Modeling 4D Human-Object Interactions for Joint Event Segmentation, Recognition, and Object Localization

Ping Wei, Yibiao Zhao, Nanning Zheng, and Song-Chun Zhu

Abstract—In this paper, we present a 4D human-object interaction (4DHOI) model for solving three vision tasks jointly: i) event segmentation from a video sequence, ii) event recognition and parsing, and iii) contextual object localization. The 4DHOI model represents the geometric, temporal, and semantic relations in daily events involving human-object interactions. In 3D space, the interactions of human poses and contextual objects are modeled by semantic co-occurrence and geometric compatibility. On the time axis, the interactions are represented as a sequence of atomic event transitions with coherent objects. The 4DHOI model is a hierarchical spatial-temporal graph representation which can be used for inferring scene functionality and object affordance. The graph structures and parameters are learned using an ordered expectation maximization algorithm which mines the spatial-temporal structures of events from RGB-D video samples. Given an input RGB-D video, the inference is performed by a dynamic programming beam search algorithm which simultaneously carries out event segmentation, recognition, and object localization. We collected and released a large multiview RGB-D event dataset which contains 3,815 video sequences and 383,036 RGB-D frames captured by three RGB-D cameras. The experimental results on three challenging datasets demonstrate the strength of the proposed method.

Index Terms—human-object interaction, object affordance, event recognition, sequence segmentation, object localization.

1 Introduction

In this paper, we present a 4D human-object interaction (4DHOI) model for solving three vision tasks jointly: i) event segmentation from a video sequence, ii) event recognition and parsing, and iii) contextual object localization in scenes. Learned from RGB-D videos, our 4DHOI model is a hierarchical spatial-temporal graph. It represents the geometric, temporal, and semantic relations in daily events involving human-object interactions. In comparison with the existing methods which often study these tasks separately [1], [2], [3], [4], [5], [6], our method for joint modeling and inference is motivated by the following observations.

Firstly, many objects in daily scenes, such as the hand-held objects in Fig. 1, can hardly be detected or recognized due to heavy occlusion and appearance variation. They need high level contextual information of human-object interactions. An extreme example is the capability of human vision in understanding a 'pantomime' where the actors pretend to use objects which do not appear and are imagined by the observers. Recognizing other functional objects, such as tables, chairs, etc., also relies on the context of human actions.

Secondly, daily events can often be hierarchically divided into sequences of atomic events defined by human poses and contextual objects. The objects and the relative duration of the atomic events also contribute to the recognition of events. For example, the events *drink with mug* and *call with*

Fig. 1: Examples of objects in video frames. These objects can hardly be recognized by appearance features inside the bounding boxes due to heavy occlusion and large appearance variation, but they can be recognized in the context of actions.

cellphone can hardly be distinguished by poses and motion features, because they are both performed by the upper body parts and have similar motion patterns. They can be told apart by the subtle difference between the appearance of *mug* and *cellphone*, and by the duration of the actions, i.e. a phone call often takes longer than a sip.

Thirdly, there is an increasing interest in inferring object functionality by affordance [7] in recent literature [8], [9], [10]. Our 4DHOI model attempts to describe affordance by a hierarchical spatial-temporal graph involving 3D human poses and contextual objects in events.

In this paper, we will demonstrate that the 4DHOI model significantly improves the performance of each individual task through joint inference. As Fig. 2 shows, the input is an RGB-D video with 3D human skeletons from a Kinect camera [11]. The output includes three parts: i) hierarchical graphs segmenting the video into events and atomic events, ii) event category labels, and iii) the contextual objects localized in the RGB-D frames.

We adopt a dynamic programming beam search algorithm for the joint inference. Based on the human pose, the objects, the relations between the pose and the objects,

- *Ping Wei is with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Shaanxi, 710049 China, and University of California, Los Angeles, CA, 90095 USA. E-mail: pingwei.pw@gmail.com.*
- *Nanning Zheng is with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Shaanxi, 710049 China. E-mail: nnzheng@mail.xjtu.edu.cn.*
- *Yibiao Zhao and Song-Chun Zhu are with the Department of Statistics, University of California, Los Angeles, CA, 90095 USA. E-mail: {yibiao.zhao,sczhu}@stat.ucla.edu.*

Manuscript received XX, 2015; revised XX, 2015.

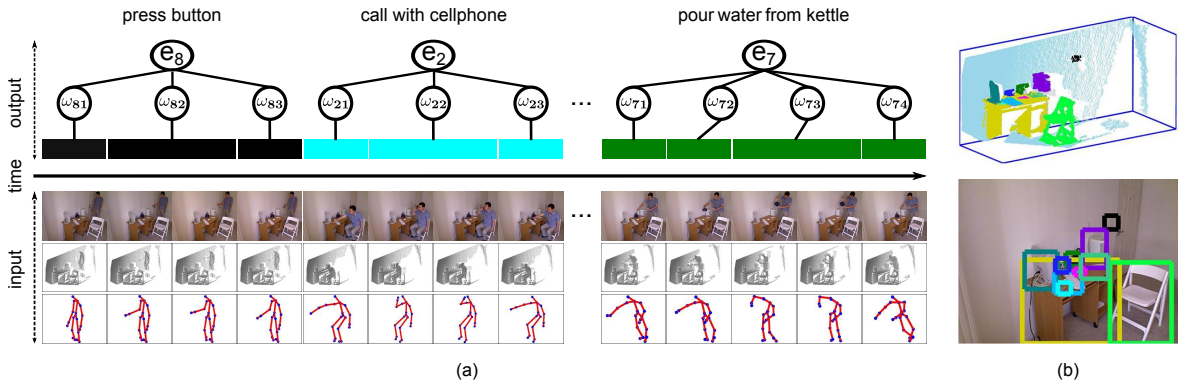

Fig. 2: The 4DHOI model. (a) The framework of the model. The input is an RGB-D video with human skeletons. The outputs are the hierarchical interpretations of the video sequence, including event recognition, segmentation, and object localization. (b) Object recognition and localization in the 3D point cloud (upper) and the RGB image (lower) through the 4DHOI model after analyzing the video events.

and the interpretations of the past frames, the algorithm proposes all possible interpretations of the current frame. Interpretations with low probability are pruned. This process iterates forward frame by frame until the video ends.

To learn the spatial-temporal graph structures and the model parameters, we propose an ordered expectation maximization algorithm (OEM). Different from conventional EM [12], OEM incorporates the temporal order of video frames and the temporal alignment of atomic events into clustering. It therefore produces temporally continuous clusters.

We collected a large-scale multiview RGB-D event dataset and have released it publicly[1]. It was captured by three stationary Kinect cameras from different viewpoints simultaneously. It includes 8 event categories, 11 object classes, 3,815 event videos, and 383,036 RGB-D frames. We tested our method on this dataset and two other challenging datasets by event recognition, segmentation, and object localization. The experimental results prove the strength of our method.

This paper is an extension of a previous conference paper [13]. The extension includes two main aspects: i) in methodology, we rewrote some variable representations in Section 3, introduced the learning algorithm in Section 5, added implementation details in Section 4.2, and discussed new strategies in Section 6; ii) in experiments, we re-trained and tested the model on more data, added more comparisons, and obtained improved results with new figures.

2 Related Work and Our Contributions

In this section, we briefly review related work and discuss our contributions compared with existing work.

2.1 Action Modeling in RGB-D Data

In recent years, RGB-D cameras, such as Kinect with 3D pose estimation [11], have motivated a new wave of studying 3D human poses, actions, and affordance from RGB-D videos [14], [15], [16], [17], and utterly changed the landscape of action modeling and recognition. Action recognition has usually been posed as a classification problem [14], [1], [18], where a feature vector is extracted from a video clip and classified into an action category. Wang *et al.* [14] designed features to represent 3D pose sequences and mined the actionlet ensemble to classify actions. Such methods need the event clips to be given or pre-segmented, which is ineffective in real applications such as video surveillance. Moreover, these methods do not interpret the objects involved in actions. Our model represents skeleton features in each frame and overcomes noise and ambiguity by incorporating object interactions and temporal relations. In addition to recognizing actions, our method segments the video sequence and recognizes the objects in each frame.

To understand long unsegmented video sequences, some existing methods combine action recognition and segmentation or detection [2], [19], [20], [21], [22], [23]. Lv and Nevatia [2] used hidden Markov model (HMM) to describe each action class and a dynamic programming method to segment and recognize actions. Shi *et al.* [22] described temporal boundaries and addressed action segmentation and recognition in a discriminative way. These methods model actions of simple temporal structures, such as *walk* and *run*. They do not model the interactions between humans and objects.

Explicitly modeling the inner structures of actions contributes to action recognition [24], [25], [2], [26], [27], [28], [29]. HMM [24] is usually used to describe the state transitions [2], [26] between frames or action snippets. Tang *et al.* [25] introduced duration to HMM. Pei *et al.* [27] represented an action with atomic actions and employed a temporal filter embedded in an And-Or graph for video parsing. Sung *et al.* [29] decomposed a human activity into sub-activities and solved the model under the dynamic programming framework. Inspired by these methods, our model integrates human actions, objects, and their interaction relations into a unified framework.

2.2 Object Modeling and Localization

Many existing methods for object detection represent objects with appearance [30], [5], [6], [31], such as HOG [5] features. Lai *et al.* [6] extended the HOG features to RGB-D images. Such features are often weakened by low resolution, heavy occlusion, and large appearance variation. To solve such problems, contextual information was introduced into object

1. http://vcla.stat.ucla.edu/download.html

modeling. The methods in [32] and [33] incorporated relations with other objects to improve detection or recognition. Zhao and Zhu [9] defined objects by integrating function, geometry, and appearance information. Gupta *et al.* [10] extended object localization and recognition to human-centric scene understanding by inferring human and 3D scene interactions. These methods aim at object detection or recognition in still images.

Some studies aim at recognizing and localizing objects in videos [34], [15]. Gupta *et al.* [34] labeled objects according to human actions in videos. The method in [15] tracked objects in each video frame. In comparison, our method localizes objects in both 3D point clouds and 2D images, and does not need accurate initialization of object locations.

2.3 Human-object Interaction and Affordance

The concept of affordance was originally studied by Gibson [7] and further developed by many studies to describe the relations between organisms (humans) and environments (objects) [35], [36], [37], [38], [39]. Many researchers have recently applied human-object relations to event, object, and scene modeling [10], [34], [40], [15], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51]. Gupta *et al.* [34] combined spatial and functional constraints between humans and objects to recognize actions and objects. Prest *et al.* [44] inferred spatial information of objects by modeling 2D geometric relations between human bodies and objects. Yao and Fei-Fei [46] detected objects by modeling relations between actions, objects, and poses in still images. These methods define the human-object interactions in 2D images. Such contextual cues are often weakened by viewpoint change and occlusion. Koppula *et al.* [15] modeled relations between human activities and object affordance, and their changes over time. This method requires videos to be pre-segmented, and the object detection is independent of human actions. Our model incorporates event recognition, segmentation, and object localization into a unified framework, under which these tasks mutually facilitate each other.

Human-object interactions are also used in robotics [52], [53], [54]. Aksoy *et al.* [52] recognized manipulations by learning object-action semantics. Wörgötter *et al.* [53] modeled manipulation actions for robot task execution. This stream of research demonstrates the significance of human-object interactions from the perspective of robot learning and task execution.

2.4 Action Structure Learning

Many existing approaches mine action structures by explicitly modeling the latent structures [4], [2], [25], [27], [55]. HMM [2] learned the hidden states and transition probabilities with maximum likelihood estimation. Hidden conditional random field (HCRF) [55] learned the hidden structures of actions in a discriminative way. These methods define the temporal structures on video frames or fixed-size video segments, which can not effectively characterize nor utilize the duration information of hidden structures.

Yao and Fei-Fei [46] defined atomic poses in still images and learned them through clustering human poses. Zhou *et al.* [4] segmented an action sequence into motion primitives with hierarchical cluster analysis. These clustering methods are under the framework similar to the expectation-maximization (EM) clustering [12]. Conventional EM does not consider the temporal order of sequence frames, which may produce undesirable clustering results. For example, as is shown in Fig. 3, the poses of *approach the dispenser* and *leave the dispenser* are very similar. Without considering the temporal order, these two poses may be clustered into the same cluster. Though the method in [4] introduced the temporal order into clustering, it did not consider the mutual constraints among the sequences of the same category, but rather carried out the frame clustering for each independent sequence.

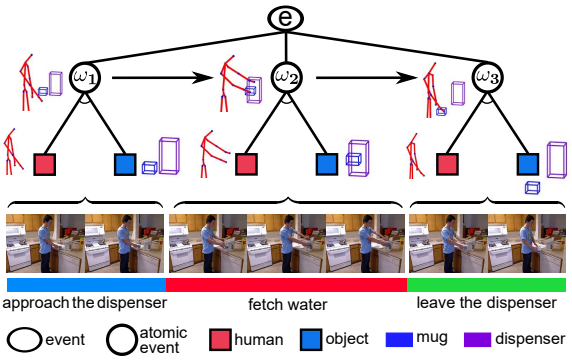

Fig. 3: A hierarchical graph of the 4D human-object interactions for an example event *fetch water from dispenser*.

2.5 Our Contributions

In comparison with the previous work, this paper makes four contributions.

1. It presents a 4D human-object interaction model as a stochastic hierarchical spatial-temporal graph, which represents the 3D human-object relations and the temporal relations between atomic events in RGB-D videos.
2. It develops a unified framework for joint inference of event recognition, sequence segmentation, and object localization.
3. It proposes an unsupervised algorithm to learn the latent temporal structures of events and the model parameters from sequence samples.
4. It tests the model on three challenging datasets, and the performance demonstrates the strength of the model.

3 4D Human-Object Interaction Model

As Fig. 3 illustrates, the 4DHOI model is a hierarchical graph for an event. On the time axis, an event is decomposed into several ordered atomic events. For example, the event *fetch water from dispenser* is decomposed into three sequential atomic events - *approach the dispenser*, *fetch water*, and *leave the dispenser*. An atomic event corresponds to a continuous segment in a video sequence, and contains similar human poses and object interactions. These atomic events are treated as hidden variables, which are learned by mining and clustering sequence samples.

In 3D space, an atomic event is decomposed into a human pose, one or multiple objects, and the relations between the pose and the objects. The relations include semantic relations and 3D geometric relations. The semantic relations

between the object classes and the atomic event are treated as hard constraints. For example, the atomic event *fetch water* consists of the pose *fetch* and the objects *dispenser, mug*, as is shown in Fig. 3.

We formulate the 4DHOI with a stochastic hierarchical graph similar to And-Or Graph [56]. Let $V = (f_1, ..., f_\tau)$ be a video sequence in the time interval $[1, \tau]$, where $f_t = (I_t, h_t)$ is the frame at time t. I_t is the RGB-D data. h_t is the human pose feature extracted from the 3D skeletons estimated by motion capture technology [11].

The sequence V is interpreted by the hierarchical graph $G =< E, L >$ as follows.

i) $E \in \Delta = \{e_i | i = 1, ..., |\Delta|\}$ is the event category such as *fetch water from dispenser*. Δ is the set of event categories.

ii) $L = (l_1, ..., l_\tau)$ is a sequence of frame labels. $l_t = (a_t, o_t)$ is the interpretation of the frame f_t. $a_t \in \Omega_E = \{\omega_i | i = 1, ..., K_E\}$ is the atomic event label such as *fetch water*.

Ω_E is the atomic event set of E. Each event category e_i has its distinct atomic event set Ω_{e_i}, i.e. the relations between an event and its atomic events are hard constraints.

$o_t = (o_t^1, ..., o_t^{n_t})$ are the objects interacting with the human at time t, where n_t is the number of objects. Each object has a class label and a 3D location.

Similar to the graphical formulation in [56], the energy that the video V is interpreted by the graph G is defined as

$$\mathrm{En}(G|V) = \sum_{t=1}^{\tau} \Phi(f_t, l_t) + \sum_{t=2}^{\tau} \Psi(l_{1:t-1}, l_t). \quad (1)$$

$\Phi(\cdot)$ is the spatial energy term of a single frame, encoding the human-object interactions in 3D space.

$\Psi(\cdot)$ is the temporal energy term encoding the temporal relations between the current frame l_t and all previous frames $l_{1:t-1}$. This is different from conventional HMM [24]. The variable E is omitted in the right side of Eq. (1) since each event has its own distinct atomic event set.

3.1 Human-object Interactions in 3D Space

$\Phi(f_t, l_t)$ in Eq.(1) describes the human-object interactions in 3D space. The interactions include:

1) semantic co-occurrence between a specific type of human pose and the object classes; and
2) geometric compatibility describing the 3D spatial constraints between the human pose and the objects.

Thus, $\Phi(f_t, l_t)$ is further decomposed into three terms which will be defined in the remainder of the subsection,

$$\Phi(f_t, l_t) = \phi_1(a_t, h_t) + \phi_2(a_t, o_t, I_t) + \phi_3(a_t, h_t, o_t). \quad (2)$$

3.1.1 Pose Model

$\phi_1(a_t, h_t)$ is the human pose model. A human skeleton consists of multiple 3D joints estimated by motion capture technology, like the Kinect camera [11]. To normalize the data, we align all the skeletons to a reference skeleton so that the torsos and shoulders of all the skeletons have the same location, scale, and direction.

The feature of each joint is defined as the 3D coordinate concatenating the motion vector which is the difference of joint coordinates in two successive frames. A feature vector containing the features of joints on the human body is extracted and processed with PCA to reduce the correlation and noise. h_t is the PC parameter vector. It is assumed to follow a Gaussian distribution for each atomic event. The model is

$$\phi_1(a_t, h_t) = -\ln N(h_t; \mu_{a_t}, \Sigma_{a_t}), \quad (3)$$

where μ_{a_t} and Σ_{a_t} are respectively the mean and the covariance in the atomic event a_t.

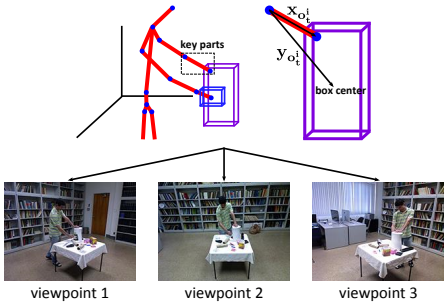

Fig. 4: Human-object geometric relations in 3D space.

3.1.2 Contextual Object Model

$\phi_2(a_t, o_t, I_t)$ is the term for fitting the contextual objects o_t to the RGB-D data I_t. Each object o_t^i includes a class label, e.g. *mug*, and a 3D bounding box located at z_t^i in 3D space. The 3D box is projected onto the RGB and depth images to form 2D bounding boxes, from which the RGB-D HOG features [5], [6] are extracted. Let $s(z_t^i)$ be the score of the linear SVM object detector using the RGB-D HOG features at location z_t^i. We convert this score into a probability using Platt scaling [6], [57], i.e. $p(z_t^i|I_t) = 1/\{1 + \exp\{\nu_1 s(z_t^i) + \nu_2\}\}$, where ν_1, ν_2 are parameters. The object model is

$$\phi_2(a_t, o_t, I_t) = -\frac{1}{n_t} \sum_{i=1}^{n_t} \ln p(z_t^i|I_t), \quad (4)$$

where n_t is the number of objects. $1/n_t$ offsets the influence of different object numbers. Because the relation between a_t and the object label is a hard constraint, we omit a_t in the right side of Eq. (4) for clarity.

Our model defines the object location in 3D space, and the object appearance in 2D images, which is more robust to viewpoint and scale changes. The 3D boxes also provide a natural way for defining the 3D geometric relations between objects and human poses.

3.1.3 3D Geometric Compatibility and Object Prediction

The third energy term $\phi_3(a_t, h_t, o_t)$ measures the geometric relations between a human pose and objects. Geometric relations are mostly defined in 2D images [44], [46], and are not suitable for different viewpoints, as Fig. 4 illustrates. We model the geometric relations in 3D space.

Geometric relations between a human pose and objects are time-varying in different atomic events. For example, in Fig. 3, the human hand is far from the dispenser in the atomic event *approach the dispenser* while touches it in *fetch water*. Each atomic event has different geometric relations.

In an atomic event, an object interacts with some body parts, which we call the key parts. For example, in Fig. 4, the left arm interacts with the dispenser and the right arm interacts with the mug. The locations of objects in 3D space

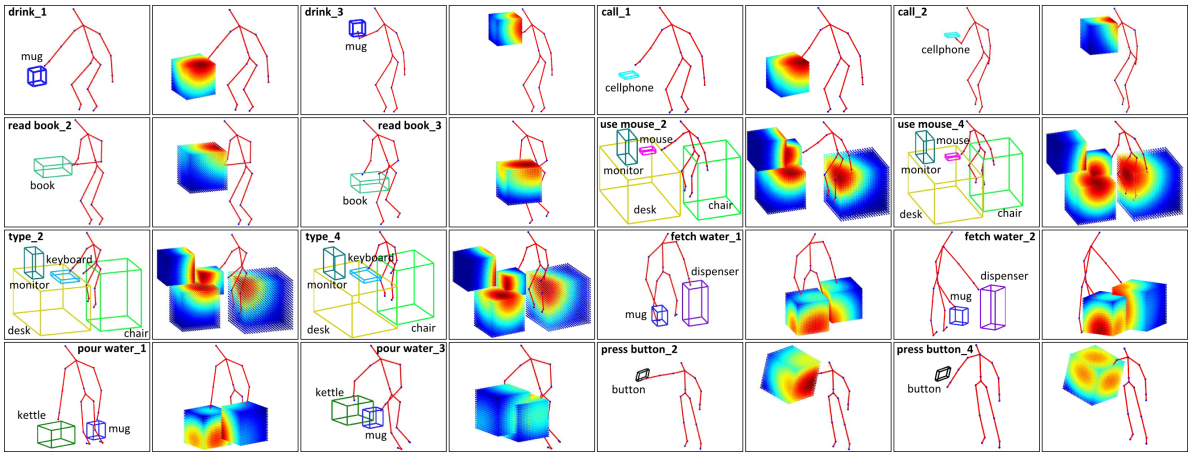

Fig. 5: Examples of the learned geometric relations in atomic events. The odd-number columns are about the instances of the learned atomic events. The indices denote the atomic event number in each event. The even-number columns are about the probability maps of object prediction, where warmer colors indicate higher probabilities of locations where objects appear.

are closely related to and largely revealed by the locations and orientations of the key parts.

As is shown in Fig. 4, let $y_{o_t^i}$ be the difference vector from one joint of the key parts to the object bounding box center. $x_{o_t^i}$ is the difference vector between the end points of the key parts. $y_{o_t^i}$ is closely related to $x_{o_t^i}$. We define $\eta_{o_t^i} = y_{o_t^i} - W_{o_t^i}^{a_t} x_{o_t^i}$, where $W_{o_t^i}^{a_t}$ is a similarity transformation matrix. $\eta_{o_t^i}$ describes the location of the object relative to the key parts. We assume $\eta_{o_t^i}$ follows a Gaussian distribution. This formulation is motivated by the observation that, for an atomic event, the instances of an object category have similar locations relative to the key parts. The 3D geometric relation is modeled as

$$\phi_3(a_t, h_t, o_t) = -\frac{1}{n_t} \sum_{i=1}^{n_t} \ln N(\eta_{o_t^i}; \mu_{o_t^i, a_t}^R, \Sigma_{o_t^i, a_t}^R), \quad (5)$$

where $\mu_{o_t^i, a_t}^R$ is the mean and $\Sigma_{o_t^i, a_t}^R$ is the covariance. The sign R is used to differentiate the 3D relation parameters from others. The subscript (o_t^i, a_t) indicates that the geometric relation varies for different atomic events and objects.

The vector $x_{o_t^i}$ is like a local reference, by which we can estimate $y_{o_t^i}$ and therefore predict the locations of related objects. Fig. 5 illustrates some examples of atomic events and the probability maps of the learned geometric relations. As Fig. 5 shows, according to the key parts, the probability that an object appears at a 3D location can be evaluated.

3.2 Temporal Relation

The temporal relation $\Psi(l_{1:t-1}, l_t)$ is decomposed as

$$\Psi(l_{1:t-1}, l_t) = \psi_1(a_{1:t-1}, a_t) + \psi_2(o_{t-1}, o_t), \quad (6)$$

where $a_{1:t-1}$ are the atomic event labels of the frames from time 1 to $t-1$. The first term encodes the atomic event transition, and the second term encodes the temporal coherence of objects.

3.2.1 Atomic Event Transition

In an event, the transition probability from the current atomic event ω_{k-1} to the next atomic event ω_k depends on the duration of the current atomic event, denoted by d_{k-1}. We model the time-varying transition probability with a logistic sigmoid function.

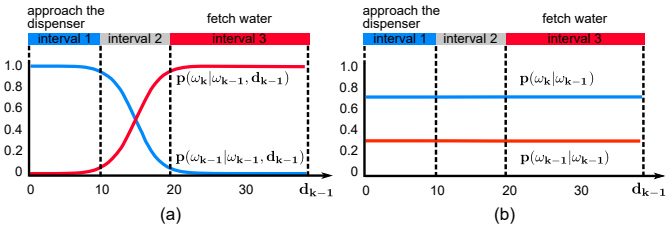

Fig. 6: The atomic event transition probability. (a) Duration-dependent transition. (b) Duration-independent transition.

Fig. 6 compares two kinds of transition probabilities. Fig. 6 (a) is a duration-dependent transition. In interval 1 when the hand is still far from the dispenser, the probability of staying at *approach the dispenser* is much larger than the possibility of changing to the next atomic event *fetch water*. As the duration of *approach the dispenser* becomes long, in interval 3, the probability of staying at *approach the dispenser* will be close to zero, and the probability of transitioning to *fetch water* is almost 1. During interval 2, the transition will most likely happen. In contrast, if we use a duration-independent transition probability, they will be constant, regardless of the duration, as Fig. 6 (b) shows.

Let ω_{k-1} and ω_k be two consecutive atomic events of an event E. d_{k-1} is the duration of ω_{k-1} up to time $t-1$. We define the time-varying transition probability as

$$p(a_t = \omega_k | a_{t-1} = \omega_{k-1}, d_{k-1}) = \sigma(\beta d_{k-1} + \gamma), \quad (7)$$

where $\sigma(v) = 1/(1+e^{-v})$ is a logistic sigmoid function. We simplify $p(a_t = \omega_k | a_{t-1} = \omega_{k-1}, d_{k-1})$ as $p(\omega_k | \omega_{k-1}, d_{k-1})$. The transition probability to ω_{k-1} is $p(\omega_{k-1} | \omega_{k-1}, d_{k-1}) = 1 - p(\omega_k | \omega_{k-1}, d_{k-1})$. Then the energy $\psi_1(a_{1:t-1}, a_t)$ is $-\ln p(\omega_k | \omega_{k-1}, d_{k-1})$ or $-\ln p(\omega_{k-1} | \omega_{k-1}, d_{k-1})$, up to the value of a_t.

3.2.2 Temporal Coherence of Objects

$\psi_2(o_{t-1}, o_t)$ describes the temporal coherence of objects. In an event, the locations of some objects, such as dispensers, are almost static, while other objects, such as mugs, move

with hands. For *moveable* objects, we assume the location follows a Gaussian distribution $p(z_t^i|z_{t-1}^i) = N(z_t^i - z_{t-1}^i; \mu^Z_{o_t^i,a_t}, \Sigma^Z_{o_t^i,a_t})$. For *static* objects, we set a threshold. If the difference of the proposed location z_t^i in the current frame and the location at the last frame z_{t-1}^i is smaller than the threshold, $p(z_t^i|z_{t-1}^i)$ is 1, otherwise 0. The energy is

$$\psi_2(o_{t-1}, o_t) = -\frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} \ln p(z_t^i|z_{t-1}^i). \quad (8)$$

4 Inference

Given a video $\mathbf{V}$ in the time interval $\wedge = [1, T]$ which contains Q ($Q \geq 1$) events, the goal of inference is to interpret $\mathbf{V}$ with a graph list $\mathbf{G} = (G_1, G_2, ..., G_Q)$. The graph G_q is the interpretation of the video clip V_q in the interval $\wedge_q$, where $\bigcup_{q=1}^{Q} \wedge_q = \wedge$ and $\bigcap_{q=1}^{Q} \wedge_q = \varnothing$. We segment $\mathbf{V}$ into Q disjoint segments $(V_1, V_2, ..., V_Q)$ and interpret V_q with G_q by optimizing a posterior probability

$$p(\mathbf{G}|\mathbf{V}) = \prod\nolimits_{q=1}^{Q} p(G_q|V_q),$$

or equivalently, minimizing the total energy

$$\mathcal{E}(\mathbf{G}|\mathbf{V}) = \sum\nolimits_{q=1}^{Q} \mathrm{En}(G_q|V_q), \quad (9)$$

where $\mathrm{En}(G_q|V_q)$ is the energy of each video clip defined in Eq. (1). The most likely interpretation of $\mathbf{V}$ is computed as

$$\mathbf{G}^* = \arg\min \mathcal{E}(\mathbf{G}|\mathbf{V}). \quad (10)$$

4.1 Dynamic Programming Beam Search

We use a dynamic programming beam search algorithm (DPBS) to solve Eq. (10). The DPBS was previously used in machine language translation [58]. We improve and extend it for the hierarchical interpretation of videos. The general framework of our DPBS includes four processes:

1) searching for objects and producing multiple hypothesized objects in the current frame;
2) proposing the possible interpretations of the current frame according to the pose feature, the object detection, and the 3D relations between them;
3) computing multiple graph lists and their energies of the current video with the interpretations of the past video and the current frame;
4) keeping those graph lists with higher probability and continuing to interpret the next frame.

The above processes iterate forward frame by frame until the video ends. The graph list with the highest probability will be the final interpretation of the video. Fig. 7 illustrates the DPBS. In the following, we elaborate on how to compute the graph lists and their energies.

Let $\mathbf{G}_{t-1}^1, ..., \mathbf{G}_{t-1}^J$ be J interpretive graph lists for the video in $[1, t-1]$, with the energies $\mathcal{E}_{t-1}^1, ..., \mathcal{E}_{t-1}^J$ respectively. They are shown as the paths from time 1 to $t-1$ in Fig. 7(b). At time t, we need to compute an interpretation of the current frame f_t, based on each of the J paths, for example, the jth path (the green path) in Fig. 7(b). Let a_{t-1} and a_t be the atomic event labels of the frames f_{t-1} and f_t, respectively. Given the jth path $\mathbf{G}_{t-1}^j$, there are three types of interpretations of the frame f_t (shown in Fig. 7(d)):
1) a_t repeats the same atomic event with a_{t-1};
2) a_t transitions to the next atomic event in the same event;
3) a_t is the atomic event of a new event.

All the possible values of a_t are appended to $\mathbf{G}_{t-1}^j$ according to the three types of interpretations, which generates m_j new graph lists $\mathbf{G}_t^1(\mathbf{G}_{t-1}^j), ..., \mathbf{G}_t^{m_j}(\mathbf{G}_{t-1}^j)$ with the energy $\mathcal{E}_{t-1}^j + \Phi(f_t, l_t) + \Psi(l_{1:t-1}, l_t)$. With all $\mathbf{G}_{t-1}^1, ..., \mathbf{G}_{t-1}^J$, we obtain $m_1 + ... + m_J$ possible solutions. We keep the J solutions $\mathbf{G}_t^1, ..., \mathbf{G}_t^J$ with the lowest energies $\mathcal{E}_t^1, ..., \mathcal{E}_t^J$ as the possible interpretations of the video in the interval $[1, t]$.

We use a simplified example shown in Fig. 7 to illustrate the algorithm. $\mathbf{G}_{t-1}^j$ is the jth interpretation of the video in the interval $[1, t-1]$. Based on $\mathbf{G}_{t-1}^j$, there exist three interpretations of the current frame f_t. Appending them to $\mathbf{G}_{t-1}^j$ produces three possible interpretations of the video in $[1, t]$, i.e. $\mathbf{G}_t^1(\mathbf{G}_{t-1}^j), \mathbf{G}_t^2(\mathbf{G}_{t-1}^j), \mathbf{G}_t^3(\mathbf{G}_{t-1}^j)$, and $m_j = 3$. With all $\mathbf{G}_{t-1}^1, ..., \mathbf{G}_{t-1}^J$, there exist a total of $m_1 + ... + m_J$ possible interpretations of the sequence in $[1, t]$.

4.2 Implementation Details

Let N_a be the number of all atomic events. Without any constraints, there are a total of ${N_a}^T$ interpretations of a sequence of length T. It is incalculable because T is often larger than 100. With the beam search number J, the complexity of our algorithm is $O(JTN_a)$, which can be computed efficiently.

Event recognition is to predict an event category for a video. By setting $Q = 1$, DPBS computes an interpretive graph for the video. The graph root is the event label. Sequence segmentation is to cut a long video sequence into coherent segments that each segment corresponds to an event. DPBS can interpret a frame as a *new event*, at which the sequence is cut into segments of different events.

Object localization is to determine the object location in both the 3D point cloud and the 2D image of each video frame. SVM-trained detectors (as defined in Section 3.1.2) scan each frame to generate multiple hypothesized objects. Then the object detection score, the human pose context, and the temporal coherence are incorporated to maintain the locations with high probability.

5 Learning

In this section, we present how to learn the hierarchical structures of events, the atomic events, the temporal relations, and the model parameters.

5.1 Ordered Expectation-Maximization Learning

To learn the hierarchical structures of an event category, each sample video of the category should be cut into disjoint segments so that each segment corresponds to an atomic event, as is shown in Fig. 8(a). However, event structures are hidden. Though EM is widely used for unsupervised clustering [12], it clusters frames without considering the temporal order, and thus can not produce continuous segments, as is shown in Fig. 8(b). We propose an ordered expectation maximization algorithm (OEM) to learn the event structures and the model parameters, as is shown in Fig. 8(a).

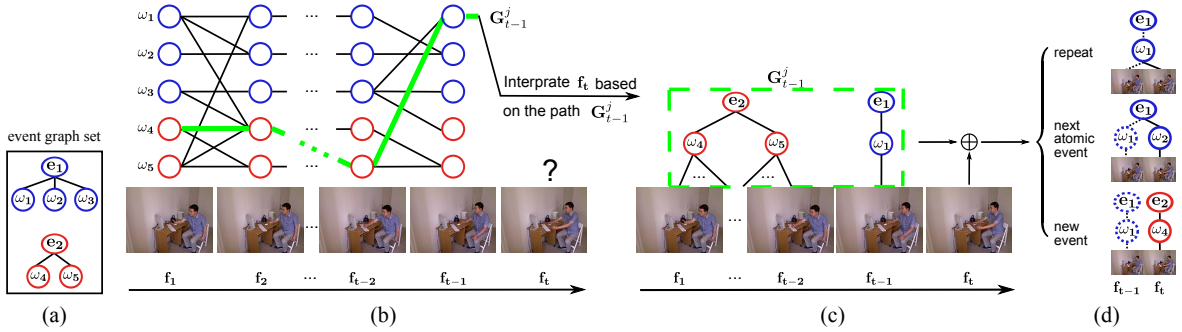

Fig. 7: The dynamic programming beam search inference algorithm. (a) Toy examples of the given graph set. The goal is to interpret the input video sequence with the graphs in this set. (b) Dynamic programming process to interpret each frame. Each path denotes one possible interpretation of the video. (c) Interpreting the frame f_t based on the jth interpretation $\mathbf{G}^j_{t-1}$ of the video in the interval $[1, t-1]$. (d) Three types of interpretations of the frame f_t based on $\mathbf{G}^j_{t-1}$.

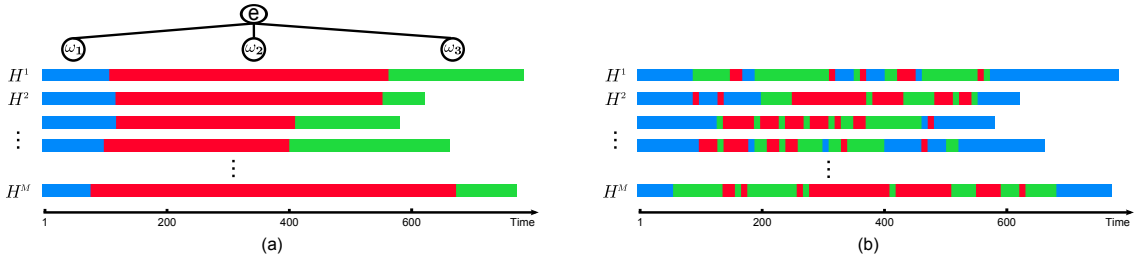

Fig. 8: Comparison between OEM and EM. (a) Our OEM. Each color denotes an atomic event. (b) Conventional EM.

As Section 3.1.1 states, the human pose feature of each frame follows a Gaussian distribution. From another perspective, we can assume each frame pose feature in a video follows a Gaussian mixture and each component of the mixture corresponds to an atomic event. This assumption is based on, and characterizes the fact that, the human poses of an atomic event in a video and different videos are similar.

Let $\{H^m|m = 1, 2, ..., M\}$ be M video samples of an event category, as the color bars in Fig. 8. Each sequence is composed of T_m frames $H^m = (h^m_1, h^m_2, ..., h^m_{T_m})$, where h^m_t is the frame pose feature at time t in the mth sequence. With these samples, the goal is to cut each sequence H^m into K disjoint segments so that the frames in the kth segment belong to the kth atomic event, as shown in Fig. 8(a). The number K is decided empirically, and an event typically has $K = 3, 4$ atomic events in our experiments.

To achieve this goal, we introduce latent variables $\mathbf{s}^m = (s^m_2, ..., s^m_K)$, and two constants, $s^m_1 = 1, s^m_{K+1} = T_m + 1$, for each sequence H^m, where $s^m_k \in \{1, ..., T_m, T_m + 1\}$ is the time boundary of the segment, and $s^m_k < s^m_{k+1}$. These latent variables cut the sequence H^m into K disjoint segments. The kth segment starts at time s^m_k and ends at time $s^m_{k+1} - 1$. The frame pose feature in the kth segment follows the kth component distribution $\mathcal{N}(h^m_t|\mu_k, \Sigma_k)$ of the Gaussian mixture. For all the sequence samples $\{H^m|m = 1, 2, ..., M\}$, we define

$$L(\boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{S}) = \prod_{m=1}^{M} \prod_{k=1}^{K} \prod_{t=s^m_k}^{s^m_{k+1}-1} \theta_k \mathcal{N}(h^m_t|\mu_k, \Sigma_k), \quad (11)$$

where $\boldsymbol{\theta} = (\theta_1, ..., \theta_K)$, $\boldsymbol{\mu} = (\mu_1, ..., \mu_K)$, $\boldsymbol{\Sigma} = (\Sigma_1, ..., \Sigma_K)$. $\mathbf{S} = (\mathbf{s}^1, ..., \mathbf{s}^M)$ is the latent variable set of all the sequences. The optimal parameters are computed as

$$(\boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{S})^* = \arg\max L(\boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{S}). \quad (12)$$

5.1.1 Optimization

It is difficult to solve Eq. (12) due to its exponential complexity. Our OEM optimizes Eq. (12) under the framework similar to EM [12]. Instead, OEM introduces the latent segment variables to determine sample assignment. These segment variables are unknown and should also be optimized. In the E step, we compute the optimal temporal segmentation for each sequence to cluster the samples, instead of computing the responsibilities as in EM. Given $\mathbf{S}$, we can optimize Eq. (12) with respect to $\boldsymbol{\theta}$, $\boldsymbol{\mu}$, and $\boldsymbol{\Sigma}$ by Lagrange multiplier method. The OEM optimization is summarized as:

1) **Initialization.** Initialize $\mathbf{S}$ by uniformly segmenting each video sequence into K segments. With $\mathbf{S}$, initialize $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{\theta}$ with Eq. (14).
2) **E step**. Compute the optimal segmentation for each sequence $H^m (m = 1, ..., M)$ with the current $\boldsymbol{\theta}$, $\boldsymbol{\mu}$, and $\boldsymbol{\Sigma}$,

$$(\mathbf{s}^m)^* = \arg\max_{\mathbf{s}^m} \prod_{k=1}^{K} \prod_{t=s^m_k}^{s^m_{k+1}-1} \theta_k N(h^m_t|\mu_k, \Sigma_k). \quad (13)$$

   In Section 5.1.2, we will detail how to optimize Eq. (13).
3) **M step**. Re-estimate the parameters with the new $\mathbf{S}$,

$$\mu_k = \frac{1}{N_k} \sum_{m=1}^{M} \sum_{t=s^m_k}^{s^m_{k+1}-1} h^m_t,$$
$$\Sigma_k = \frac{1}{N_k} \sum_{m=1}^{M} \sum_{t=s^m_k}^{s^m_{k+1}-1} (h^m_t - \mu_k)(h^m_t - \mu_k)^T, \quad (14)$$
$$\theta_k = \frac{N_k}{N},$$

   where $N_k = \sum_{m=1}^{M} (s^m_{k+1} - s^m_k)$ is the frame number of the kth cluster.

4) **Evaluate** the logarithm of Eq. (11),

$$\ln L = \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{t=s_k^m}^{s_{k+1}^m-1}\{\ln\theta_k - \frac{dim}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_k| - \frac{1}{2}(h_t^m-\mu_k)^T{\Sigma_k}^{-1}(h_t^m-\mu_k)\}, \quad (15)$$

where dim is the dimension of h_t^m. If the convergence criterion is satisfied, stop and output the optimization results; else, return to 2) E step.

5.1.2 Computing Optimal Temporal Segmentation

The optimization space size of Eq. (13) is exponentially related to the sequence length. It is difficult to get the global optimization by exhaustive search.

We propose to optimize it with an iterative approximation programming (IAP). At each iteration, each component of $\mathbf{s}^m$ is sequentially optimized conditioned on its other components. For clarity, we denote $\mathbf{s}^m = (s_2^m, ..., s_K^m)$ as $\mathbf{s} = (s_2, ..., s_K)$, and define $\mathbf{s}^{(i)} = (s_2^{(i)}, ..., s_K^{(i)})$ as the value in the iteration step i. We denote

$$\lambda(\mathbf{s}) = \prod_{k=1}^{K}\prod_{t=s_k}^{s_{k+1}-1}\theta_k N(h_t|\mu_k,\Sigma_k). \quad (16)$$

Eq.(13) is rewritten as

$$\mathbf{s}^* = \arg\max_{\mathbf{s}} \lambda(\mathbf{s}). \quad (17)$$

The IAP algorithm is summarized as:

1) **Initialize** $\mathbf{s}^{(0)} = (s_2^{(0)}, ..., s_K^{(0)})$ with the values in the last E step.
2) **Iterate** $i = i+1$, and optimize

$$\begin{aligned} s_2^{(i)} &= \arg\max_{s_2} \lambda(s_2, s_3^{(i-1)}, ..., s_K^{(i-1)}),\\ s_3^{(i)} &= \arg\max_{s_3} \lambda(s_2^{(i)}, s_3, s_4^{(i-1)} ..., s_K^{(i-1)}),\\ &\vdots\\ s_K^{(i)} &= \arg\max_{s_K} \lambda(s_2^{(i)}, ..., s_{K-1}^{(i)}, s_K). \end{aligned} \quad (18)$$

3) **Check** for convergence. If the convergence condition is satisfied, stop and output $\mathbf{s}^{(i)}$ as the optimal segmentation; else, return to step 2).

5.1.3 Estimating Parameters

The pose model of the kth atomic event is the kth component of the Gaussian mixture. The co-occurring object categories in all the frames of the kth segment are set as the interacting object categories for the kth atomic event. The 3D geometric relation parameters are computed using maximum likelihood estimation with the samples of the kth segment.

5.2 Learning Temporal Relation

While learning atomic events with OEM, each sample sequence of the event E is cut into K segments. The number of frames in each segment suggests the duration of each atomic event. We use these duration samples to learn the parameters of the atomic event transition, i.e. β and γ in Eq. (7), for two neighbouring atomic events ω_{k-1} and ω_k.

We use the logistic function learning strategy [12]. Given $a_{t-1} = \omega_{k-1}$, a_t can be ω_{k-1} or ω_k. Let d be the continuous duration of ω_{k-1} in previous frames. We introduce a $0-1$ variable b corresponding to each d. Given d, $b=1$ if the atomic event in the next frame is ω_k; $b=0$ if it is ω_{k-1}.

s_{k-1}^m, s_k^m, and s_{k+1}^m are the segment boundaries of ω_{k-1} and ω_k in the mth sequence. In the interval $[s_{k-1}^m, s_{k+1}^m]$, the duration d_{k-1} of the atomic event ω_{k-1} can be $1, 2, ..., s_k^m - s_{k-1}^m, ..., s_{k+1}^m - s_{k-1}^m$. When $d \in \{1, 2, ..., s_k^m - s_{k-1}^m - 1\}$, the atomic event in the next frame is ω_{k-1}, therefore $b=0$; when $d \in \{s_m^k - s_m^{k-1}, ..., s_m^{k+1} - s_m^{k-1}\}$, the atomic event in the next frame is ω_k, therefore $b=1$. In this way, we obtain $s_{k+1}^m - s_{k-1}^m$ pair samples from the mth sequence: $\{(d_i, b_i)|i = 1, 2, ..., s_{k+1}^m - s_{k-1}^m\}$. Suppose we obtain a total of D pair samples from all the M sequences. $\mathbf{b} = (b_1, ..., b_D)$ is the label set. The likelihood function is

$$p(\mathbf{b}|\beta,\gamma) = \prod_{n=1}^{D} c_n^{b_n}(1-c_n)^{1-b_n}, \quad (19)$$

where $c_n = p(\omega_k|\omega_{k-1}, d_{k-1})$, as defined in Eq. (7).

The parameters β and γ in Eq.(19) can be computed using maximum likelihood estimation.

6 Scene Alignment and Object Search

6.1 Scene Alignment

Due to variations of camera positions and view angles, the geometric relations between a human body and objects vary in different scenes. To learn the geometric relations, all original scenes of point clouds should be aligned. We implement the alignment in two steps:

1) Transforming the scenes from the camera coordinate system to the world coordinate system; and
2) Rotating the scenes so that all the scenes have the same direction relative to the human body.

We assume the planes of the floor and two neighbouring walls in a scene are orthogonal to each other. In the first step, the original scene is transformed so that the planes of the floor and the walls are parallel to the world coordinate planes. We adopt a Manhattan world method [59], [60]. We create a Delaunay triangulation of the scene point cloud and compute the norms of all the triangles, as shown in Fig. 9(b). These norms are clustered into several clusters with k-means in Fig. 9(c). The top three clusters with maximum samples correspond to the planes of the floor and two neighboring walls. This assumption is based on the observation that the floor and wall directions are dominant in all the plane directions in the data of an indoor scene.

The three mean directions of the three clusters are orthogonalized using the Gram-Schmidt method. The three orthogonal directions are the norms of the floor and wall planes, which are aligned to the world coordinate system to obtain the transformation matrix. With this matrix, the scene is transformed from the camera coordinate system to the world coordinate system in Fig. 9(d).

In the second step, we rotate all the scenes and the human skeletons in the scenes so that the skeletons have the same direction as the reference skeleton, as Fig. 9 (e) and (f) show.

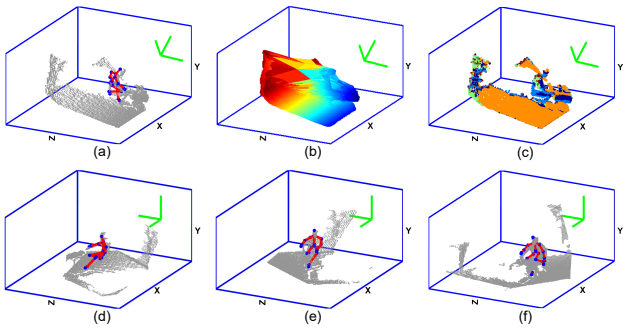

Fig. 9: Scene alignment. (a) Original scene. (b) Delaunay triangles surfaces. (c) Norm clustering. (d) Transformed scene. (e) Aligned scene. (f) Reference scene.

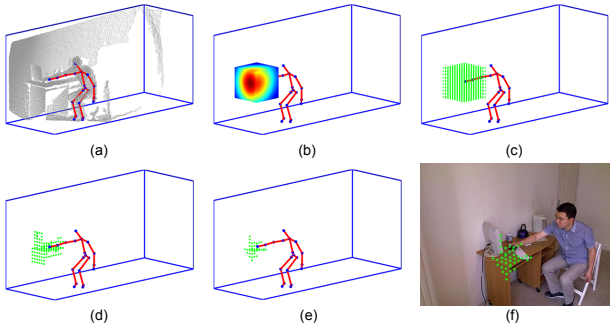

Fig. 10: Object search. (a) Point cloud. (b) Object prediction probability. (c) Potential locations. (d) Non-void locations. (e) Refined locations. (f) Final locations on 2D image.

6.2 Object Search

As Section 4.1 states, to interpret a video, the objects in each video frame should first be searched for. We search for the objects in each video frame with a sliding-window strategy. In 2D images, the conventional sliding-window strategy often densely searches many invalid locations, where there is no object, like the background. This process is inefficient considering the massive amount of frames in videos. Differently, we slide the 3D window box at valid locations in 3D space. Then the 3D window is projected onto the 2D image to extract the appearance feature. This strategy largely reduces the search space and thus increases the efficiency. We implement this in three steps.

Step 1: proposing potential locations. In 3D space, we propose potential object locations inside a 3D box near the human body parts, as the green cubic area shows in Fig. 10 (c). The 3D box size and the step between the locations are determined in experiments empirically.

Step 2: removing void locations. We further remove the locations at which the 3D spatial occupancy is void. Given the 3D locations from Step 1, we put a 3D box at each location and check to see if the box contains cloud points. If the number of the points is smaller than a threshold, the location is considered void and discarded. The box size and the threshold are set empirically. After this step, the locations with 'real object entity' remain and are shown in Fig. 10(d).

Step 3: refining valid locations. In this step, the object locations are refined using the human-object geometric compatibility probabilities shown in Fig. 10 (b). Those locations with low compatibility probability are pruned. The result is shown in Fig. 10 (e). The resulting 3D locations are transformed to 2D locations in the image plane, where the objects are searched for, as shown in Fig. 10 (f).

Through the three steps, most invalid locations are removed and the object search space is greatly reduced.

7 Experiments

7.1 Experimental Dataset

We tested the proposed method on three challenging datasets.

Multiview RGB-D Event Dataset. We collected a large multiview RGB-D event dataset. The videos were captured by 3 stationary Kinect cameras simultaneously at different

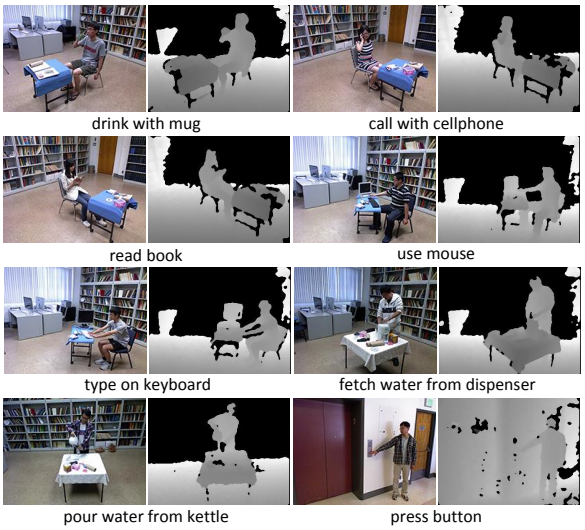

Fig. 11: Samples of Multiview RGB-D Event Dataset

viewpoints around the human. Each video frame includes an RGB image at a resolution of 640×480 pixels, a depth image, and a 3D human skeleton. The events were performed by 8 actors in indoor scenes, such as the hallway and the library. Each actor performed the events with different object instances and various styles. It includes 8 event categories: *drink with mug*, *call with cellphone*, *read book*, *use mouse*, *type on keyboard*, *fetch water from dispenser*, *pour water from kettle*, and *press button*, which involve 11 object classes: *mug*, *cellphone*, *book*, *mouse*, *keyboard*, *dispenser*, *kettle*, *button*, *monitor*, *chair*, and *desk*. Fig. 11 shows some RGB and depth frames.

We manually segmented the long videos into short sequences with each segment containing one event from beginning to end. The dataset contains 3,815 event videos and 383,036 RGB-D frames. Each event category has about 477 video instances on average.

Our dataset has several characteristics which make it challenging. Firstly, it is multiview. We used three cameras to capture the videos. However, due to various styles of the actors' actions, the number of the viewpoints of each event is much larger than three. Secondly, the events involve various objects and the human skeletons are very noisy. Thirdly, the data has large variance due to the particular style of each actor to perform an event. Table 1 shows a comparison of several well-known event datasets.

DailyActivity3D Dataset (Daily3D) [14] contains 320

Subject	MV	3D	TN	AN	AL
CMUHOI [34]			54	9	110
Daily3D [14]		✓	320	20	195
MSRA3D [61]		✓	567	28	42
Our Dataset	✓	✓	3815	477	100

TABLE 1: Dataset comparison. MV, multiview; TN, total video number; AN, average number of the videos of each event category; AL, average length (frame) of each video.

RGB-D videos and 3D human joint sequences of 16 daily activity classes: *drink, eat, read book, call cellphone, write on a paper, use laptop, use vacuum cleaner, cheer up, sit still, toss paper, play game, lay down on sofa, walk, play guitar, stand up,* and *sit down*. These activities involve contextual objects, such as *mug, food package, cellphone, laptop*, etc. This dataset does not provide object labels of ground truth. To evaluate our method, we manually labeled the object classes and regions in each frame of the videos with contextual objects.

This dataset is very challenging. Firstly, the data is very noisy due to occlusion and low resolution. Secondly, the activities have huge variances because each subject performs activities in different ways.

MSR-Action3D Dataset (MSR3D) [61] contains 567 sequences performed by 10 subjects of 20 action classes: *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side-boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pick up & throw*. It contains depth frames from which the 3D joints are extracted.

This dataset has a large number of action classes, and includes many subtle and highly-similar actions, such as *draw x, draw tick, draw circle, tennis swing*, and *tennis serve*. These make the dataset very challenging.

7.2 Event Recognition

Multiview RGB-D Event Dataset. We use two classical event recognition methods as baselines - Motion Template (MT) [1] and Hidden Markov Model (HMM) [24]. Since the data captured by Kinect is noisy, the feature proposed in [1] is not available for our input. We used the 3D joint points on arms as the input frame features for MT and HMM. For MT, we trained a motion template for each event category, and matched the testing samples with the templates with the dynamic temporal warping distance. For HMM, we trained a Hidden Markov Model for each event category. We also computed the recognition accuracy of 4DH, which is the same framework as 4DHOI except that it only uses the human pose information as input and omits the information of contextual object interaction.

Table 2 shows that the performance of our method is better than the other three methods. It outperforms other methods in 6 categories of all 8 event categories, and improves the overall accuracy, which demonstrates the strength of our joint modeling and inference.

Fig. 12 shows the confusion matrices of 4DH and 4DHOI. The comparison between 4DH and 4DHOI proves the effect of the human-object interaction on event recognition. For example, the human body movements in the events *drink with mug* and *call with cellphone* are highly similar. It is hard to distinguish them with only the human pose information. By incorporating the object information of *mug* and *cellphone*, the two events are better distinguished. Consider another event - *pour water from kettle*, it is complex in the temporal structures and human body movements because it involves the movements of the two arms as well as the coordination between them. The object *kettle* has special appearance information and only exists in the event *pour water from kettle*, which makes it provide strong support to this event. When incorporating the information of *kettle*, the performance is significantly improved.

Event	MT [1]	HMM [24]	4DH	4DHOI
drink with mug	0.51	0.62	0.64	**0.85**
call with cellphone	0.32	0.41	**0.64**	0.63
read book	0.83	0.73	0.98	**1.00**
use mouse	0.84	0.87	0.98	**1.00**
type on keyboard	0.77	0.89	**0.98**	**0.98**
fetch water from dispenser	0.82	0.76	0.90	**0.95**
pour water from kettle	0.68	0.67	0.86	**0.98**
press button	0.73	**0.99**	0.97	0.95
Overall	0.69	0.74	0.87	**0.92**

TABLE 2: Event recognition accuracy comparison on Multiview RGB-D Event Dataset.

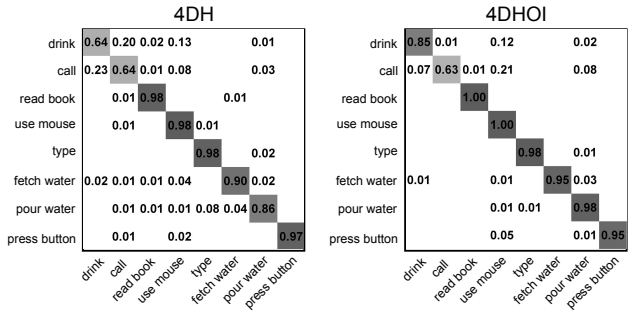

Fig. 12: Confusion matrix of 4DH and 4DHOI on Multiview RGB-D Event Dataset. The event names are simplified.

Method	DTW [1]	ROP [62]	ALEJ [14]	4DH	4DHOI
Accuracy	0.54	0.64	0.74	0.74	**0.80**

TABLE 3: Event recognition accuracy comparison on DailyActivity3D Dataset.

DailyActivity3D Dataset. On this dataset, we compare our method with three other methods - Dynamic Temporal Warping (DTW) [1], Random Occupancy Pattern (ROP) [62], and Actionlet Ensemble on Joint Features (ALEJ) [14]. ALEJ was proposed in the same work as the dataset [14]. Table 3 shows a comparison of recognition accuracy, where the results of DTW, ROP and ALEJ are cited from [14].

Our 4DH and 4DHOI achieve the accuracy of 0.74 and 0.80, respectively, while DTW is 0.54, ROP is 0.64, and ALEJ is 0.74. This demonstrates the strength of our method, especially considering that our method can not only recognize the event, but also segment the sequence and localize objects simultaneously, while the other three methods were particularly designed for action recognition.

MSR-Action3D Dataset. Since this dataset contains no object interactions, we compare our 4DH method with five other recently proposed methods which use skeleton information. Table 4 shows a comparison of overall recognition accuracy, where the results of other baseline methods are cited from [14].

Method	DTW [1]	HMM [2]	AG [61]	MIJ [63]	ALEJ [14]	4DH
Accuracy	0.54	0.63	0.75	0.47	0.69	**0.83**

TABLE 4: Event recognition accuracy comparison on MSR-Action3D Dataaset.

Metric	EMC	SC [3]	ACA [4]	HACA [4]	Our 4DH
Segment Accuracy	0.75	0.61	0.69	0.69	**0.84**
Frame Accuracy	0.74	0.52	0.63	0.64	**0.76**

TABLE 5: Comparison of sequence segmentation algorithms with two accuracy metrics.

Our method achieves the accuracy of 0.83 on this dataset. Action Graph (AG) [61], which released the MSR-Action3D Dataset, obtains the accuracy of 0.75. The other methods, Dynamic Temporal Warping (DTW) [1], Hidden Markov Models (HMM) [2], Most Informative Joints (MIJ) [63], and Actionlet Ensemble with Absolute Joint Positions (ALEJ) [14], achieve the accuracy of 0.54, 0.63, 0.47, and 0.69, respectively. This comparison proves the strength of our method. It also demonstrates that our model can effectively handle actions with subtle structures.

7.3 Sequence Segmentation

We tested our sequence segmentation method on 300 long video sequences which were generated by concatenating the videos from the Multiview RGB-D Event Dataset. The event number, categories, and actors in each sequence are random. Each sequence contains one to ten event instances, and the length ranges from 29 to 1409 at a frame rate of 15 fps.

Our segmentation data is challenging for several reasons. Firstly, the event instances have complex temporal structures, and the human motion in an event is not coherent. For example, the human actions in the initial and middle steps of the event *pour water from kettle* are very different, which may lead to false segmentation of the two phases. Secondly, some similar events occur in one consecutive sequence, and some events occur many times in one sequence. Thirdly, some sequences contain only one event instance, which tests the generality of the segmentation algorithm and increases the data difficulty.

To comprehensively test the segmentation algorithms, we use two criteria for evaluation. The first criterion is frame accuracy, which is defined as the ratio between the number of correctly labeled frames and the number of all testing frames. The second criterion is the segment accuracy defined in [4] and is used commonly in the literature. It evaluates the algorithm by computing the segment correspondence between the testing result and the ground truth. The frame accuracy measures the local similarity between the testing result and the ground truth while the segment accuracy measures the global similarity.

We compare our 4DH model with four methods - EM Classification (EMC), Spectral Clustering (SC) [3], Aligned Cluster Analysis (ACA) [4], and Hierarchical Aligned Cluster Analysis (HACA) [4]. We used 4DH not 4DHOI because the four baselines used only human motion information, not contextual objects. To have a fair comparison, we dropped

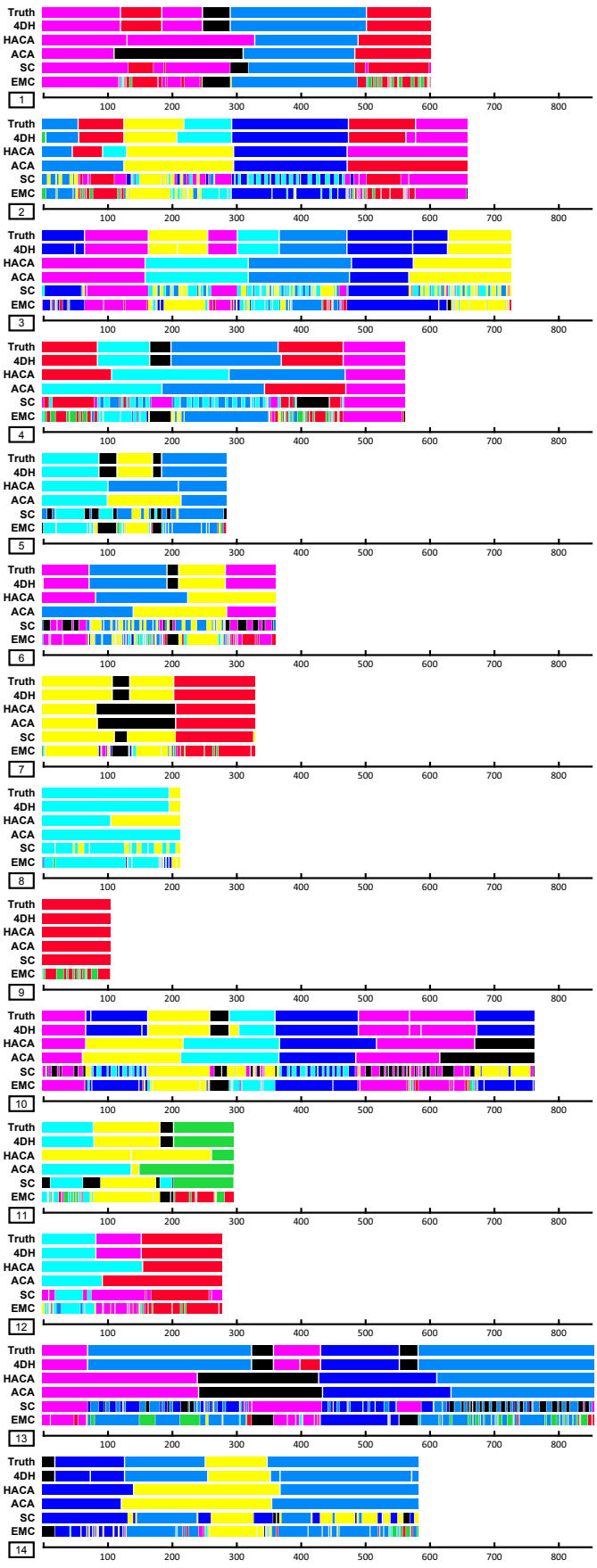

Fig. 13: Sequence segmentation comparison. Each row is a video sequence. Each color denotes an event.

	mug	cellphone	book	mouse	keyboard	dispenser	kettle	button	monitor	chair	desk
HOG[5]	0.22	0.44	0.18	0.12	0.30	0.58	0.25	0.37	0.43	0.88	0.97
RDH[6]	0.29	0.48	0.19	0.13	**0.75**	0.69	0.38	**0.43**	0.69	0.96	0.97
4DHOI	**0.45**	**0.51**	**0.26**	**0.20**	**0.75**	**0.79**	**0.40**	0.39	**0.76**	**0.97**	**0.98**

TABLE 6: Average precision (AP) comparison on Multiview RGB-D Event Dataset.

object information. EMC recognizes each frame independently without temporal context. It trains the model parameters with the EM algorithm and recognizes an event by Bayesian classification, where the prior distribution of each event category is assumed uniform. SC segments the sequence by maximizing the inter-cluster distance and minimizing the intra-cluster distance. It is one of the most widely-used clustering algorithms. ACA incorporates the temporal order of action frames into clustering and HACA is similar to ACA but solves the model in a hierarchical manner. We used the implementation of SC, ACA, and HACA released in [4].

Table 5 shows the segmentation accuracy of different methods. Fig. 13 visualizes some segmentation results of the five methods. Recognizing each frame independently produces many small incoherent clips, as EMC shows in Fig. 13. Our 4DH incorporates the temporal structures of events, which provides the contextual and duration information among successive frames. Therefore, it produces coherent segmentation and better performance than EMC. SC [3], ACA [4], and HACA [4] are unsupervised clustering methods, where the real event number in each sequence needs to be given. Our method does not need these parameters to be given, which is advantageous in real applications.

7.4 Object Localization

Multiview RGB-D Event Dataset. We use average precision (AP) as the criterion for evaluating object localization. In each frame, we obtain many positive proposal locations, with which we compute the precision, recall and AP. This criterion uses more localized instances than the localization accuracy criterion in the conference paper [13]. We compare our method with the detection methods using the HOG features [5] and RDH using the RGB-D HOG features [6], which detect objects in a sliding-window way. The originally detected boxes of these methods are processed with the non-maximum suppression. Table 6 shows the AP comparison of each object class. Fig. 14 shows the overall AP and precision-recall curves. Fig. 15 visualizes some examples.

Those results demonstrate the strength of our model. Our method achieves the highest AP in 10 of all 11 object categories. It also largely improves the overall AP compared to the other two methods. The objects involved in the events present large appearance variance. Some objects have non-rigid structures, such as the book. Some objects move with human actions and present different directions, sizes, and views in the motion, such as the mug. Some small objects are often occluded by the human body in the action, such as the cellphone and the mouse. The HOG and RDH methods localize objects with appearance information. Non-rigid structure, movement, occlusion, view variation, and

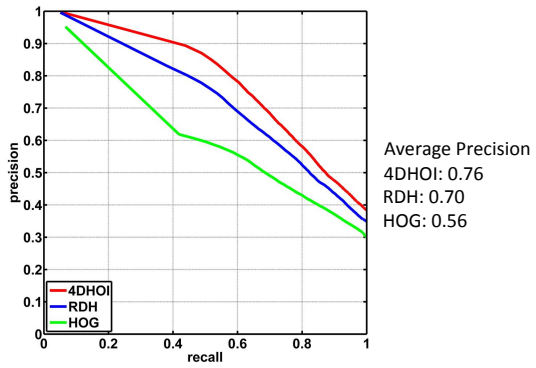

Fig. 14: Overall precision-recall comparison on Multiview RGB-D Event Dataset.

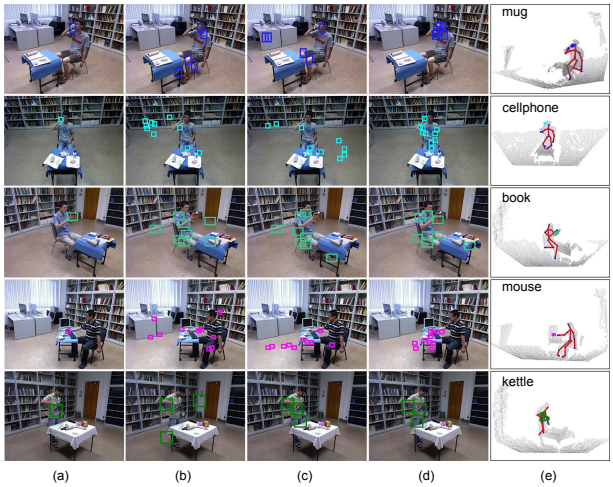

Fig. 15: Examples of object localization on Multiview RGB-D Event Dataset. For each object category, we show the same number of instances with top scores. (a) Ground truth. (b) HOG. (c) RDH. (d) 4DHOI in RGB images. (e) 4DHOI in 3D point clouds. We show one instance for clarity.

low resolution lead to the low AP in Table 6. Human action information facilitates object localization by using temporal and human body contexts, and thus improves the accuracy.

For the keyboard and the button, the performance of 4DHOI is equal to or lower than the appearance-based methods. This is for two reasons. Firstly, the keyboard and the button are so thin that the Kinect camera did not accurately capture their depth information. Thus the 3D geometric compatibility relations between the human and the objects are not accurate. Secondly, these objects are almost static in the video, and thus do not fit the action model well when the human body is constantly moving with noise.

DailyActivity3D Dataset. We compare our method with HOG [5] and RDH [6] on the interacting objects of 9 classes - *mug*, *food packet*, *book*, *cellphone*, *laptop*, *cleaner*, *paper*, *gamebox*, and *guitar*. Table 7 shows the overall AP comparison.

The resolution of objects in this dataset is low, and many objects are corrupted by noise. Thus, the appearance-based methods of HOG and RDH obtain the AP of 0.41 and 0.51, respectively. When incorporating the human interactions, our 4DHOI method achieves the average precision of 0.70. It significantly improves the performance.

Method	HOG [5]	RDH [6]	4DHOI
Average Accuracy	0.41	0.51	0.70

TABLE 7: The overall average precision (AP) comparison on DailyActivity3D Dataset.

8 Discussion and Future Work

In this paper, we presented a 4D human-object interaction model for joint event recognition, sequence segmentation, and contextual object localization in RGB-D videos. The 4DHOI model represents the geometric, temporal, and semantic relations in daily events involving human-object interactions. The experiments demonstrated improved performance on challenging datasets for all the three tasks. Several issues still need to be investigated in future work.

Firstly, the overall precision in object localization is still unsatisfactory. This is mainly due to the noise of the 3D human skeletons and the depth data from the Kinect camera. The large noise and holes in the depth data weaken the HOG features and therefore influence the object detection.

Secondly, modeling the human-object interaction in concurrent actions [16] is another interesting but challenging problem. Our 4DHOI model can potentially be applied to the concurrent actions because of its part-based definition of features and relations. However, in concurrent actions, in addition to the human-object interaction, complex interactions often exist among different actions [16]. Such complicated interactions of multi-types need to be further studied.

Thirdly, defining objects and scenes with multi-source information is an interesting problem [8], [9], [10]. This paper discusses several types of information, such as appearance, affordance, and coherence. Quantitatively describing and measuring the weights of different factors are significant tasks of our future work.

The 4DHOI model can be further used for inferring scene functionality [9] and object affordance [15]. It can potentially be used for recovering 3D poses from RGB videos. These tasks will be the topics of our future work.

Acknowledgments

Ping Wei and Nanning Zheng thank the support of grants: Key Program of NSFC 61231018, NSFC 61503297, and 973 Program of China 2012CB316402. Yibiao Zhao and Song-Chun Zhu thank the support of ONR MURI N00014-10-1-0933, and DARPA MSEE FA 8650-11-1-7149.

References

[1] M. Müller and T. Röder, "Motion templates for automatic classification and retrieval of motion capture data," in *Proc. ACM SIGGRAPH/Eurographics Symp. on Comput. Animat.*, 2006, pp. 137–146.

[2] F. Lv and R. Nevatia, "Recognition and segmentation of 3-D human action using HMM and multi-class adaboost," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 359–372.

[3] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: analysis and an algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 849–856.

[4] F. Zhou, F. D. la Torre, and J. K. Hodgins, "Hierarchical aligned cluster analysis for temporal clustering of human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 582–596, 2013.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.

[6] K. Lai, L. Bo, X. Ren, and D. Fox, "Detection-based object labeling in 3D scenes," in *Proc. Int. Conf. Robot. Autom.*, 2012, pp. 1330–1337.

[7] J. J. Gibson, "The theory of affordances," in *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*. Lawrence Erlbaum, 1977, pp. 67–82.

[8] H. Grabner, J. Gall, and L. V. Gool, "What makes a chair a chair?" in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1529–1536.

[9] Y. Zhao and S.-C. Zhu, "Scene parsing by integrating function, geometry and appearance models," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3119–3126.

[10] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert, "From 3D scene geometry to human workspace," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1961–1968.

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1297–1304.

[12] C. M. Bishop, *Pattern Recognit. Mach. Learn.* Springer, 2006.

[13] P. Wei, Y. Zhao, N. Zheng, and S.-C. Zhu, "Modeling 4D human-object interactions for event and object recognition," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 3272–3279.

[14] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, 2014.

[15] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The Int. J. of Robot. Research*, vol. 32, no. 8, pp. 951–970, 2013.

[16] P. Wei, N. Zheng, Y. Zhao, and S.-C. Zhu, "Concurrent action detection with structural prediction," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 3136–3143.

[17] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu, "Cross-view action modeling, learning and recognition," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2649–2656.

[18] S. Sadanand and J. J. Corso, "Action bank: a high-level representation of activity in video," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1234–1241.

[19] A. Bargi, R. Y. D. Xu, and M. Piccardi, "An online hdp-HMM for joint action segmentation and classification in motion capture data," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. Workshop*, 2012, pp. 1–7.

[20] M. Hoai, Z.-Z. Lan, and F. D. la Torre, "Joint segmentation and classification of human actions in video," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3265–3272.

[21] A. Ali and J. K. Aggarwal, "Segmentation and recognition of continuous human activity," in *Proc. Workshop on Detection and Recognition of Events in Video*, 2001, pp. 28–35.

[22] Q. Shi, L. Cheng, L. Wang, and A. J. Smola, "Human action segmentation and recognition using discriminative semi-Markov models," *Int. J. Comput. Vis.*, vol. 93, pp. 22–32, 2011.

[23] J. Yuan, Z. Liu, and Y. Wu, "Discriminative video pattern search for efficient action detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1728–1743, 2011.

[24] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[25] K. Tang, L. Fei-Fei, and D. Koller, "Learning latent temporal structure for complex event detection," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1250–1257.

[26] B. Z. Yao, B. X. Nie, Z. Liu, and S.-C. Zhu, "Animated pose templates for modelling and detecting human actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 436–452, 2014.

[27] M. Pei, Z. Si, B. Z. Yao, and S.-C. Zhu, "Learning and parsing video events with goal and intent prediction," *Comput. Vis. and Image Understanding*, vol. 117, pp. 1369–1383, 2013.

[28] N. N. Vo and A. F. Bobick, "From stochastic grammar to bayes network: probabilistic parsing of complex activity," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2641–2648.

[29] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *Proc. Int. Conf. Robot. Autom.*, 2012, pp. 842–849.

[30] L. Yang, N. Zheng, M. Chen, Y. Yang, and J. Yang, "Categorization of multiple objects in a scene using a biased sampling strategy," *Int. J. Comput. Vis.*, vol. 105, pp. 1–18, 2013.

[31] K. Murphy, A. Torralba, D. Eaton, and W. Freeman, "Object detection and localization using local and global features," in *Toward Category-Level Object Recognition*, ser. Lecture Notes in Comput. Sci., 2006, vol. 4170, pp. 382–400.
[32] C. Desai, D. Ramanan, and C. C. Fowlkes, "Discriminative models for multi-class object layout," *Int. J. Comput. Vis.*, vol. 95, pp. 1–12, 2011.
[33] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–8.
[34] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: using spatial and functional compatibility for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1775–1789, 2009.
[35] D. A. Norman, *The Psychology of Everyday Things*. Basic Books, 1988.
[36] M. T. Turvey, "Affordances and prospective control: an outline of the ontology," *Ecological Psychology*, vol. 4, no. 3, pp. 173–187, 1992.
[37] M. Steedman, "Formalizing affordance," in *Proc. The 24th Annual Meeting of the Cognitive Science Society*, 2002, pp. 834–839.
[38] T. A. Stoffregen, "Affordances as properties of the animal-environment system," *Ecological Psychology*, vol. 15, no. 2, pp. 115–134, 2003.
[39] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: a new formalization of affordances towards affordance-based robot control," *Adaptive Behavior*, vol. 15, no. 4, pp. 447–472, 2007.
[40] J. Gall, A. Fossati, and L. van Gool, "Functional categorization of objects using real-time markerless motion capture," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1969–1976.
[41] M. Marszałek, I. Laptev, and C. Schmid, "Actions in context," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit*, 2009, pp. 2929 – 2936.
[42] B. Packer, K. Saenko, and D. Koller, "A combined pose, object, and feature model for action understanding," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1378–1385.
[43] A. Prest, V. Ferrari, and C. Schmid, "Explicit modeling of human-object interactions in realistic videos," INRIA, Tech. Rep., 2011.
[44] A. Prest, C. Schmid, and V. Ferrari, "Weakly supervised learning of interactions between humans and objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 601–614, 2012.
[45] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, "A scalable approach to activity recognition based on object use," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–8.
[46] B. Yao and L. Fei-Fei, "Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1691–1703, 2012.
[47] D. J. Moore, I. A. Essa, and M. H. Hayes, "Exploiting human actions and object context for recognition tasks," in *Proc. Int. Conf. Comput. Vis.*, 1999, pp. 80–86.
[48] V. G. Kim, S. Chaudhuri, L. Guibas, and T. Funkhouser, "Shape2pose: human-centric shape analysis," *ACM Trans. on Graph.*, vol. 33, no. 4, pp. 120:1–120:12, 2014.
[49] D. F. Fouhey, V. Delaitre, A. Gupta, A. A. Efros, I. Laptev, and J. Sivic, "People watching: human actions as a cue for single view geometry," *Int. J. Comput. Vis.*, vol. 110, pp. 259–274, 2014.
[50] V. Delaitre, D. F. Fouhey, I. Laptev, J. Sivic, A. Gupta, and A. A. Efros, "Scene semantics from long-term observation of people," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 284–298.
[51] H. Kjellström, J. Romero, and D. Kragić, "Visual object-action recognition: inferring object affordances from human demonstration," *Comput. Vis. and Image Understanding*, vol. 115, pp. 81–90, 2011.
[52] E. E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter, "Learning the semantics of object-action relations by observation," *The Int. J. of Robot. Research*, vol. 30, no. 10, pp. 1229–1249, 2011.
[53] F. Wörgötter, E. E. Aksoy, N. Krüger, J. Piater, A. Ude, and M. Tamosiunaite, "A simple ontology of manipulation actions based on hand-object relations," *IEEE Trans. Auton. Mental Develop.*, vol. 5, no. 2, pp. 117–134, 2013.
[54] Y. Yang, C. Fermüller, and Y. Aloimonos, "Detection of manipulation action consequences (mac)," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2563–2570.
[55] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1848–1852, 2007.
[56] S.-C. Zhu and D. Mumford, "A stochastic grammar of images," *Found. and Trends in Comput. Graph. and Vis.*, vol. 2, no. 4, pp. 259–362, 2006.
[57] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, 1999, pp. 61–74.
[58] C. Tillmann and H. Ney, "Word reordering and a dynamic programming beam search algorithm for statistical machine translation," *Computational Linguistics*, vol. 29, pp. 97–133, 2003.
[59] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S.-C. Zhu, "Beyond point clouds: scene understanding by reasoning geometry and physics," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3127–3134.
[60] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Manhattan-world stereo," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1422–1429.
[61] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. Workshop*, 2010, pp. 9–14.
[62] J. Wang, J. Yuan, Z. Chen, and Y. Wu, "Spatial locality-aware sparse coding and dictionary learning," in *Proc. Asian Conf. Mach. Learn.*, 2012, pp. 491–505.
[63] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (smij): a new representation for human skeletal action recognition," *J. of Visual Commun. and Image Representation*, vol. 25, pp. 24–38, 2014.

Ping Wei received his BE degree and PhD degree from Xi'an Jiaotong University, China. He is currently an assistant professor with the Institute of Artificial Intelligence and Robotics at Xi'an Jiaotong University. From Nov. 2011 to Apr. 2013, he was a visiting PhD student at the VCLA Center of UCLA. His research interests include computer vision, machine learning, and cognition modeling. He is a member of IEEE.

Yibiao Zhao received a PhD degree from University of California, Los Angeles. His research interests include computer vision, cognitive modeling, cognitive robotics, statistical learning and inference. He has been working on scene parsing integrating functionality, geometry and appearance. He is the co-chair of the series of Int'l Workshops on Vision Meets Cognition: Functionality, Physics, Intents and Causality at CVPR 2014 and CVPR 2015. He is a member of IEEE.

Nanning Zheng received a PhD degree from Keio University, Japan, in 1985. He is currently a professor and the director of the Institute of Artificial Intelligence and Robotics at Xi'an Jiaotong University. His research interests include computer vision, pattern recognition, computational intelligence, image processing, and hardware implementation of intelligent systems. Since 2000, he has been the Chinese representative on the Governing Board of the International Association for Pattern Recognition. He became a member of the Chinese Academy of Engineering in 1999. He is a Fellow of IEEE.

Song-Chun Zhu received a PhD degree from Harvard University, and is a professor with the Department of Statistics and the Department of Computer Science at UCLA. His research interests include computer vision, statistical modeling and learning, cognition and AI, and visual arts. He received a number of honors, including the Marr Prize in 2003 with Z. Tu et. al. on image parsing,the Aggarwal prize from the Int'l Association of Pattern Recognition in 2008, twice Marr Prize honorary nominations in 1999 for texture modeling and 2007 for object modeling with Y.N. Wu et al., a Sloan Fellowship in 2001, the US NSF Career Award in 2001, and the US ONR Young Investigator Award in 2001. He is a Fellow of IEEE.