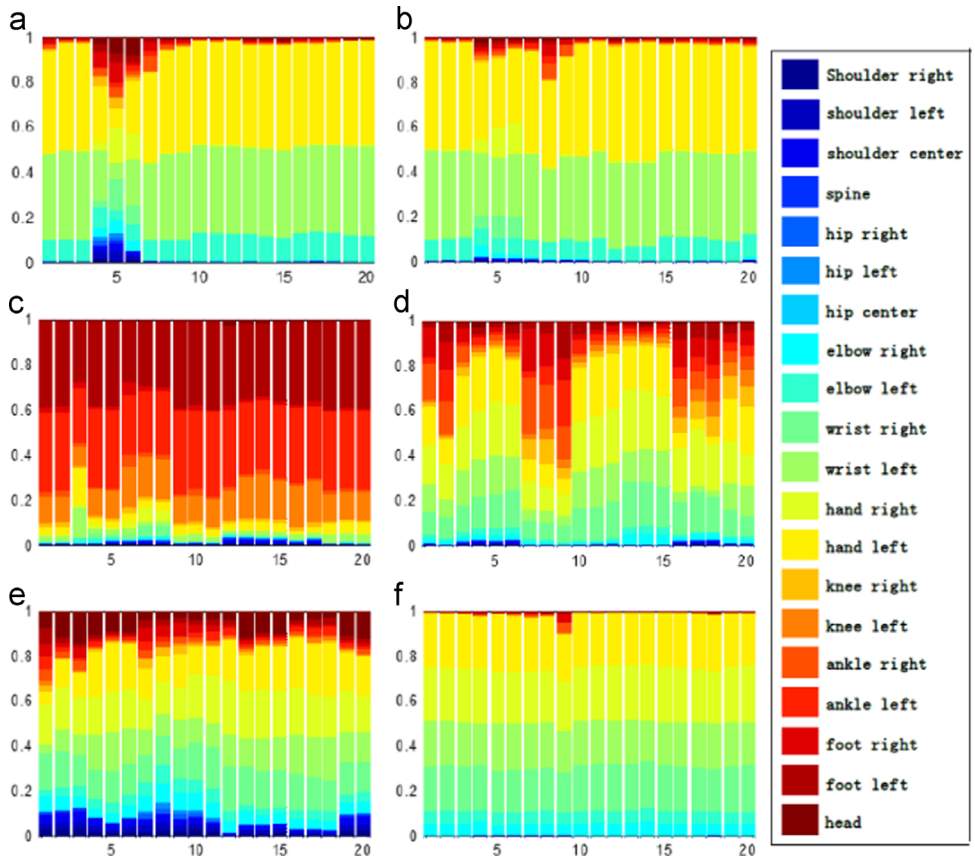

Fig. 5. Stacked histogram of the joint contribution for 6 typical action in the MSRAction3D dataset: high arm wave, draw X, side kick, jogging, golf swing and two hand wave. Each bar indicates the contribution ratio of each joint to the action in one instance. For instances in the same action class, the contribution bars follow a similar pattern. (a) High arm wave. (b) Draw X. (c) Side Kick. (d) Jogging. (e) Golf swing. and (f) Two hand wave.

We accumulate the entropy of all the instances in action class j for joint i. Fig. 4 shows the contribution of the joints for 20 different actions across all performance by 10 actors in the MSRAction3D dataset [13]. Each stacked histogram column in the figure gives the percentage of the contribution of 20 joints for a single action. Notice that for different actions, the engaged joints are very different. For high arm wave (Action 1), only left hand, left wrist and left elbow participate in the performance. However, for Forward Kick (Action 14), the joints on the leg deliver most of the information. Fig. 4 shows that many joints contribute very little for action recognition in most actions. On the contrary, they even bring noises. Thus, we only need to concentrate on some of the joints, which highly agrees with the human instinct. Fig. 5 illustrates the detailed contribution of each instance. 6 typical actions with 20 randomly chosen instances are presented. For each instances in the same action class, the contribution bars follow a similar pattern. For different actions, the patterns are very different. Motivated by this phenomenon, Ofli et al. [17] proposed a new representation of human actions called sequence of the most informative joints (SMIJ). They use joint angle trajectory variances to evaluate the value of the joints and then represent the action as a sequence of these most informative joints. However, some actions, such as high arm wave and draw X, have very similar SMIJ representation. Fig. 5(a) and (b) gives contributions of high arm wave and draw X evaluated based on the moving distance of the joints. The patterns are essentially the same for these two classes of actions. Experiments in [17] achieve a poor performance on the MSRAction3D dataset [13]. 8 actions, which are almost performed by a same single arm, got a 0% recognition rate. Instead of using the sequence of the most informative joints, we choose to create a most-valuable joint subset called informative joints, in which the joints contribute 85% of the entropy (Fig. 6 gives the selected informative joints on the MSRAction3D dataset) . We neglect the rest of the joints by setting them zero. With this compact joint set, which directly leads to a feature dimensionality reduction, our method obtains a good improvement in the efficiency and precision.

3.3. Skeleton contexts

In our approach, we treat an action as a posture code sequence and then train the sequence with a linear CRFs framework. To encode an action into a posture code sequence, all the posture samples are clustered with Affinity Propagation (AP) [21] and labeled with its cluster center. A key problem to AP cluster method is how to determine the similarity between two data points. Due to

the different view point and the posture deformation presented by different subjects, there are large deviations for the 3D space position of skeleton joints between postures in the same category.

As a key contribution, we propose a novel descriptor, called skeleton contexts. Consider a pairwise feature vector originating from joint J_i to all other points in a skeleton. This vector expresses the configuration of the entire skeleton relative to joint J_i. For a whole skeleton, sets of $M\times(M-1)$ (M is the number of joints) features give the details of skeleton configuration. However, this detailed representation is too exact since the posture samples are greatly varied from one instance to another for the same posture class.

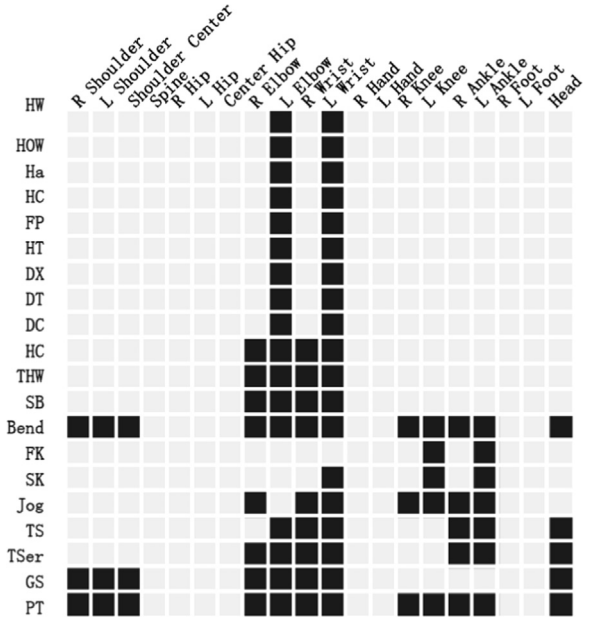

Fig. 6. Informative joints on the MSRAction3D dataset. Each row indicates the configuration of informative joints for one action class (the abbreviation for actions can be found in Table 1). The dark blocks in each row indicate the selected joints for this action class.

We identify the distribution over relative positions of informative joints as a more robust and compact, yet highly discriminative descriptor. We partition the 3D space into $m\times n\times l$ bins with the center o of joint J_i, as shown in Fig. 7(a). We cast the rest joints into 3D bins separately with Gaussian probability density function to indicate the probability that a human joint could be located in the range of a bin with respect to joint J_i (Fig. 7(c)).

Let (x_J, y_J, z_J) be the coordinate of joint J, x_l, x_u, y_l, y_u, and z_l, z_u indicate the lower and upper boundary of the bin B_j in x-direction, y-direction and z-direction respectively. For each direction, the probability function P for joint J in bin B_j can be described as

$$P_x(x_l < x_J \le x_u; x_J; \sigma_x) = |\Phi(x_l, x_J, \sigma_x) - \Phi(x_u, x_J, \sigma_x)| \tag{4}$$

Similarly,

$$P_y(y_l < y_J \le y_u; y_J; \sigma_y) = |\Phi(y_l, y_J, \sigma_y) - \Phi(y_u, y_J, \sigma_y)| \tag{5}$$

$$P_z(z_l < z_J \le z_u; z_J; \sigma_z) = |\Phi(z_l, z_J, \sigma_z) - \Phi(z_u, z_J, \sigma_z)| \tag{6}$$

where Φ is the CDF (cumulative distribution function) of Gaussian distribution, $\sigma_x = d/3m$, d is the diagonal length of bins, m is the number of the bins in the x-direction, similarly $\sigma_y = d/3n$ and $\sigma_z = d/3l$.

The joint probability function is then calculated as

$$\begin{aligned}\wp\left(x_l < x_J \le x_u, y_l < y_J \le y_u, z_l < z_J \le z_u; \sum\right) &= P_x(x_l < x_J \le x_u; x_J; \sigma_x)\\ &\times P_y(y_l < y_J \le y_u; y_J; \sigma_y)\\ &\times P_z(z_l < z_J \le z_u; z_J; \sigma_z)\end{aligned} \tag{7}$$

We calculate the joint distributions for all informative joints. For the rest of the joints (non-informative joints), we simply set $\wp = 0$.

For joint J_i, we compute the coarse histogram $h_{i,j}$ of the relative distribution for the remaining $M-1$ joints located in Bj:

$$h_{i,j} = \sharp\{J_k | \wp^i_{kj} > \rho \wedge k \ne i\}, \quad \rho \in [0\ \ 1) \tag{8}$$

where $\wp^i_{kj}$ denotes the distribution of joint J_k in Bj with the reference joint J_i. As $\wp^i_{kj}$ denotes the pairwise distribution

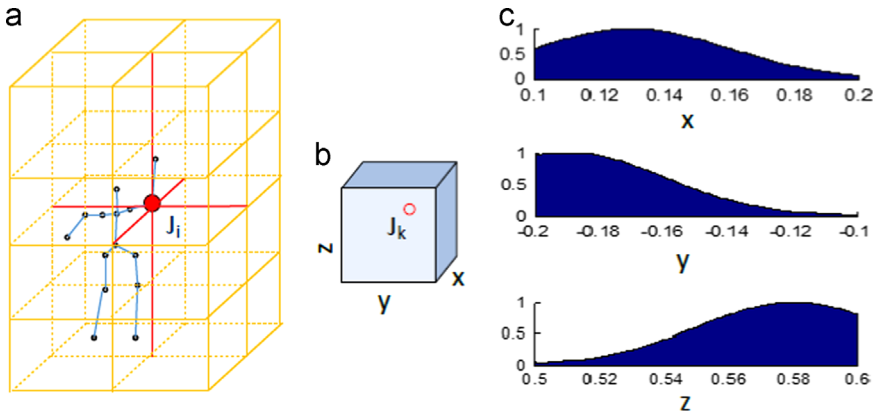

Fig. 7. (a) $m\times n\times l$ bin partitions of 3D normalized skeleton space with the center of red joint. (b) Example of bin $(x_l{:}x_u, y_l{:}y_u, z_l{:}z_u)$ with joint J_k located in it, $x_l = 0.1$, $x_u = 0.2$, $y_l = -0.2$, $y_u = -0.1$, $z_l = 0.5$, $z_u = 0.6$, $x_J = 0.13$, $y_J = -0.19$, $z_J = 0.58$. (c) Distributions with respect to the three dimensions separately to indicate the probability that a human joint could be located in the range of a bin.

in bins, lower ρ allows wider range of neighbor bins to be counted in Eq. (8). In our experiment, $\rho = 0.12$.

The collection of the histogram $H_{SC} = \{h_{i,j} | 1 \le i \le M, 1 \le j \le m \times n \times l\}$ forms the posture feature descriptor, named skeleton contexts (SCs), which is high-dimensional and sparse. To improve the robustness, we extract SCs features at different scales of bins. In our experiment, SCs features are extracted at the partitions of $3 \times 3 \times 3$, $5 \times 5 \times 5$ and $7 \times 7 \times 7$. To keep the classification task more efficient, we perform PCA separately for each scale to reduce dimensionality. We choose to retain 99% of the variance. Thus we get a compact descriptor composed of SCs feature at different scales, which is denoted as $\hat{H}_{SC} = \{\hat{h}_i | 1 \le i \le L\}$ where L is the number of the chosen principal components from all the scales.

3.4. Feature vector clustering

We applied a powerful clustering algorithm, Affinity Propagation (AP), to group the informative joints based SCs features. AP was proposed by Frey and Dueck [21]. The best advantage is that AP takes all the data as initial exemplars. Thus we do not need to worry about the bad initial problem. Moreover, AP takes "preferences" for each data point which automatically determine the number of clusters. The number of the clusters rises with bigger "preferences". Since the data points with larger "preferences" are more likely to be chosen as exemplars. To ensure equality for all data point, we choose a common value, the median of the input similarities, as the "preferences" for all data point. Without the influence of initialization, AP is steady and fast.

This algorithm takes as input a collection of real-valued similarities between data points, where the similarity $s(a, b)$ indicates how well the data point b is suited to be the exemplar for data point a. As SCs features are distributions represented as histograms, we choose the chi-square test to describe the similarity between $\hat{H}_{SC_a}$ and $\hat{H}_{SC_b}$:

$$s(a,b) = -\frac{1}{L}\sum_{i=1}^{L}\frac{|\hat{h}_i^a - \hat{h}_i^b|^2}{\hat{h}_i^a + \hat{h}_i^b} \tag{9}$$

where $\hat{h}_i^a$ denotes the ith principal components in compact SCs histogram feature for posture a. The higher the $s(a, b)$ is, the more similar the two postures are. To further weaken the noise of view variation, we rotated the skeletons to find the best match of orientation for skeletons a and b. Since all the skeletons have been rotated in the preprocess procedure, in this step, we only need to make minor adjustments.

Let φ denote the cluster exemplar set and cluster(k) denote the dataset in which all data choose k as their exemplar. Let us define the cost function as

$$E(\varphi) = -\sum_{k \in \varphi, a \in \text{cluster}(k)} s(a,k) \tag{10}$$

Thus, the main goal is to minimize the overall sum of similarities between data points and their exemplars:

$$\varphi = \text{argmin}_{\varphi}\, E(\varphi) \tag{11}$$

In our experiment, the MSRAction3D dataset contains 20 classes of actions, 23,478 postures. The resulting similarity matrix is $23{,}478 \times 23{,}478$. With such a large-scale dataset, finding the minimum of $E(\varphi)$ is a practically hard task. AP cluster takes the idea of belief propagation (BP) which passes messages between data points recursively until the clustering is stable. To improve the efficiency of clustering, we analyze the histogram of the huge similarity matrix upon our test datasets. The result shows that nearly half of the similarities are relatively small and the corresponding joint pair is too far to be grouped in the same cluster. Since messages do not pass between the points a and b if the similarity $s(a, b) = -\infty$, we replaced the small similarities who satisfy $s(a, b) < \rho$ with $-\infty$. In our experiment, ρ is the median of similarities.

According to the original AP cluster, there are two types messages passing between data points. One is the responsibility $r(i, k)$, which reflects the support of data point i to choose k as its exemplar. The other is the availability $\alpha(i, k)$, which indicates the appropriateness for k to become an exemplar. After a certain number of iterations, $K_i = \text{argmax}_{k=1\ldots N}\,(r(i,k) + \alpha(i,k))$ identifies the data point K as the exemplar for point i. If $K_i = i$, then i is exactly the exemplar.

As some actions share the same informative joints set, like high arm wave and draw X, the postures performed in these actions are relatively more similar than others and tend to be clustered in a same group. This leads to a low discrimination in these actions. To solve this problem, we set up a punishment $B_k(\varphi)$ for those who choose an exemplar in a different action class, with $\gamma \in [0, 1]$:

$$B_k(\varphi) = \begin{cases} \gamma & \text{if } C_{ik} = 1 \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

where $C_{ik} = 1$ denotes that data point i chooses data point k as its exemplar who belongs to another action class, otherwise, $C_{ik} = 0$. Thus, the main aim of this task is modified as

$$\varphi = \text{argmin}_{\varphi}\, E(\varphi) \tag{13}$$

where

$$E(\varphi) = -\sum_{k \in \varphi, a \in \text{cluster}(k)} s(a,k) - \beta \sum_{k \in \varphi} \ln(B_k(\varphi)) \tag{14}$$

The feature space is classified automatically into I clusters and the exemplars of each cluster form a K-word vocabulary. All the posture samples are represented by its exemplar. Then each action is represented as a time series of the visual words.

4. Action recognition with CRFs

CRFs have been proven to be very useful in structured data related applications. Especially in natural language processing (NLP), CRFs are currently a state-of-the-art technique for many of its subtasks. Similar to NLP, human action recognition can be regarded as the prediction of a class label for an input posture sequences. The true meaning of a posture sequence highly depends on the order that the postures appear. Prior and posterior observation context is critical for improving the confidence and accuracy of action recognition.

In this paper, we consider a linear chain CRFs model. Linear chain CRFs can be thought as the undirected graphical model version of HMMs. The biggest advantage of CRFs over HMMs is that they can handle overlapping features and seamlessly represent contextual dependencies.

As shown in Fig. 8, the random variables are linked by undirected edges indicating dependencies. A labeling of a CRFs corresponds to a classification of the vertices by assigning a label to each vertex (variable) from a set of labels $\mathbb{L}=\{1,\ldots,K\}$. $y_{t\in[1,T]}$ is a set of random discrete temporal states whose values the task required the model to predict. X is a set of input postures represented by the SCs feature of its exemplar. Intuitively, we construct the features to express correlations among states and the observation vector forward and backward in the whole time, as well as the dependence between y_t and y_{t-1}. We only consider the dependency between linked vertices pair and the correlations among states and the observation vector. For the model of chain CRFs, the joint distribution over the label sequence Y given X has the form

$$p\left(Y|X\right)=\frac{1}{Z_\theta(X)}\exp\left(\sum_{a,t}\lambda_a h_a(y_t,X)+\sum_{a,b,t}\beta_{ab}g_{ab}\left(y_t,y_{t-1}\right)\right) \tag{15}$$

where $Z_\theta(X)$ is the normalizing factor. Let $1[\cdot]$ be an indicator function, $y\in\mathbb{L}$. Intuitively, we construct the features to express correlations among states and the observation vector as

$$h_a(y_t,X)=1[y_t=m]\cdot x_t^i \tag{16}$$

where x_t^i denotes the ith feature of the observation X at time t, $t\in[0,T]$. Similarly, the features that model the dependence between y_t and y_{t-1} are

$$g_{ab}(y_t,y_{t-1})=1[y_t=m_1\wedge y_t=m_2] \tag{17}$$

We train the models for each action separately. Without the initial problem of clustering, we only need to train our training dataset once. For the test dataset, we first calculate the most active joint set JS_{test} during a performance and compare it with the informative joint set JS_{info} for each action. We match these instances over those models whose informative joint set satisfies $JS_{info}\subseteq JS_{test}$. Action recognition for each motion sample is then achieved by selecting the model with the highest log likelihood value.

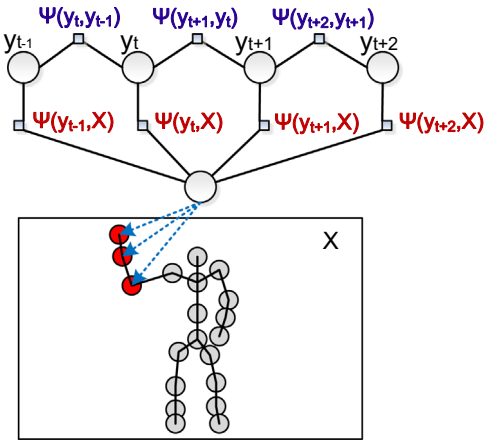

Fig. 8. Factor graph of a linear chain CRFs action model. The red vertices stand for the selected informative joints.

5. Experiments

We evaluate the proposed action representation over two different datasets: MSRAction3D [13] and UTKinect [16]. We compare our method with the state-of-the-art skeleton-based methods. The empirical results show that the proposed framework outperforms most of the state-of-the-art methods based on skeletons estimated from depth maps.

5.1. Experiments on the MSRAction3D dataset

The public MSRAction3D database [13] is an action dataset captured by a depth camera similar to the Kinect device with 15 Hz. The resolution is 320×240 DPI. This dataset contains 20 classes of actions: 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 and pickup & throw. There are ten actors; each actor performs each action 2 or 3 times. There are totally 23,478 posture samples, 567 action samples. Although the scenery background is clean, this action dataset is still challenging for action recognition. The postures it contains cover the movements of all human 20 joints. Many of the actions in the dataset are highly similar to each other. For example, 9 of the 20 actions (high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw X,

Table 1
The three subsets of actions used for the MSRAction3D dataset.

Action set 1 AS1	Action set 2 AS2	Action set 3 AS3
Horizontal arm wave (HOW)	High arm wave (HW)	High throw (HT)
Hammer (Ha)	Hand catch (HC)	Forward kick (FK)
Forward punch (FP)	Draw X (DX)	Side kick (SK)
High throw (HT)	Draw tick (DT)	Jogging (Jog)
Hand clap (HC)	Draw circle (DC)	Tennis swing (TS)
Bend	Two hand wave (TW)	Tennis serve (TSer)
Tennis serve (TS)	Forward kick (FK)	Golf swing (GS)
Pickup & throw (PT)	Side boxing (SB)	Pickup & throw (PT)

draw tick and draw circle) use only left arm during the whole performance; forward kick and side kick have no difference except for the directions that joints move. Some postures are largely occlusive, like bend. MSRAction3D database contains color images, depth maps and skeleton data. We only use the skeleton data for action recognition.

To analyze the ability of the proposed method to differentiate similar actions and complex actions, we group the actions into 3 subsets the same as in [16]. Each subset comprises 8 actions (see Table 1). Actions in dataset AS1 and AS2 are similar. While the actions in dataset AS3 are relatively complex with more joints engaged. We use the full set of MSRAction3D database.

We conduct two experiments by following the same experimental setup as the works evaluated on MSRAction3D. In experiment I, we evenly choose $\frac{2}{3}$ samples per action as training data, the rest $\frac{1}{3}$ samples as testing. To analyze the contribution of SCs (skeleton contexts) feature and informative joints, we first perform the experiment merely using SCs feature. Then we evaluate the SCs feature on the selected informative joints. As shown in Table 2, informative joints can eliminate the noises introduced by other non-relevant joints and produce higher recognition precision. We also compare our method with HO3DJ [16], EigenJoints [24] and space–time pose [25]. Similar to our method, these methods use skeletal joint locations to form their representation of postures. As shown in Table 2, on AS1 dataset, we outperform existing methods [16,24,25]. We obtain competitive accuracies on AS2 dataset. Space–time pose [25] achieves remarkable recognition rate on AS3. Note that 10 corrupted sequences are excluded in their tests. We examine the confusion matrix for the 3 subsets, presented in Fig. 9. As we use informative joints to build the features, actions using different joint sets are highly discriminated, as expected. Instead, actions with same informative joints, such as draw X and draw Circle, are more easily confused due to the similarity of the binned space distribution. Actions in AS3 are more complex and noisy, and the contributions of the joints are much more evenly distributed, especially Pickup & throw, which is severely self-obscured. With the dependance on 3D space distribution, our method gives a lower precision comparing to the results on AS1 and AS2.

Table 2
Recognition rate (%) on the MSRAction3D dataset in experiment I.

Method	AS1	AS2	AS3	Overall
HOJ3D [16]	98.6	97.9	94.9	97.2
EigenJoints [24]	97.3	**98.7**	97.3	**97.8**
Space–time pose [25]	93.4	93.9	**98.6**	95.3
SCs	98.4	97.9	94.3	96.9
SCs + informative joints	**98.8**	98.3	95.8	97.7

The best precision for each experiment is highlighted in bold.

In experiment II (cross-subject experiment), we randomly choose samples from half of the actors (5 actors) as a training set. We use the rest half actors as test set. We run the experiment 10 times with different training sets and report the mean performance, as shown in Table 3. We compare our method with other four state-of-the-art algorithms that use skeleton information. We can observe that our method shows stronger invariant to the posture deformation of the individual actor. Our method outperforms HOJ3D [16] and EigenJoints [24] on AS1 and AS2. On AS3, Points in a Lie Group [28] achieves remarkable precision of 98.22% and indicates that it is the best in modeling complex actions among these methods. Table 4 shows the cross-subject recognition performance of some other state-of-the-art methods. These methods utilize the skeleton information as well. To our best knowledge, among the methods merely using skeleton information, [25] achieves by far the best performance as high as 92.8% on the MSRAction3D in cross-subject experiment. Note that tests in Table 4 are performed on refined data of MSRAction3D database, while we use the full dataset without excluding any corrupted sequences. Fig. 10 shows the confusion matrix of experiment II using SCs based on informative joints. We observed that actions that share the similar informative joints tend to be more confusable, such as draw X, draw tick and draw circle. Accordingly, those actions with distinct informative joints set achieve 100% precision. For example. Hand Clap in AS1 uses and only uses both arms to perform the action, which is rather different to other actions in AS1.

5.2. Experiments on the UTKinect dataset

We also evaluate our proposed method on the UTKinect dataset [16]. This dataset is captured by a single stationary Kinect. There are 10 types of human actions in indoor

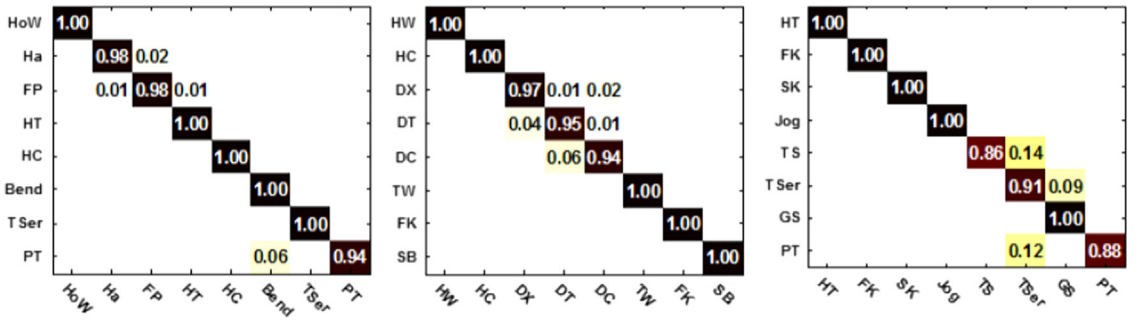

Fig. 9. Confusion matrix of experiment I on the MSRAction3D dataset using SCs and informative joints(left: AS1; center: AS2; right: AS3). Each column of the matrix represents the action class that predicted, while each row represents the actual action class for the instances. Each value represents the percentage that the *i*th row class instances are recognized as the *j*th column class.

settings, including Walk, SitDown, StandUp, PickUp, Carry, Throw, Push, Pull, Wave and ClapHands. Each action was performed by 10 different subjects for 2 times: 9 males and 1 female. Altogether, the dataset contains 6220 frames of 200 action samples. This dataset offers great challenges. First, in some action samples, parts of the human body are invisible. This is mainly caused by the object–person occlusions and body part out of field of view. Second, subjects are instructed to use different limbs while performing the same action. Third, the action sequences are captured from different views.

To compare our work to the state-of-the-art algorithms in [16,25], we follow the same experimental setup. We use the Leave One sequence Out Cross Validation (LOOCV) method. For each iteration, one sequence is used as test and all other sequences are used as training. We run the experiment 10 times. Each time a sequence is randomly chosen and used as testing. Fig. 11 illustrates the contribution of each joints for 10 actions on the UTKinect dataset. As shown in Fig. 11, in actions of Walk, SitDown, StandUp, PickUp and Carry, almost all the joints equally contribute to the action. While in actions of Throw, Push and Pull, only right arm moves significantly. Both arms are used in the action of WaveHands and ClapHands. UTKinect dataset is far more noisy than MSRAction3D dataset. Not only because subjects use different limbs while performing the same action, but because they also perform more casually. More irrelevant joints are involved in the performance. For example, some subjects perform an exaggerated arm movement while sitting down. Fig. 12 gives the learned informative joints for the UTKinect dataset. To our best knowledge, so far Points in a Lie Group [28] achieves the best precision on the UTKinect dataset. We obtain an accuracy similar to the work in [16,25,29] (see Table 5). From the results we can see that, the recognition rate is significantly lower than the results on MSRAction3D dataset. With the selected informative joints, the confusion between actions using different informative joints is well controlled. But for most of the actions, informative joints does not play a positive role. The recognition precision of Carry is pretty low by using space–time pose [25], SCs, and SCs with informative joints. The major problem is that Carry is highly confused with Walk. Carry is composed of the movements of legs and the arms. The movement of legs is almost the same as Walk. Thus some samples produce higher log likelihood value over the model of Walk. Our method achieves low precision on Push, even if we applied the informative joints. We argue that the differences are subtle between Push and Throw in the case of the pairwise space distribution of informative joints.

To eliminate the disturbance of irrelevant joints, we manually set the interested joints set (see Fig. 13) and repeat the experiment with the same setup. Fig. 14 compares the result of experiments upon auto-selected informative joints with the manually selected set. The overall precision using the manually selected informative set is 92.9% which improves the original experiment by 1.0%.

Table 3
Recognition rate (%) on the MSRAction3D dataset in experiment II.

Method	AS1	AS2	AS3	Overall
HOJ3D [16]	88.0	85.5	63.5	79.0
EigenJoints [24]	74.5	76.1	96.4	83.3
Space–time pose [25]	90.1	**90.6**	97.6	**92.8**
Points in a Lie group [28]	**95.29**	83.87	**98.22**	92.46
SCs	81.3	80.7	83.2	81.7
SCs with informative joints	88.7	87.7	88.5	88.3

The best precision for each experiment is highlighted in bold.

Table 4
Recognition rate (%) of existing methods on the MSRAction3D dataset in cross subject test.

Method	Overall
JAS (LCSS) [15]	53.95[a]
SVM on joint angles [15]	80.29[a]
JAS (Cosine) [15]	81.37[a]
SVM on joint angles+MaxMin [15]	81.63[a]
JAS (weighted Euclidean) [15]	82.20[a]
JAS (Cosine)+MaxMin [15]	83.53[a]
FV of skeletal quads [18]	89.86[a]
Space–time pose [25]	**92.80**[a]
SMIJ+SVM [17]	33.99[b]
HMIJ+SVM [17]	41.18[b]
Joint angles + SIMJ [17]	47.06[b]
Discriminative LDS [19]	90.00[b]

The best precision for each experiment is highlighted in bold.
[a] 10 corrupted sequences are excluded.
[b] A subset of 17 actions is used.

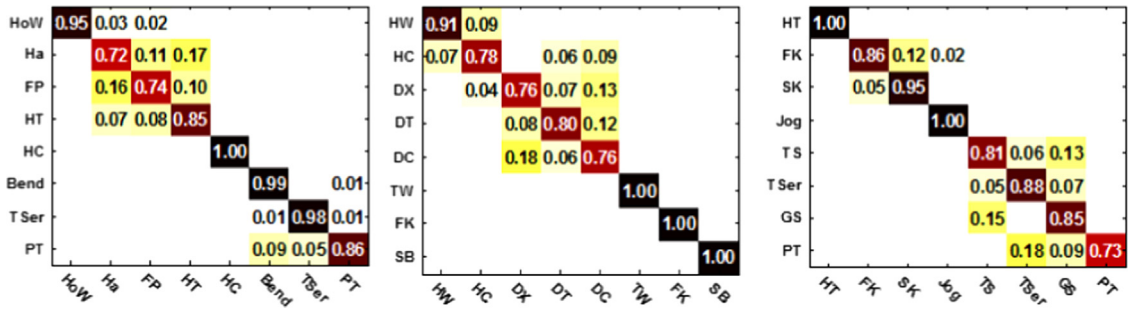

Fig. 10. Confusion matrix of experiment II on the MSRAction3D dataset using SCs and informative joints(left: AS1; center: AS2; right: AS3). Each column of the matrix represents the action class that predicted, while each row represents the actual action class for the instances. Each value represents the percentage that the *i*th row class instances are recognized as the *j*th column class.

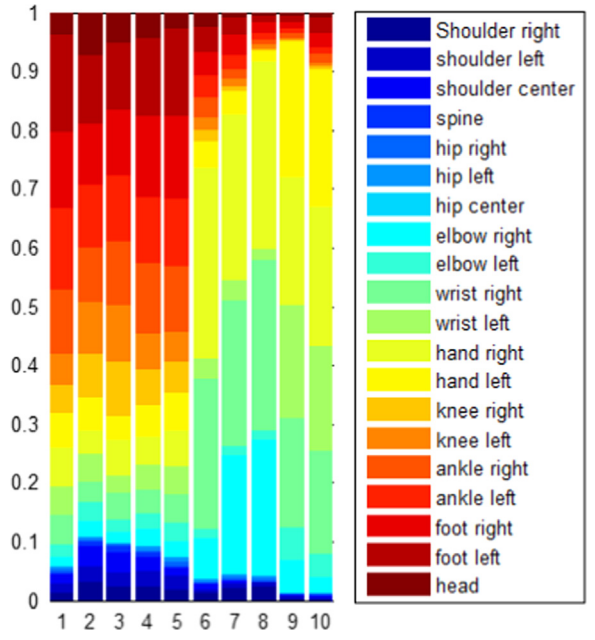

Fig. 11. Stacked histogram for the joint contribution of 10 different actions on the UTKinect dataset. Each bar corresponds to a distinct action and shows the relative contribution each joint makes to the action. The action series from 1 to 10 is 1: Walk; 2: SitDown; 3: StandUp; 4: PickUp; 5: Carry; 6: Throw; 7: Push; 8: Pull; 9: WaveHands; 10: ClapHands.

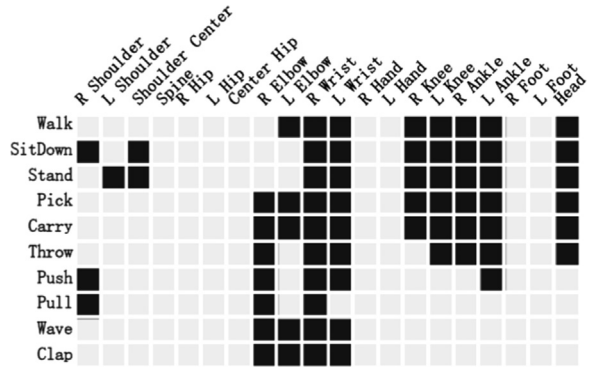

Fig. 12. Informative joints on the UTKinect dataset. Each row indicates the configuration of informative joints for one action class. The dark blocks in each row indicate the selected joints for this action class.

The recognition rate of Throw is observably improved since that only one subject uses left arm and thus the left arm is excluded from the informative joints with our selection strategy. However, Carry and Push are still highly confused with Walk and Throw for the similar skeleton configuration.

6. Conclusions and future work

In this paper, we have proposed a skeletal representation and have demonstrated its improved performance on the task of action recognition using skeleton data only. The main contributions of the paper are as follows: (1)

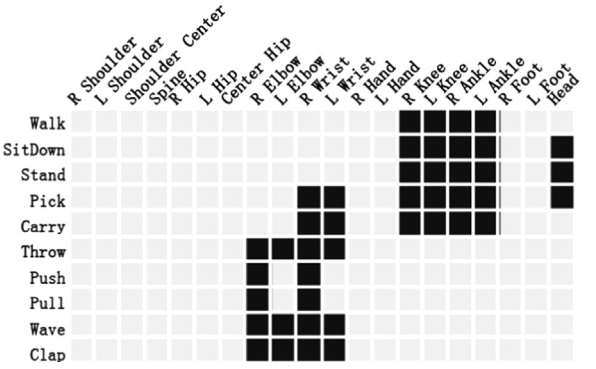

Fig. 13. Manually selected joints on UTKinect dataset. The dark blocks in each row indicate the selected joints for this action.

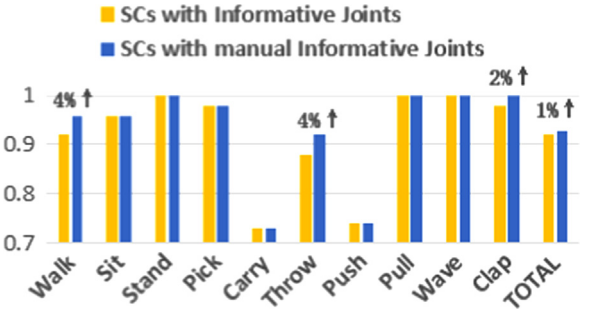

Fig. 14. Comparisons of recognition accuracy (%) on UTKinect dataset between methods using automatically selected informative joints and manually selected joints.

Table 5
Recognition rate (%) on the UTKinect dataset.

Method	Walk	Sit	Stand	Pick	Carry	Throw	Push	Pull	Wave	Clap	TOTAL
HO3DJ [16]	**97**	92	94	98	**98**	59	82	93	**100**	**100**	90.9[a]
Space–time pose [25]	90	**100**	**100**	**100**	68	**95**	**90**	**100**	**100**	80	91.5[a]
SCs	88	91	**100**	95	73	89	74	94	**100**	98	90.2[a]
SCs with informative joints	92	96	**100**	98	73	88	74	**100**	**100**	98	**91.9**[a]
Random forests [29]	90	90	**100**	**100**	78	90	70	**100**	**100**	**100**	91.9[b]
Points in a Lie group [28]	–	–	–	–	–	–	–	–	–	–	**97.08**[b]

The best precision for each experiment is highlighted in bold.
[a] LOOCV: 1 observation is used as the validation set and the remaining observations as the training set.
[b] Cross-subject: half of the subjects are used for training and the remaining half for testing.

informative joints indicate the most active joints during an action performance which draws most of the attention of the human vision system. Consistent with the instinct of the human vision system, our informative joints based method eliminates noise by ignoring the joints of small contributions. (2) We use binned pairwise space distribution of informative joints to build discriminative skeleton contexts. All the skeletons were first be normalized by retargeting, rotation and validation after being estimated from depth maps. In this sense, our representation is strongly invariant to individual size and shape, and viewpoint. (3) Improved affinity propagation was used to automatically find the exemplar features without worrying about bad initialization. The experiment results on the public datasets show that our informative joints based method is robust dealing with the mis-labeled joints.

So far, performance of skeleton based methods is generally less accurate than the depth map based methods or methods using combined features. The first reason is that joints of some complex actions tend to be mis-labeled with current skeleton tracking methods. With the remarkable development of skeleton tracking technology, as well as the fast improvement of sensor hardware, great progresses can be expected for skeleton based methods in the near future. The second reason is that skeleton based methods cannot tell the difference between actions with similar skeleton configuration. Future work regards the combination of our method with surrounding cues, like the object near the informative joints or the type of the environment.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (61201429, 61362030), Xinjiang Uygur Autonomous Regions Natural Science Foundation (201233146-6), Technology Research Project of the Ministry of Public Security of China (2014JSYJB007), Xinjiang Uygur Autonomous Regions University Scientific Research Key Project (XJEDU2012I08) and the 111 Project (B12018).

Furthermore, we are grateful to Lu Xia for her valuable help and enlightenment in this project.

References

[1] A. Gaidon, Z. Harchaoui, C. Schmid, Actom sequence models for efficient action detection, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3201–3208.
[2] M.A.R. Ahad, J.K. Tan, H. Kim, S. Ishikawa, Motion history image: its variants and applications, Mach. Vis. Appl. 23 (2) (2012) 255–281.
[3] T.-K. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Trans. Pattern Anal. Mach. Intell. 31 (8) (2009) 1415–1428.
[4] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local SVM approach, in: Proceedings of the 17th International Conference on Pattern Recognition (ICPR), vol. 3, 2004, pp. 32–36.
[5] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 2929–2936.
[6] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-time human pose recognition in parts from single depth images, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 1297–1304.
[7] A. Kurakin, Z. Zhang, Z. Liu, A real time system for dynamic hand gesture recognition with a depth sensor, in: The 20th European Signal Processing Conference (EUSIPCO), 2012, pp. 1975–1979.
[8] H. Benko, A. Wilson, Depthtouch: Using Depth-Sensing Camera to Enable Freehand Interactions on and Above the Interactive Surface, Technical Report MSR-TR-2009-23, March 2009.
[9] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, A. Blake, Efficient human pose estimation from single depth images, Pattern Anal. Mach. Intell. 35 (12) (2013) 2821–2840.
[10] L.A. Schwarz, D. Mateus, V. Castaeda, N. Navab, Manifold learning for ToF-based human body tracking and activity recognition, in: The 21st British Machine Vision Conference (BMVC), 2010, pp. 1–11.
[11] L. Xia, J.K. Aggarwal, Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera, in: The 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2834–2841.
[12] O. Oreifej, Z. Liu, HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences, in: The 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 716–723.
[13] W. Li, Z. Zhang, Z. Liu, Action recognition based on a bag of 3D points, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2010, pp. 9–14.
[14] R. Dubey, B. Ni, P. Moulin, A depth camera based fall recognition system for the elderly, in: The Ninth International Conference on Image Analysis and Recognition (ICIAR), vol. 7325, 2012, pp. 106–113.
[15] E. Ohn-Bar, M.M. Trivedi, Joint angles similarities and HOG2 for action recognition, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2013, pp. 465–470.
[16] X. Lu, C. Chia-Chih, 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 (CVPRW), 2012, pp. 20–27.
[17] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, R. Bajcsy, Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition, J. Vis. Commun. Image Represent. 25 (1) (2014) 24–38.
[18] G. Evangelidis, G. Singh, R. Horaud, Skeletal Quads: Human action recognition using joint quadruples, in: The 22nd International Conference on Pattern Recognition (ICPR), 2014, p. NA.
[19] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, R. Vidal, Bio-inspired dynamic 3D discriminative skeletal features for human action recognition, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2013, pp. 471–478.
[20] C. Plagemann, V. Ganapathi, D. Koller, S. Thrun, Real-time identification and localization of body parts from depth images, in: 2010 IEEE International Conference on Robotics and Automation (ICRA), 2010, pp. 3108–3113.
[21] B.J. Frey, D. Dueck, Clustering by passing messages between data points, Science 315 (5814) (2007) 972–976.
[22] S. Jaeyong, C. Ponce, B. Selman, A. Saxena, Unstructured human activity detection from RGBD images, in: 2012 IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 842–849.
[23] L. Shih-Yao, S. Chuen-Kai, C. Shen-Chi, L. Ming-Sui, H. Yi-Ping, Human action recognition using action trait code, in: The 21st International Conference on Pattern Recognition (ICPR), 2013, pp. 3456–3459.
[24] X. Yang, Y. Tian, Effective 3D action recognition using eigenjoints, J. Vis. Commun. Image Represent. 25 (1) (2014) 2–11.
[25] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, A. Del Bimbo, Space–time pose representation for 3D human action recognition, in: ICIAP Workshop on Social Behaviour Analysis, 2013, pp. 456–464.
[26] J. Wang, Z. Liu, Y. Wu, J. Yuan, Mining actionlet ensemble for action recognition with depth cameras, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 1290–1297.
[27] G. Johansson, Visual perception of biological motion and a model for its analysis, Percept. Psychophys. 14 (2) (1973) 201–211.
[28] R. Vemulapalli, F. Arrate, R. Chellappa, Human action recognition by representing 3d skeletons as points in a Lie group, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[29] Y. Zhu, W. Chen, G. Guo, Fusing spatiotemporal features and joints for 3D action recognition, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2013, pp. 486–491.