J. Vis. Commun. Image R. 35 (2016) 103–111

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

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

Learning hierarchical spatio-temporal pattern for human activity prediction ☆

Wenwen Ding, Kai Liu *, Fei Cheng, Jin Zhang

School of Computer Science and Technology, Xidian University, Xi'an, China

ARTICLE INFO

Article history:
Received 7 September 2015
Accepted 9 December 2015
Available online 18 December 2015

Keywords:
Skeleton joints
3D action feature representation
Self-organizing map
Hebbian learning
Variable order Markov model
Probabilistic suffix tree
RGB-D dataset
3D trajectory segmentation

ABSTRACT

Human activity prediction has become increasingly valuable in many applications. This paper, initially from the perspective of cognition science, presents a novel approach to learning a hierarchical spatio-temporal pattern of human activities to predict ongoing activities from videos that contain only the onsets of the activities. Spatio-temporal pattern can be learned by a Hierarchical Self-Organizing Map (HSOM), which consists of two self-organizing maps (i.e., action map and actionlet map) connected via associative links trained by Hebbian learning. Ongoing activities can be predicted by Variable order Markov Model (VMM), which provides the means for capturing both large and small order Markov dependencies based on the training actionlet sequences. Experiments of the proposed method on four challenging 3D action datasets captured by commodity depth cameras show promising results.

1. Introduction

Human action recognition, the automatic recognition of ongoing actions performed by humans, is an active research topic in computer vision. It has a variety of real-world applications, including video surveillance, video retrieval, and even health-care. Over the past few decades, research has primarily concentrated on processes of learning and recognizing actions from video sequences [1,2]. In contrast, less attention has been paid to early detection of unfinished activities from video streams where early prediction of ongoing activity is extremely valuable [3]. For instance, in a supermarket, it would be beneficial to equip a surveillance system that can provide real-time surveillance, detect suspicious activities, and raise the alarm for theft before it happens.

Neurobiological studies [4] have concluded that the human brain can perceive actions by observing only a few actionlets[1] and component action units obtained from temporal decomposition during action execution. In the neuropsychological perspective, Friston [7] has linked the hierarchical theory of action with the organization of the brain and also described how action representations are selected, maintained, and inhibited at multiple levels of abstraction and how layers are mediated by effective connectivity.

In this vein, this paper uses a Hierarchical Self-Organizing Map (HSOM) [8] to generate model whose structure can capture the natural hierarchy which can be layered as actionlet and action from a small granularity to a large, thus make it easier to comprehension and decomposition activities at varying levels of abstraction present in human activity. Furthermore, a worthwhile approach is proposed to describe actions as sequences of consecutive actionlets and recognize action depend on a little actionlets extracted from the beginning of this action.

This paper proposes a novel framework, shown in Fig. 1, for human activity recognition from partially observed videos that use sequences of 3D skeleton joint positions as input. To obtain meaningful action units, we first learn superior segmentation points $\mathcal{S} = \{s_1, \dots s_i, \dots, s_j, \dots, s_m\}(1 < i < j < m)$ to segment 3D trajectory of an action, as shown in Fig. 2a. Then decompose action, using motion velocities, the direction of motion, and the curvatures of trajectories, into a sequence of actionlets, as shown in Fig. 2b. The detailed process can be viewed clearly in our previous work [9]. The features of actions ξ_a and actionlets ξ_{al} are extracted from these segmented trajectories. Two Self-Organizing Maps (SOMs)

☆ This paper has been recommended for acceptance by M.T. Sun.

* Corresponding author.

E-mail addresses: dww2048@163.com (W. Ding), kailiu@mail.xidian.edu.cn (K. Liu), chengfei8582@163.com (F. Cheng), jinzhang.cv@gmail.com (J. Zhang).

[1] In this paper, we use actionlet to refer to meaningful atomic actions obtained from spatio-temporal decomposition using motion velocities, the direction of motion, and the curvatures of trajectories. It should be noted that the same term *actionlet* has also been used in the recent work, which refers to action components based on a spatial segmentation [5] and component action units obtained from temporal decomposition [6].

http://dx.doi.org/10.1016/j.jvcir.2015.12.006
1047-3203/© 2015 Elsevier Inc. All rights reserved.

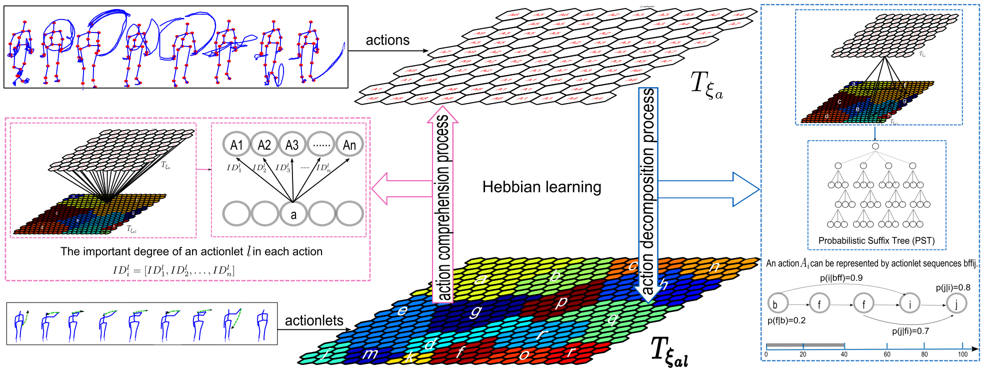

actions
T_{ξ_a}
action comprehension process
Hebbian learning
action decomposition process
A1
A2
A3
......
An
ID^l_1
ID^l_2
ID^l_3
ID^l_n
a
The important degree of an actionlet l in each action
$ID^l_i = [ID^l_1, ID^l_2, \ldots, ID^l_n]$
actionlets
$T_{\xi_{al}}$
a
b
c
e
g
p
h
n
q
r
d
f
o
j
m
k
Probabilistic Suffix Tree (PST)
An action A_i can be represented by actionlet sequences bffij.
p(i|bff)=0.9
p(j|i)=0.8
p(f|b)=0.2
p(j|fi)=0.7
b
f
f
i
j
0
20
40
60
80
100

Fig. 1. The general framework of the proposed approach.

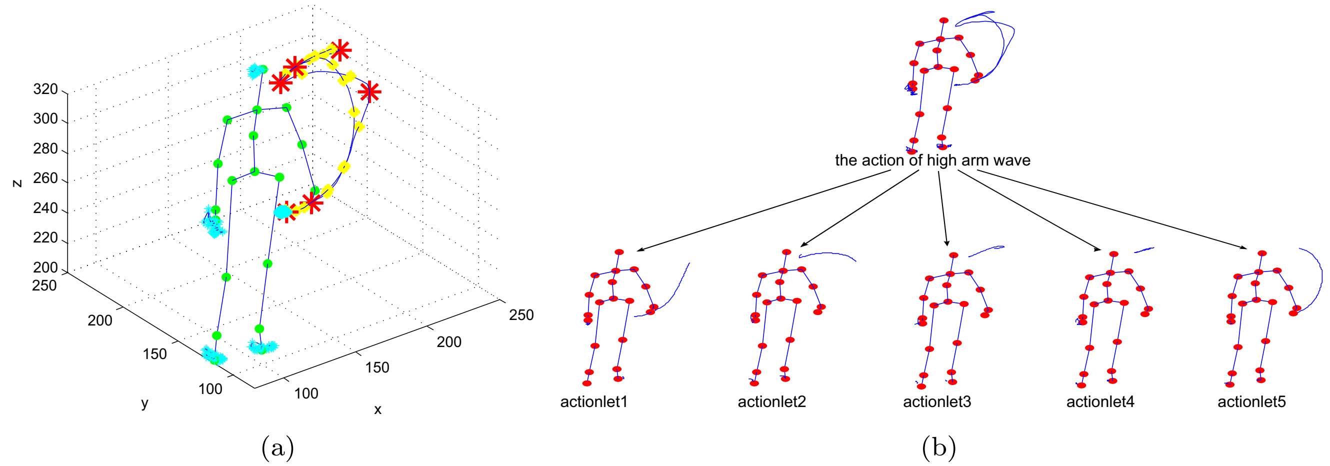


Fig. 2. (a) The trajectory of an action of *high hand wave* is segmented by *red stars*. (b) The action *high hand wave* can be decomposed by five actionlets through motion velocities, the direction of motion, and the curvatures of trajectories. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

constitute the Hierarchical SOM. ξ_a is mapped to one SOM as T_{ξ_a}, and ξ_{al} is mapped to the other SOM as $T_{\xi_{al}}$. Thus, numerically similar adjacent features of actions and actionlets can be mapped to a single representative vector $\boldsymbol{m}$ (model vector) on a SOM, which itself is a form of clustering process. Unlike actions that have labels (for example, high hand wave) that can be acquired from human natural language, actionlets do not have such labels. Therefore, $T_{\xi_{al}}$ can be scattered in plots named with English alphabets referred as the labels of actionlets according to the Davies–Bouldin index value [10], which is a good candidate for map unit clusters. The associative weights between T_{ξ_a} and $T_{\xi_{al}}$ can be obtained through Hebbian learning [11], which can measure the important degree (ID) of an actionlet in each action. Thus, complex actions can be represented by sequences of the English letters, which are seen as context. With the help of the context and a Variable order Markov Model (VMM) [12], the probability of the next possible actionlet or the whole action can be predicted.

From the spatio-temporal perspective, an action can be characterized by the spatio-temporal information of actionlets, which can be effectively used in the learning and predicting processes. Suppose an action and actionlet in context are regarded as a word and a letter, respectively. These processes can be described as person A predicting the meaning of a sentence written by person B. For example, *eat apple* and *eat banana* represent two different meaning of English words. When person B is writing, letters are shown one by one, such as *eat na*. The word *eat banana* can be predicted by person A because the ID of letters *a*, *n* and *e* is high in word *eat banana* (especially *a*), and the causality between *na* and *eat banana* exists. However, if the next letter extracted is *p*, the sequence of letters becomes *eat nap*. Thus, the word *eat apple* may be predicted because the ID of letters *a* and *p* is not low in the word *eat apple* and *ap* has more direct causality with *eat apple* than *eat banana*.

The major contributions in this paper include: (1) HSOM is proposed to systematically exhibit the intrinsic hierarchical structure of human activity accordance with human cognition and perception from global to local as well as coarse to fine, thus making it easier to comprehension and decomposition activities at varying levels of abstraction present in human activity; and (2) Hebbian learning between actions and actionlets is modeled that allows for the representation of the important degree of actionlets in each action.

The rest of the paper is organized as follows, Section 2 presents the related work; Section 3 elaborates on the proposed method of action and actionlet representation, mapping, clustering, learning and prediction; Section 4 presents our experimental results and discussion; and Section 5 concludes this paper.

2. Related work

Action recognition. There has been tremendous amount of work on human action recognition from static images and 2D video sequences. Wang et al. [13] learns multiple features from a small number of labeled videos, and automatically utilizes data distributions between labeled and unlabeled data to boost the recognition performance. Sadanand and Corso [14] present the conception of

Action Bank, which comprised of many individual action detectors sampled broadly in semantic space as well as viewpoint space. Merler et al. [15] propose semantic model vectors extracted using a set of discriminative semantic classifiers, each being an ensemble of SVM models trained from thousands of labeled web images, for a total of 280 generic concepts.

With the development of the commodity depth sensors like Microsoft Kinect [16], there has been a lot of interests in human action recognition from depth data. Wang et al. [17] presents a new superpixel-based hand gesture recognition system based on a novel superpixel earth mover's distance metric, together with Kinect depth camera. Li et al. [18] employed a bag-of-3D-points graph approach to encode actions based on 3D projection of body silhouette points. Xia et al. [19] mapped 3D skeletal joints to a spherical coordinate system and used a histogram of 3D Joint Locations (HOJ3D) to achieve view-invariant posture representation. But these current algorithm only works well when the human subject is in an upright position facing the camera. The human body viewed partly will lead to that the results of experiment are not very reliable.

Action prediction. Most of the existing work in action prediction aims at recognizing unfinished action videos. The goal of human activity prediction was proposed by Ryoo [3]. He considered the unseen part of an event as a latent variable and used two bags of words to construct histograms for ongoing activity recognition. However, his approach failed to account for the sequential nature of temporal events. Cao et al. [20] extended Ryoo's work to recognize human activities from partially observed videos in the general case. Additionally, an early event detector Hoai and De la Torre [21] was proposed to augment the training set using partial events as positive examples, which is different from our goal. There is also some emerging researches [22–24,6] termed early recognition, where the task is to classify an incoming temporal sequence as early as possible while maintaining a level of detection accuracy. However, these methods have worked on human activity prediction more from static images and 2D video sequences than from depth data.

Recently, several works have presented various ways to handle activity recognition and prediction based on neural network. Martinez-Contreras et al. [25] describes a new method to deal with the temporal features needed for the detection of human actions using a SOM to model temporal templates in the lower dimensional space formed by the neurons whose characteristics are tracked in time by means of a HMM to carry out the action recognition process. Sumpter and Bulpitt [26] introduced feedback to the second competitive network to suggest a novel approach for learning long-term spatio-temporal patterns of objects in image sequences, using a neural network paradigm to predict future activities. Hu et al. [27] constructed activity patterns for anomaly detection and activity prediction using a fuzzy self-organizing neural network. Sun and Liu [28] introduced Recurrent Self Organizing Map (RSOM) trajectories to represent the ongoing human activities, and the Dynamic Time Warping-Edit (DTW-E) distance was specially proposed to measure the structural dissimilarity between RSOM trajectories. However, all these methods do not predict human activity from the view of the hierarchical theory of action and sub-action.

Unlike existing methods based on neural networks, our method segments sequences of 3D joint positions as an input to the neural network, not only learning an action on one SOM, but also learning an actionlet on the other SOM. This makes HSOM much easier to construct as the organization of brain through Hebbian learning and makes the learning process much more efficient. Therefore, this process let us to obtain an important prior knowledge that informative action information is increasing when new observations are available.

3. Proposed method

In this section, we present a HSOM model (two-layers) to imitate human learning and cognition about how an action is decomposed into actionlets and how to comprehend an action from actionlets point of view.

We will first extract the spatio-temporal feature ξ_a and ξ_{al} as input vectors to HSOM constituted by T_{ξ_a} and $T_{\xi_{al}}$. Next, we will learn two process of action decomposition and comprehension through Hebbian learning. Also, in this section, we will discuss uncertainties in action comprehension process through VMM. Finally, we will present a technique that models human cognition thereby creating information that facilitates the human prediction process.

3.1. Action and actionlet representation

To achieve efficient and effective prediction, a well-structured training and learning mechanism should be developed. For each frame t of a sequence, the real world 3D position of each joint of the skeleton is represented by three coordinates $x,\ y$, and z. Therefore, a joint point trajectory in 3D space can be parameterized as a matrix:

$$\boldsymbol{R}(t) = [\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3, \ldots, \boldsymbol{p}_t, \ldots, \boldsymbol{p}_n], \tag{1}$$

where $\boldsymbol{p}_t = [x_t, y_t, z_t, \upsilon_t, \Delta h_t, \Delta\phi_t, \Delta\theta_t, k_t, d_t]^T;\ \{x_t, y_t, z_t\}$ are the coordinates of a joint point at frame $t;\ \upsilon_t$ is the motion velocity of the joint point at frame $t;\ \Delta h_t,\ \Delta\phi_t,\ \Delta\theta_t$ are the directions of up–down, left–right and further-closer information of a joint point at frame $t;\ k_t$ is the curvature of a joint point at frame $t;\ d_t$ is the distance between the hip center and the joint point; n is the number of frames (trajectory length) and t is the time-stamp index.

Five trajectories of head, left hand, right hand, left foot, and right foot are required to analyze an action. The trajectory having a maximum velocity shows the distinguishing characteristics of an action. For example, a *high hand wave* action captures a variety of motions only related to the right or left arm. Therefore, segmentation points can be extracted from this trajectory having a maximum velocity. To obtain superior segmentation points s_i, we use the value of curvature k, as this point is closer to the hilltop, whereas the value of $\Delta h,\ \Delta\theta$ or $\Delta\phi$ at this point are zero crossing. The actionlet, derived from the set of segmentation points $S = \{s_1, \ldots, s_i, \ldots, s_j, \ldots, s_m\}$, is a meaningful atomic action. Hence, an action can be decomposed into a sequence of actionlets. Therefore, the feature of an actionlet can be parameterized as follows:

$$\xi_{al}^T = \left[\boldsymbol{R}_1(s_i : s_j), \ldots, \boldsymbol{R}_k(s_i : s_j), \ldots, \boldsymbol{R}_5(s_i : s_j)\right], \tag{2}$$

where $\boldsymbol{R}_k(s_i : s_j) = [\boldsymbol{p}_{s_i}, \ldots, \boldsymbol{p}_{s_j}]$, with $k = 1, 2, 3, 4, 5$ denoting head, left hand, right hand, left foot, and right foot, respectively.

In this paper, an action is referred to as a single period of a human motion pattern and the periodic motion in multi-period action video are eliminated by methods as shown in [9]. Therefore, the feature of an action can be parameterized as follows:

$$\xi_a^T = \left[\xi_{al_1}, \xi_{al_2}, \ldots, \xi_{al_n}\right], \tag{3}$$

where $\xi_{al_1}, \xi_{al_2}, \ldots, \xi_{al_n}$ are aperiodic sequences.

3.2. Action and actionlet map construction

There are large intra-class variations in the human actions. Differ in thousands ways of people's movements, whatever select any one as template is not suitable. Therefore, ξ_a and ξ_{al} will be clustered to find the pattern of action and actionlet via unsupervised learning rather than supervised learning.

A SOM [8] is a type of artificial neural network for the visualization of high-dimensional data using unsupervised learning. It also produces some kind of abstractions of high-dimensional data. The map is defined as a grid array whose neurons are uniformly arranged, and each neurons of the i-th location is associated with a parametric real vector $\boldsymbol{m}_i = [m_{i1}, m_{i2}, \ldots, m_{in}]^T \in \mathcal{R}^n$, which is called a model vector. The model vectors m of neurons are initialized either to small random values or sampled evenly from the subspace spanned by the two largest principal component eigenvectors. The input vector, ξ_a and ξ_{al}, has the same dimension with the model vector $\boldsymbol{m}$.

The training process for constructing T_{ξ_a} is based on three procedures.

(1) Competition: The winning neuron c is determined by the node location whose model vector $\mathbf{m}_i$ has the minimum Euclidean distance to the input vector ξ_a as

$$c = \min_i \|\xi_a(t) - \mathbf{m}_i(t)\|, \tag{4}$$

where t = 0, 1, 2, ... is the index of iteration step, $\|\cdot\|$ denotes the Euclidean distance.

(2) Cooperation: The neighborhood function h is a decreasing function of the distance between the i-th and c-th nodes as a smoothing kernel. A typical choice is often taken to be the Gaussian

$$h_{c(x),i} = \alpha(t) \exp\left(-\frac{\|r_i - r_c\|^2}{2\sigma^2(t)}\right), \tag{5}$$

where $0 < \alpha(t) < 1$ is the learning rate factor, $r_i \in \mathcal{R}^2$ and $r_c \in \mathcal{R}^2$ are the locations in the display grid. $\sigma(t)$ is the "effective width" of the topological neighborhood.

(3) Adaptation: The update process for each neuron is adapted with respect to its lateral distance from the wining neuron as follow:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{c(x),i}(\xi_a(t) - \mathbf{m}_i(t)). \tag{6}$$

The training process for constructing $T_{\xi_{al}}$ is also based on three procedures above.

After the training process for constructing SOM, the neurons in SOM can help us to understand ξ_a and ξ_{al} as its topological structures also denote the characteristics of ξ_a and ξ_{al}. In our system, two SOMs are needed to construct the HSOM, which shows hierarchical relations in human activity. The first SOM maps ξ_a to T_{ξ_a} and the second SOM maps ξ_{al} to $T_{\xi_{al}}$.

3.3. Clustering of actionlets

Action labels are easily labeled in real life, such as walk, sit down, stand up, and throw. In our experiment, each action was labeled as $A_1, A_2, \ldots, A_n$ to facilitate quantitative analysis of the map and the data. Unlike actions with labels that are shown on a map grid, an actionlet is hardly labeled or highly generalized using our human language. Therefore, similar neural units in $T_{\xi_{al}}$ need to be grouped and labeled later. To find initial partitioning, we use the Davies–Bouldin index value [10], which is a metric for evaluating map unit clusters to scattered the $T_{\xi_{al}}$ in plots with actionlet symbolized by English letters. This process may be represented in Fig. 3.

The Davies Bouldin Index is defined as the ratio of S_c and d_{ce} and the best clustering minimizes

$$DBI = \frac{1}{C}\sum_{k=1}^{c}\max_{l\neq k}\left\{\frac{S_c(Q_k) + S_c(Q_l)}{d_{ce}(Q_k, Q_l)}\right\}, \tag{7}$$

where C is the number of clusters, $\{Q_i | i = 1, \ldots, C\}$ is a set of clusters, $S_c = \frac{\sum_i \|x_i - c_k\|}{N_k}$ is the within-cluster distance, $d_{ce} = \|c_k - c_l\|$ is the between-clusters distance, $x_i \in Q_i,\ N_k$ is the number of samples in cluster Q and $c_k = \frac{1}{N_k}\sum_{x_i \in Q_k} x_i$. By definition, the lower the DBI, the better the separation of the clusters and the tightness inside the clusters.

Let $\mathcal{L}$ be a finite set of English letters discussed above, as well as labels of actionlets. Thus, after partitioning and labeling to $T_{\xi_{al}}$, each action can be represented by a sequence of actionlet symbols from $\mathcal{L}$. For example, one sample vector ξ_a, labeled A_1, can be represented as actionlet symbols through the segmentation of the action. After the completion of training, this kind of action will correspond to training data sequence *r = #diik#ik#dik#diik#ibk#iik#dkkf#dkik#dkk#didik*, which means the action has ten samples separated by # and each sample can be represented as a sequence of actionlet symbols.

3.4. Hebbian learning between two SOMs

The Canadian Psychologist Donald Hebb [11] speculated in 1949 that "When neuron A repeatedly and persistently takes part in exciting neuron B, the synaptic connection from A to B will be strengthened." Simultaneous activation of neurons leads to pronounced increases in synaptic strength between them. Thus Hebb's principle can be described as a method for determining how to alter the weights between two patterns of T_{ξ_a} and $T_{\xi_{al}}$, their rate of co-occurrence and the strength of their co-activated in the representation. When an action activates T_{ξ_a}, the associative weights can be used to project to several neurons onto $T_{\xi_{al}}$(action decomposition process). This spurs a production such that actions having the same semantic meaning possess several identical or similar actionlets, in general. For example, an action of *wave hands* can be represented as actionlet sequences with *bffij*, *bfifj*, and *bifj*, as shown in Fig. 4. Conversely, when several actionlets activate $T_{\xi_{al}}$, the associative weights can be used to project a neuron onto T_{ξ_a}(action comprehension process), spurring a composition such that these actionlets correspond to an action.

Formally, T_{ξ_a} is defined as a graph $G(A, E)$, where A is a set of neurons, and $E \subset A \times A$ is a set of connections between the neu-

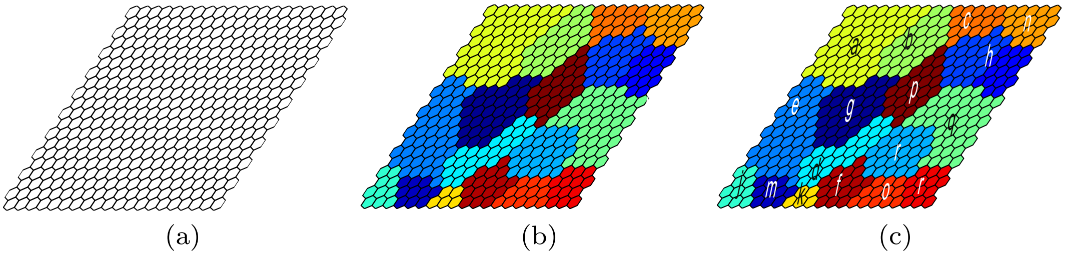
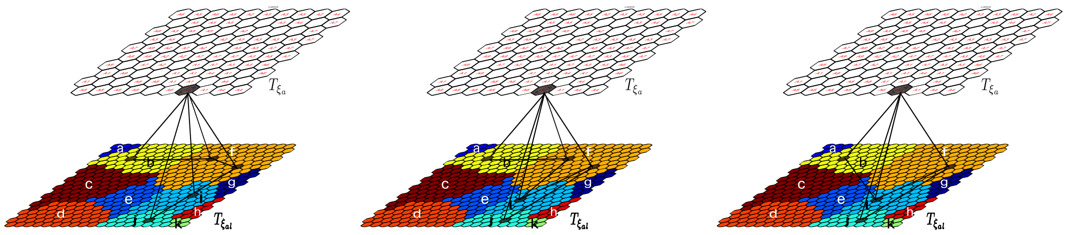
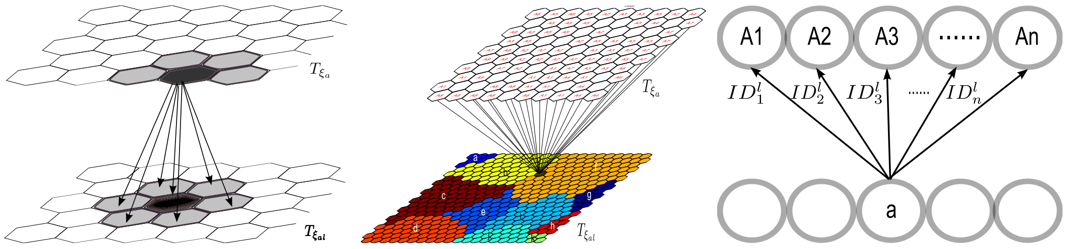


Fig. 3. (a) The input vectors ξ_{al} are first clustered using the SOM to produce the model vectors $\boldsymbol{m}$. (b) $T_{\xi_{al}}$ is clustered again to organize the model vectors into "natural groups". (c) Different English letters are represented by different clusters in $T_{\xi_{al}}$.



(a) (b) (c)

Fig. 4. An action of wave hands can be represented as actionlet sequences with *bffij*, *bfifj*, and *bifj*, as shown in this figure (a), (b), and (c), respectively. Notably, actions with similar feature vectors may map to the same neuron. For example, the feature vectors of *high wave hands* and *horizontal wave hands* can map to the same neuron. In other words, one neuron can be mapped to only one action model vector, or some action model vector can map to the same neuron.



(a) (b) (c)

Fig. 5. (a) Dark neurons are the winning neurons and light gray neurons are the neighborhood neurons. The winning neuron and its neighbors in region N_c are activated to different extents, while neurons outside N_c are retained. (b) Each neuron in $T_{\xi_{al}}$ has associative links with every neuron in T_{ξ_a}. Some weights are smaller and some weights are bigger according to the Hebbian learning results. (c) These weights on associative links are classified in terms of the label of action. The important degree of an actionlet can be obtained from the sum of these classified weights.

rons. Each neuron k in a T_{ξ_a} has an associated model vector $\mathbf{m}_k$. Given an input vector ξ_a, the localized output response a_k of a neuron k in T_{ξ_a} is computed as

$$a_k = \begin{cases} 1 - \frac{\|\xi_a - \mathbf{m}_k\| - d_{min}}{d_{max} - d_{min}}, & k \in Nc \\ 0, & \text{otherwise} \end{cases}, \tag{8}$$

where N_c is the set of neighbors of winner c, as shown in Fig. 5a; and d_{min} and d_{max} are the smallest and the largest Euclidean distances of ξ_a to node's weight vectors, respectively, within N_c. Given an input vector ξ_{al}, the localized output response a_l of a neuron l in $T_{\xi_{al}}$ may also be computed as

$$a_l = \begin{cases} 1 - \frac{\|\xi_{al} - \mathbf{m}_l\| - d_{min}}{d_{max} - d_{min}}, & l \in Nc \\ 0, & \text{otherwise} \end{cases}. \tag{9}$$

The HSOM are bi-directionally linked with associative connections. Simultaneously with input vectors, the associative weights between the active neurons in $T_{\xi_{al}}$ and T_{ξ_a} are updated using Hebbian learning

$$\Delta w_{kl} = \alpha(t) a_k a_l, \tag{10}$$

where w_{kl} is the unidirectional associative weight leading from node k in T_{ξ_a} to node l in T_{ξ_a}, $\alpha(t)$ is a learning rate.

The associative weight vectors can be normalized by:

$$w_{kl}(t+1) = \frac{w_{kl}(t) + \Delta w_{kl}}{\left\{\sum_l [w_{kl}(t) + \Delta w_{kl}]^2\right\}^{1/2}}. \tag{11}$$

If the size of the grid arrays of T_{ξ_a} and $T_{\xi_{al}}$ are $s_1 = m_1 \times n_1$ and $s_2 = m_2 \times n_2$, respectively, the associative weights between them can be seen as a weight matrix $\boldsymbol{W}_{s_1 \times s_2} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{s_2})$, as depicted in Fig. 5b. The important degree of an actionlet l in each action, the spatial character of spatio-temporal patterns, can be computed as

$$\boldsymbol{ID}^l = \boldsymbol{w}_l^T \boldsymbol{K}, \tag{12}$$

where $\boldsymbol{ID}^l$ is an n-dimension vector, n is the number of prediction action needed, $\boldsymbol{w}_l$ is the l-th column of the weight matrix $\boldsymbol{W}$, as illustrated as Fig. 5c and $\boldsymbol{K}$ is a matrix representing the amount of input vectors with the same label that has been mapped to one neuron in T_{ξ_a}.

3.5. Probabilistic suffix tree

VMM, according to the correlation between the characteristic of a simple symbolic sequence, provides the means for capturing both large and small order Markov dependencies based on the training data sequence r. The main objective of VMM is to predict the next possible actionlet or the whole activity. In this paper, the data sequence r of every kind action will correspond to a Probabilistic Suffix Tree (PST) [29]. PST, which is used to construct the D-bounded VMM, also provides a conditional probability distribution $p(l_{i+1}|l_1 l_2 \ldots l_i)$ for an ongoing actionlet sequence $l_1 l_2 \ldots l_i$ over $\mathcal{L}$, where l_i is the label of an actionlet.

A PST uses a suffix tree for the storage structure, where the degree of each node D varies between zero and $\|\mathcal{L}\|$. The label of a parent node is a suffix of labels of its children nodes, which induces a "suffix set" S consisting of the labels of all nodes. Each node also corresponds to a probability vector $\mathbf{v}$, which stores the conditional probability of the next symbol of the symbolic sequences in this node. The goal of the PST learning algorithm is to assign a conditional probability distribution $P(\sigma|s)$ over $\mathcal{L}$ to associate a meaningful context $s \in S$ with the next alphabet, where the next possible actionlet $\sigma \in \mathcal{L}$.

The algorithm for constructing a PST consists of three stages: first, a set of "meaningful" contexts s is extracted from the training data sequence r, which forms a "suffix set" S; second, a suffix tree is built by excluding nodes that do not provide stochastic information; third and finally, we smooth the probability distributions associated with the tree nodes and normalize the resulting conditional distribution. Fig. 6 shows an example PST constructed from a training sequence of actionlets.

3.6. Action prediction

The key contribution of this work is the idea that the predictability can be ensured by the important degree of actionlets in each action acquired through Hebbian learning between actions and actionlets, while causality of actionlets can be encoded as a PST with variable temporal scale.

Given an ongoing actionlet sequence $l = l_1 l_2 \dots l_i$, we can now construct our prediction function by using the result of learning from the important degree of actionlets and causality of actionlets:

$$p^c(l) = \sum_{j=1}^{\|l\|} \boldsymbol{ID}_c^{l_j} P(l_j | l_1 l_2 \dots l_{j-1}). \tag{13}$$

Formula (13) ultimately predicts which kind of activity class $c(c = 1, \dots, C)$ the sequence belongs to. The prediction model $p^c(l)$ is computed over the ongoing actionlet sequence $l = l_1 l_2 \dots l_i$ of this class c belonging to the training set. The prediction result c_0 has a maximal prediction score $p^c(l)$

$$c_0 = \text{argmax}_c \{p^c(l), c = 1, \dots, C\}. \tag{14}$$

Thus, giving partially observed a video, the probability of the next possible actionlet or the whole action can be predicted from the prediction model $p^c(l)$ for which maximal prediction score has been obtained.

4. Experimental results

The performance of the activity prediction is primarily evaluated based on its accuracy, that is, the percentage of actions recognized correctly. We chose MSR-Action3D [18], 3D Online Action Dataset [30], UTKinect-Action [19], and UCF Kinect Dataset [31] for evaluating the early activity prediction accuracy and the full recognition accuracy.

4.1. Discussion on parameters setting

Similar to other action recognition methods, our solution depends on two important types of parameters, the sizes of the grid arrays of T_{ξ_a} and $T_{\xi_{al}}$ and the number of clusters of $T_{\xi_{al}}$.

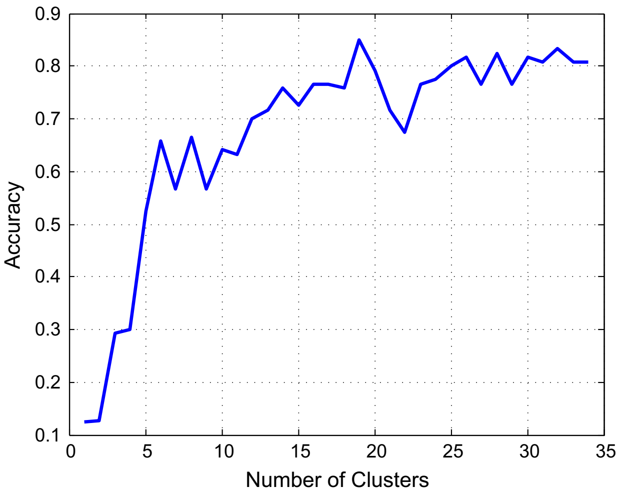


Fig. 7. Early action recognition results with the number of clusters $T_{\xi_{al}}$ from 1 to 35.

The first type of parameter refers to the sizes of the grid arrays of T_{ξ_a} and $T_{\xi_{al}}$, s_1 and s_2. The larger the size of grid array, the more easily the over-fitting phenomenon occurs, and conversely, the smaller the size of grid array, the more easily the under-fitting phenomenon occurs. In the SOM Toolbox [32], the default number of neurons is $5\sqrt{n}$ where n is the number of training samples. Hence, this parameter could be set in terms of the rule given in our paper.

The second important parameter $\|\mathcal{L}\|$ is the number of clusters of $T_{\xi_{al}}$, which determines the upper bound D on the Markov order. This parameter directly affects the early activity recognition rate as suggested by Fig. 7. From this figure, we find that the smaller the number of clusters are, the lower the early action recognition accuracy would be.

4.2. MSR Action3D Dataset

The MSR Action3D Dataset aims at providing 3D data extracted from the depth sequence. The MSR Action3D is a set of temporally segmented actions that have been pre-processed to remove the background. The dataset consists of the following 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 *pick up and throw*. These actions performed by

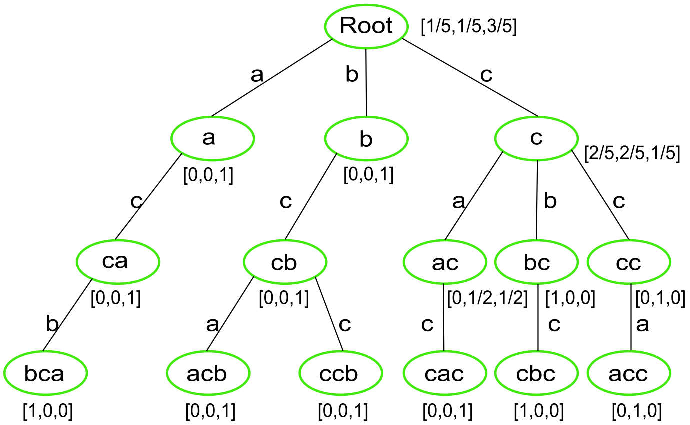


Fig. 6. An example PST corresponding to depth $D = 3$ and training sequence $r = cacbcaccbc$ over alphabet $\sum = \{a, b, c\}$. The vector of nodes c of the first layer is $[\frac{2}{5}, \frac{2}{5}, \frac{1}{5}]$, indicating the conditional possibilities of the next potential alphabet of $a,\ b$, and c are $p(a|c) = \frac{2}{5},\ p(b|c) = \frac{2}{5}$, and $p(c|c) = \frac{1}{5}$, respectively, while the symbol c appears, where $p(a|c) = \frac{|ac|}{|a^*|}$.

10 subjects facing the camera during performance. Ten subjects performed each action three times each facing the camera. Actors were requested to use their right arm or leg, when only one arm or one leg is involved in the action. The frame rate was 15 frames per second. The depth maps resolution was 320 × 240. This dataset is a challenging one due to the noise in the extracted skeletons.

In order to allow a fair comparison with the state of the art methods, the 20 actions were divided into three subsets as shown in Table 1 in accordance with the same experimental settings as in [18], where the samples of half of the subjects are used as training data, and the rest of the samples are used as testing data. In this paper, this kind experimental setting go by the name of cross-subject test setting. We compared our approach on this setting to some methods using skeleton inputs extracted from video streams, as shown in Table 2, that our approach clearly outperforms other methods.

Table 1
The three subsets of actions used in the experiments.

AS1	AS2	AS3
Horizontal Wave	High Wave	High Throw
Hammer	Hand Catch	Forward Kick
Forward Punch	Draw X	Side Kick
High Throw	Draw Tick	Jogging
Hand Clap	Draw Circle	Tennis Swing
Bend	Hands Wave	Tennis Serve
Tennis Serve	Forward Kick	Golf Swing
Pickup Throw	Side Boxing	Pickup Throw

Table 2
Recognition rate (%) of MSR-Action3D dataset.

Method	Accuracy
Dynamic Temporal Warping [33]	54.0
Hidden Markov Model [34]	63.0
Bag of 3D Points [18]	74.7
Histogram of 3D Joints [19]	78.9
Eigenjoints [35]	82.3
Proposed method	88.6

Table 3
Early action recognition results with the action progress a progress step of 20% on three sub-datasets of MSR-Action3D dataset.

Dataset	20%	40%	60%	80%	100%
AS1	64.1	75.0	87.5	88.3	89.1
AS2	60.0	71.6	81.6	82.5	82.5
AS3	77.5	80.8	92.5	93.3	94.1
Overall	67.2	75.8	87.2	88.0	88.6

Samples in MSR Action3D Dataset are about simple actions, such as *high arm wave*, *bend*, and *pick up and throw*. These type of action usually consists of 3–5 actionlets. Then we fit the prediction task into the context of supervised classification problem. To train a prediction model, we construct an order 6-bounded PST and compute important degrees of actionlet in each action respectively. To evaluate the prediction accuracy, we use the cross-subject test setting. For the result of Table 3, the recognition rate does not significantly increase after 40% of the action progress. This is mainly because most of the discriminant information is contained in the beginning of the action for this dataset. The confusion matrix for the cross-subject-test is illustrated in Fig. 8. Prediction errors occur if two actions are highly similar to each other, such as *forward punch* and *high throw* in AS1. In AS2, *draw circle* is repeatedly confused with another action, perhaps because the curve of a circle in 3D space is segmented into several parts and is not a complete curve.

4.3. 3D Online Action Dataset

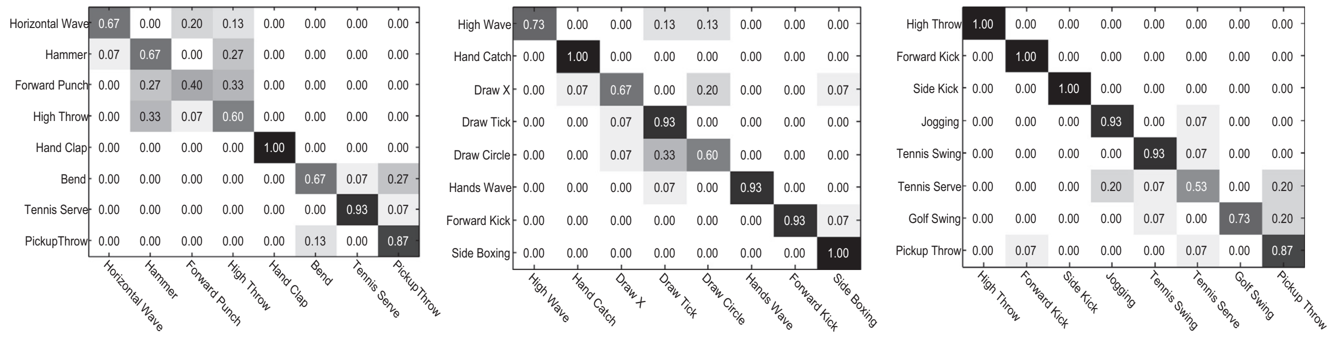
This dataset is for continuous online human action (human-object interaction) recognition from RGBD data created by Microsoft Research in 2014. There are seven action categories: Drinking, eating, using laptop, reading cellphone, making phone call, reading book, using remote. The dataset is designed for four evaluation tasks: (1) same-environment action recognition; (2) cross-environment action recognition; (3) action prediction on segmented videos; (4) continuous action recognition. We evaluate the first three tasks follow the setting in [30]. The recognition results of the first two tasks are shown in Table 4. The results of same-environment and cross-environment are almost the same, which illustrate that our algorithm is not affected by the environment. Action prediction on segmented video sequences is tested to evaluate the latency of our algorithm as shown in Fig. 9.

4.4. UTKinect-Action Dataset

UTKinect-Action Dataset was captured as part of research on action recognition using a single stationary Kinect. In this dataset, there are 10 actions types: walk, sit down, stand up, pick up, carry,

Table 4
Recognition rate (%) of 3D Online Action Dataset.

Method	Same-environment	Cross-environment
Moving Pose [36]	38.4	28.5
Eigenjoints [35]	49.1	35.7
Proposed method	49.5	50.9



	Horizontal Wave	Hammer	Forward Punch	High Throw	Hand Clap	Bend	Tennis Serve	PickupThrow
Horizontal Wave	0.67	0.00	0.20	0.13	0.00	0.00	0.00	0.00
Hammer	0.07	0.67	0.00	0.27	0.00	0.00	0.00	0.00
Forward Punch	0.00	0.27	0.40	0.33	0.00	0.00	0.00	0.00
High Throw	0.00	0.33	0.07	0.60	0.00	0.00	0.00	0.00
Hand Clap	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
Bend	0.00	0.00	0.00	0.00	0.00	0.67	0.07	0.27
Tennis Serve	0.00	0.00	0.00	0.00	0.00	0.00	0.93	0.07
PickupThrow	0.00	0.00	0.00	0.00	0.00	0.13	0.00	0.87

(a) AS1

	High Wave	Hand Catch	Draw X	Draw Tick	Draw Circle	Hands Wave	Forward Kick	Side Boxing
High Wave	0.73	0.00	0.00	0.13	0.13	0.00	0.00	0.00
Hand Catch	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Draw X	0.00	0.07	0.67	0.00	0.20	0.00	0.00	0.07
Draw Tick	0.00	0.00	0.07	0.93	0.00	0.00	0.00	0.00
Draw Circle	0.00	0.00	0.07	0.33	0.60	0.00	0.00	0.00
Hands Wave	0.00	0.00	0.00	0.07	0.00	0.93	0.00	0.00
Forward Kick	0.00	0.00	0.00	0.00	0.00	0.00	0.93	0.07
Side Boxing	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

(b) AS2

	High Throw	Forward Kick	Side Kick	Jogging	Tennis Swing	Tennis Serve	Golf Swing	Pickup Throw
High Throw	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Forward Kick	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Side Kick	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Jogging	0.00	0.00	0.00	0.93	0.00	0.07	0.00	0.00
Tennis Swing	0.00	0.00	0.00	0.00	0.93	0.07	0.00	0.00
Tennis Serve	0.00	0.00	0.00	0.20	0.07	0.53	0.00	0.20
Golf Swing	0.00	0.00	0.00	0.00	0.07	0.00	0.73	0.20
Pickup Throw	0.00	0.07	0.00	0.00	0.00	0.07	0.00	0.87

(c) AS3

Fig. 8. Confusion matrix in AS1, AS2 and AS3 at 40% observation ratio.

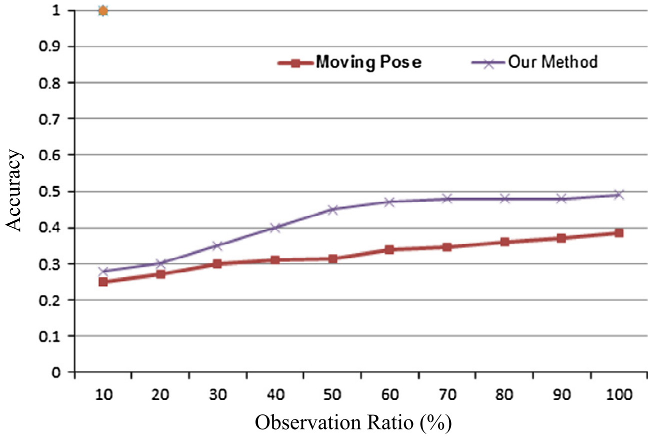


Fig. 9. Comparison of action prediction results with Moving Pose [36].

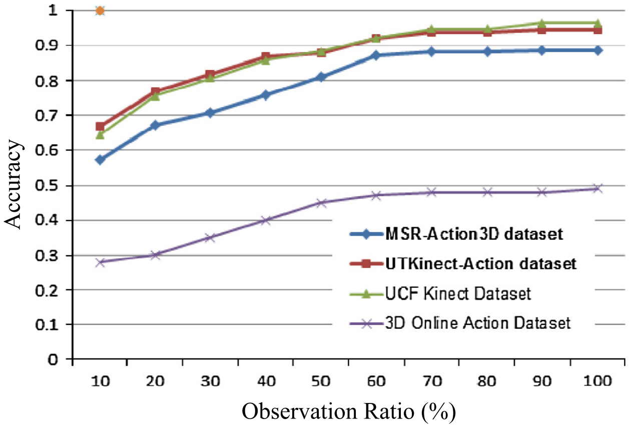


Fig. 10. Early action recognition results with the progress ranging from 10% to 100% and a progress step of 10%.

throw, push, pull, wave hands, and clap hands. Three channels were recorded: RGB, depth and skeleton joint locations. The three channel are synchronized. The frame rate is 30 f/s. Ten subjects perform 10 different actions twice, making up 200 sequences containing 6220 frames. The resolution of the depth map is 320 × 240 and the resolution of the RGB image is 640 × 480. To allow for comparison with [19], we followed the same experimental set up using Leave One Sequence Out Cross Validation (LOOCV) on the 200 sequences. For each test, one sequence was used for testing and the other 199 sequences were used for training. We also computed the mean accuracy obtained for each action separately, as shown in Table 5.

4.5. UCF Kinect Dataset

In order to confirm the effectiveness of our approach, we also evaluate the proposed method on a third dataset: UCF Kinect Dataset. The dataset was collected by Microsoft Kinect and OpenNI platform. Ellis [31] presented the Latency Aware Learning (LAL) algorithm for reducing the latency in recognizing the action. In each frame only 15 skeleton joints, orientation and binary confidence values of each joint are available, RGB images and depth maps are not stored. Each video in the dataset consists of one person performing one action. By comparing our method with methods of Eiengjoints [35] and STFC [9], as shown in Table 6, the recognition accuracies for our method is inferior to their approach. The UCF Kinect dataset includes 16 actions performed by 16 subjects. Each subject performs all 16 actions 5 times for a total of 1280 action samples. The number of action samples (1280) is much more than other dataset such as MSR Action3D Dataset (240 action samples) and UTKinect-Action Dataset (200 action samples). Our experiment to UCF Kinect dataset is implemented using the protocol of [31] which is the 4-fold cross-validation. If each action has 6 actionlet, there are 960 action samples and 5760 actionlet samples which are used for training on two SOM T_{ξ_a} and $T_{\xi_{al}}$ respectively. So the sizes of the grid arrays of T_{ξ_a} and $T_{\xi_{al}}$ will be larger. As discussed in Section 4.1, the larger the size of grid array, the more easily the over-fitting phenomenon occurs. Therefore, this is maybe a reason that the clustering performance of our proposed method on UCF Kinect dataset is inferior to Eigenjoint and STFC methods.

Table 5
UTKinect dataset: Recognition rate (%) of each action type.

Method	HO3DJ [19]	STFC [9]	Ours
Accuracy	90.9	91.5	94.5

Table 6
Comparisons of recognition accuracies (%) of LAL, Eigenjoint and our method.

Method	LAL [31]	Eigenjoint [35]	STFC [9]	Ours
Accuracy	95.9	97.1	98.04	96.5

As our solution has the potential to recognize actions before their completion, we present here the obtained results for early recognition. Fig. 10 shows detailed performance of our approach over all datasets, where the progress of activities ranges from 10% to 100% of their completion. For these dataset, the recognition rate does not significantly increase after 40% of the action progress. This is mainly because most of the discriminant information is contained in the beginning of the action for these dataset. Besides predicting global activity classes, our model can also make local predictions. That means the model can predict the most probable next actionlet given observed actionlet sequence as context.

5. Conclusions and future work

In this paper, we have developed a novel approach for the segmentation, classification and prediction of ongoing human actions that takes 3D skeletal joint locations as input inferred from depth maps. The major contributions include spatio-temporal characteristic of action generated by HSOM that are connected via associative links trained by Hebbian learning; and temporal characteristic of action generated by PST for representing various order Markov dependencies between actionlets. Acquiring these characteristics relies on a good spatio-temporal decomposition of action. We have empirically shown that incorporating spatial and temporal is particularly beneficial for predicting activities. The next step is to understand and predict human activities and object affordances combining more contextual information, and more importantly, of human interactions with the objects in the form of associated affordances.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61571345, the National Natural Science Foundation of China under Grant No. 9153801, and the National Natural Science Foundation of China under Grant No. 61550110247.

References

[1] R. Poppe, A survey on vision-based human action recognition, Image Vision Comput. 28 (6) (2010) 976–990.
[2] D. Weinland, R. Ronfard, E. Boyer, A survey of vision-based methods for action representation, segmentation and recognition, Comput. Vis. Image Underst. 115 (2) (2011) 224–241.
[3] M. Ryoo, Human activity prediction: early recognition of ongoing activities from streaming videos, in: 2011 IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 1036–1043.
[4] C.S. Soon, M. Brass, H.-J. Heinze, J.-D. Haynes, Unconscious determinants of free decisions in the human brain, Nat. Neurosci. 11 (5) (2008) 543–545.
[5] J. Wang, Z. Liu, Y. Wu, Learning Actionlet ensemble for 3D human action recognition, in: Human Action Recognition with Depth Cameras, Springer, 2014, pp. 11–40.
[6] K. Li, Y. Fu, Prediction of human activity by discovering temporal sequence patterns, IEEE Trans. Pattern Anal. Mach. Intell. 36 (8) (2014) 1644–1657.
[7] K. Friston, Learning and inference in the brain, Neural Netw. 16 (9) (2003) 1325–1352.
[8] T. Kohonen, The self-organizing map, Proc. IEEE 78 (9) (1990) 1464–1480.
[9] W. Ding, K. Liu, F. Cheng, J. Zhang, STFC: spatio-temporal feature chain for skeleton-based human action recognition, J. Vis. Commun. Image Represent. 26 (2015) 329–337.
[10] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Trans. Pattern Anal. Mach. Intell. (2) (1979) 224–227.
[11] D.O. Hebb, The Organization of Behavior: A Neuropsychological Theory, Psychology Press, 2002.
[12] R. Begleiter, R. El-Yaniv, G. Yona, On prediction using variable order Markov models, J. Artif. Intell. Res. (JAIR) 22 (2004) 385–421.
[13] S. Wang, Z. Ma, Y. Yang, X. Li, C. Pang, A.G. Hauptmann, Semi-supervised multiple feature analysis for action recognition, IEEE Trans. Multimedia 16 (2) (2014) 289–298.
[14] S. Sadanand, J.J. Corso, Action bank: a high-level representation of activity in video, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, pp. 1234–1241.
[15] M. Merler, B. Huang, L. Xie, G. Hua, A. Natsev, Semantic model vectors for complex video event recognition, IEEE Trans. Multimedia 14 (1) (2012) 88–101.
[16] J. Han, L. Shao, D. Xu, J. Shotton, Enhanced computer vision with microsoft kinect sensor: a review, IEEE Trans. Cybern. 43 (5) (2013) 1318–1334.
[17] C. Wang, Z. Liu, S.-C. Chan, Superpixel-based hand gesture recognition with kinect depth camera, IEEE Trans. Multimedia 17 (1) (2015) 29.
[18] 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), IEEE, 2010, pp. 9–14.
[19] L. Xia, C.-C. Chen, J. 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), IEEE, 2012, pp. 20–27.
[20] Y. Cao, D. Barrett, A. Barbu, S. Narayanaswamy, H. Yu, A. Michaux, Y. Lin, S. Dickinson, J.M. Siskind, S. Wang, Recognize human activities from partially observed videos, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2013, pp. 2658–2665.
[21] M. Hoai, F. De la Torre, Max-margin early event detectors, Int. J. Comput. Vision 107 (2) (2014) 191–202.
[22] M. Raptis, L. Sigal, Poselet key-framing: a model for human activity recognition, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2013, pp. 2650–2657.
[23] K.M. Kitani, B.D. Ziebart, J.A. Bagnell, M. Hebert, Activity forecasting, in: Computer Vision–ECCV 2012, Springer, 2012, pp. 201–214.
[24] T. Lan, T.-C. Chen, S. Savarese, A hierarchical representation for future action prediction, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 689–704.
[25] F. Martinez-Contreras, C. Orrite-Urunuela, E. Herrero-Jaraba, H. Ragheb, S.A. Velastin, Recognizing human actions using silhouette-based HMM, in: Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, 2009, AVSS'09, IEEE, 2009, pp. 43–48.
[26] N. Sumpter, A. Bulpitt, Learning spatio-temporal patterns for predicting object behaviour, Image Vis. Comput. 18 (9) (2000) 697–704.
[27] W. Hu, D. Xie, T. Tan, S. Maybank, Learning activity patterns using fuzzy self-organizing neural network, IEEE Trans. Part B: Cybern. Syst. Man Cybern. 34 (3) (2004) 1618–1626.
[28] Q. Sun, H. Liu, Inferring ongoing human activities based on recurrent self-organizing map trajectory, in: British Machine Vision Conference, Citeseer, 2013.
[29] G. Bejerano, G. Yona, Variations on probabilistic suffix trees: statistical modeling and prediction of protein families, Bioinformatics 17 (1) (2001) 23–43.
[30] Z.L. Gang Yu, J. Yuan, Discriminative Orderlet mining for real-time recognition of human-object interaction, in: Computer Vision–ACCV 2014, Springer, 2014, pp. 201–214.
[31] C. Ellis, S.Z. Masood, M.F. Tappen, J.J. Laviola Jr, R. Sukthankar, Exploring the trade-off between accuracy and observational latency in action recognition, Int. J. Comput. Vision 101 (3) (2013) 420–436.
[32] J. Vesanto, J. Himberg, E. Alhoniemi, J. Parhankangas, Self-organizing map in Matlab: the SOM toolbox, in: Proceedings of the Matlab DSP Conference, vol. 99, 1999, pp. 16–17.
[33] M. Müller, T. Röder, Motion templates for automatic classification and retrieval of motion capture data, in: Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer animation, Eurographics Association, 2006, pp. 137–146.
[34] F. Lv, R. Nevatia, Recognition and segmentation of 3-d human action using HMM and multi-class AdaBoost, in: Computer Vision–ECCV 2006, Springer, 2006, pp. 359–372.
[35] X. Yang, Y. Tian, Effective 3d action recognition using eigenjoints, J. Vis. Commun. Image Represent. 25 (1) (2014) 2–11.
[36] M. Zanfir, M. Leordeanu, C. Sminchisescu, The moving pose: an efficient 3D kinematics descriptor for low-latency action recognition and detection, in: 2013 IEEE International Conference on Computer Vision (ICCV), IEEE, 2013, pp. 2752–2759.