The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Tracking Occluded Objects and Recovering Incomplete Trajectories by Reasoning about Containment Relations and Human Actions

Wei Liang,[1,2] **Yixin Zhu,**[2] **Song-Chun Zhu**[2]
liangwei@bit.edu.cn, yixin.zhu@ucla.edu, sczhu@stat.ucla.edu
[1]Beijing Laboratory of Intelligent Information Technology, Beijing Institute of Technology, China
[2]Center for Vision, Cognition, Learning, and Autonomy, University of California, Los Angeles, USA

Abstract

This paper studies a challenging problem of tracking severely occluded objects in long video sequences. The proposed method reasons about the containment relations and human actions, thus infers and recovers occluded objects identities while contained or blocked by others. There are two conditions that lead to incomplete trajectories: i) *Contained*. The occlusion is caused by a containment relation formed between two objects, *e.g.*, an unobserved laptop inside a backpack forms containment relation between the laptop and the backpack. ii) *Blocked*. The occlusion is caused by other objects blocking the view from certain locations, during which the containment relation does not change. By explicitly distinguishing these two causes of occlusions, the proposed algorithm formulates tracking problem as a network flow representation encoding containment relations and their changes. By assuming all the occlusions are not spontaneously happened but only triggered by human actions, an MAP inference is applied to jointly interpret the trajectory of an object by detection in space and human actions in time. To quantitatively evaluate our algorithm, we collect a new occluded object dataset captured by Kinect sensor, including a set of RGB-D videos and human skeletons with multiple actors, various objects, and different changes of containment relations. In the experiments, we show that the proposed method demonstrates better performance on tracking occluded objects compared with baseline methods.

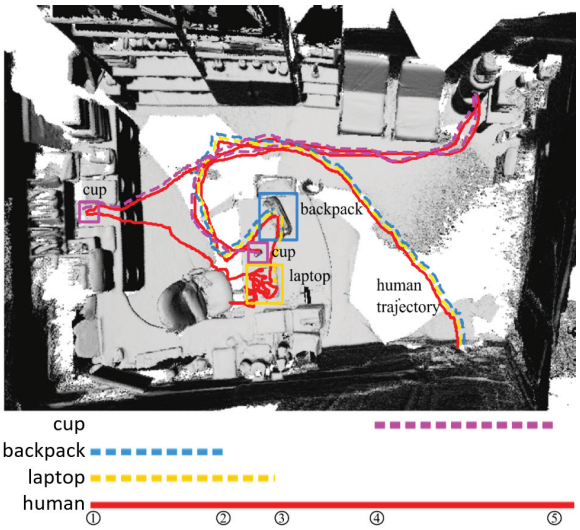


Figure 1: A scenario for tracking occluded objects in an indoor scene. The dashed lines represent the inferred trajectories and different colors indicate different objects in the scene. By explicitly reasoning about containment relations, the proposed algorithm is capable of recovering full trajectories of objects even they are contained or occluded by other objects in the video.

Introduction

We study the problem of tracking occluded objects during human daily activities in cluttered scenes, such as packing, playing, working, *etc*. Figure 1 shows an example of a daily indoor scenario captured by a RGB-D sensor: an agent ① enters a room; ② puts down her backpack; ③ takes a laptop out of the backpack and puts it on the table; ④ grabs a cup, fetches some water from a water dispenser; ⑤ sits back and puts down the cup next to her. During the course of this event, objects disappear and then re-appear frequently.

Tracking objects in such scenarios is a challenging problem due to severe occlusions caused by two conditions:

- **Contained.** The occlusion is caused by a *new containment relation formed* between two objects, *e.g.*, a person puts a laptop into a bag, which is view-independent;
- **Blocked.** The occlusion is caused by other objects observed from certain camera views, in which the *containment relations unchanged*, *e.g.*, a laptop is sitting in front of a cup, which blocks the view of the cup from the current camera view and is view-dependent.

We argue such problem is not merely a vision task compared to traditional visual tracking tasks, which primarily focuses on reliable object detectors and data association methods. Instead, significant reasoning processes are involved. To address this problem, we believe an explicit model of relations among objects as well as relations between objects and agents are needed.

The proposed framework is shown in Figure 2. Given a RGB-D video with extracted human skeleton sequence in a scene, state-of-the-art detection algorithms are applied to detect regions of interest of each object and human actions over

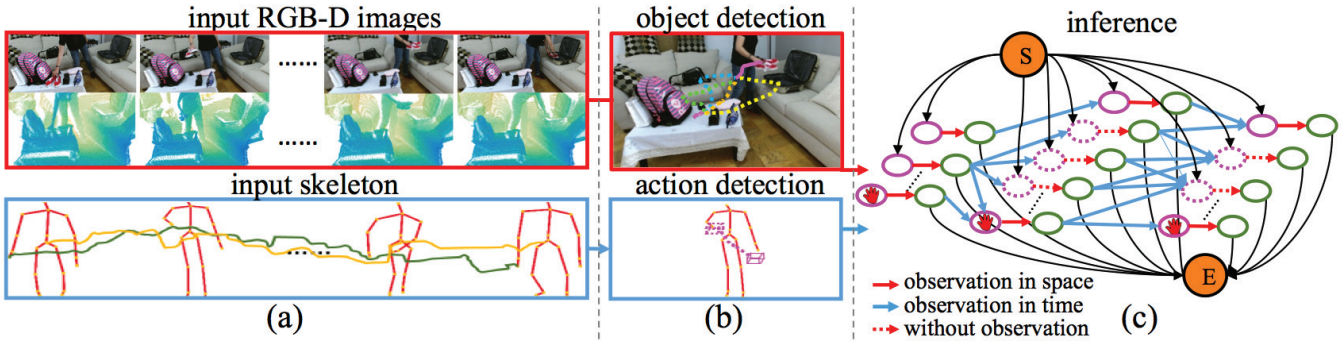


Figure 2: The framework of the proposed method. (a) Sensor input: a sequence of RGB-D images and human skeleton captured by a Kinect sensor. (b) Off-the-shelf state-of-the-art object detection and human action detection algorithms were applied to extract the object location and human actions per frame. (c) Inference on a network flow representation. The solid red lines denote the observations in space. The dashed red lines denote that the present state of the object is hidden and there is no observation. The blue lines denote the observations in time. S and E are the start and the end of the trajectories, respectively. A dynamic programming scheme is applied to search, optimize and recover the complete trajectory of each object.

time; the detection results serve as the initial proposals and the input to our algorithm. We pose the problem of recovering object trajectories as Maximizing a Posteriori (MAP) problem using a network flow representation, in which a trajectory of an object is jointly interpreted and constrained by both object detection and containment relations in space as well as human actions over time. A dynamic programming scheme is applied to search, optimize and recover the complete trajectory of each object.

This paper makes three contributions:

1. We propose a method to recover incomplete trajectories of objects by taking account of containment relations and two causes of occlusions: contained and blocked.
2. We assume that human action is the only cause that leads to occlusions and object status changes, and use it as a constraint to interpret trajectories of objects over time.
3. We introduce a new dataset including a set of RGB-D videos of human interacting with occluded objects.

Related Work

Spatial Reasoning. Spatial reasoning plays an essential role in human daily life. Although quantitative approaches can provide the most precise information, numerical information is often unnecessary or unavailable at human level. In computer vision, quantitative approaches usually study objects tracking problem, of which the literature is too expansive to survey here; we refer readers to recent survey and benchmark (Wu, Lim, and Yang 2015; Smeulders et al. 2014; Wu, Lim, and Yang 2015). Here, we focus on spatial reasoning methods related to the presented work.

As a typical example, container has been used to study spatial reasoning problem (Bredeweg and Forbus 2003; Frank 1996). Using physical-based simulations, (Liang et al. 2015) evaluated human cognition of containing relations through human studies. (Davis, Marcus, and Chen 2013) developed a knowledge base for qualitative reasoning about containers, expressed in a first-order language of time, geometry, objects, histories, and events. Some exemplary tasks include i) computational approaches for reasoning about liquid transfer (Kubricht et al. 2016; Yu, Duncan, and Yeung 2015; Mottaghi et al. 2017), ii) reason about containability and containment relations (Liang et al. 2016; Yu, Duncan, and Yeung 2015; Wang, Liang, and Yu 2017), and iii) occlusion modeling and reasoning (Eichner and Ferrari 2010; Enzweiler et al. 2010; Wojek et al. 2011). Compared to prior work, we integrate qualitative and quantitative approach and explicitly model the occlusions by containment relations when an object is contained or blocked by others.

Detection using Context. Context has been widely explored in human-object interactions (HOI) and multi-object tracking. Some typical approaches and setups include: i) Learning deformable action templates (Yao et al. 2014), actionlet (Wang, Liu, and Wu 2014) and animated pose templates (Yao et al. 2014). ii) Combining spatial and functional constraints between human and objects (Gupta, Kembhavi, and Davis 2009; Yang, Wu, and Hua 2009; Wei et al. 2013). iii) Task-oriented action recognition (Zhu, Zhao, and Zhu 2015) and utility learning (Zhu et al. 2016; Shukla et al. 2017), including complex cooking tasks (Rohrbach et al. 2012; Aksoy et al. 2011)

Different from the literature in context modeling, we explicitly model human action as a context cue. Specifically, we assume that human action is the only cause that leads to the changes of containment relations, and use the human action as a constraint to improve the tracking.

Although some recent work adopted deep neural networks to extract contexts for object detection and tracking, these data-driven feedforward methods have well-known problems: i) They are black-box models that cannot be explained and only applicable with supervised training by fitting the typical context of the object, thus difficult to generalize to new tasks. ii) Lacking explicit representation to handle occlusions, low resolution, and lighting variations—there are millions of ways to occlude an object in a given im-

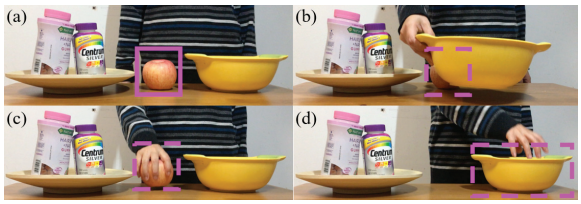


Figure 3: Two causes of occlusions. i) Blocked: An apple (a) can be detected at the beginning, but later (b) becomes occluded by a bowl. ii) Contained: an apple is contained by a person (c) and a bowl (d), respectively.

age (Wang et al. 2017), making it impossible to have enough data for training and testing such black box models. In this paper, we go beyond passive recognition by reasoning about time-varying containment relations.

Probabilistic Formulation

A key concept in the present paper is "containment relation". An object which contains or holds another object can serve as a container, forming containment relation with the object it contains. For instance, when a laptop is inside a backpack, a containment relation is formed between the laptop and the backpack, where the backpack plays the role of container.

We make the following assumptions for containers and containment relations:

- When an object is contained by a container, the object will inherit the same trajectory from its container. For example, consider a case that a laptop is inside a backpack. If a person carries the backpack around, the laptop will move together with the backpack, sharing the same trajectory.
- The containment relation is a partially ordered relation constrained by the volume of the object and its container, *i.e.*, if a container's volume is smaller than an object's, the object cannot be contained by this container.
- An object can only be contained directly by one container.

Suppose there are K objects in a scene. Our goal is to recover trajectories of all K objects $T = \{T^1, T^2, \cdots, T^K\}$ from a RGB-D image sequence $I = \{I_1, I_2, \cdots, I_\tau\}$, where τ is the length of the image sequence. T^k is defined as an ordered set of object states $T^k = \{x_1^k, x_2^k, \cdots, x_\tau^k\}$, where x_t^k is the state of the kth object in space at time t: $x_t^k = (l_t^k, c_t^k)$, where l_t^k is the location of the kth object at time t and $c_t^k \in \{1, 2, \cdots, K\}$ is an object index, representing the inferred container of kth object at time t.

Spatial Hypotheses by Containment Relations

At each frame t, instead of purely relying on detection results, our algorithm further proposes two types of hypotheses generated based on the possible causes of occlusion. These two hypotheses provides additional cues, essentially competing with the detection results. As a result, such extra info recovered by the containment relations could later help overcome the miss or wrong detection described in the next section. The two types of hypotheses are:

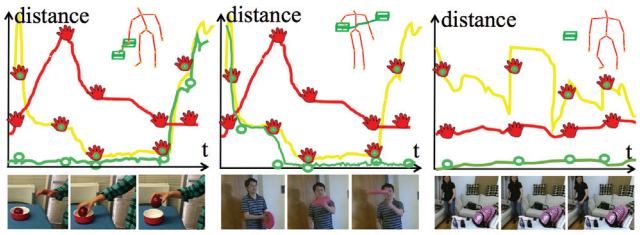


Figure 4: Human action features at time t in a sliding window with the length of 2ϵ. The red line represents the distance between the hand and the spine of the person. The yellow line represents the distance change between the hand and the object. The green line represents the location change of the object in the current sliding window.

Contained. In these situations, occlusion happens due to forming new containment relations as shown in Figure 3 (c) and 3 (d), where hypotheses are shown in dashed box. Formally, suppose such occlusion happens to the kth object at time t, the algorithm proposes that the location of the kth object the same as its container while keeping it's container the same as in the previous frame: $x_t^k = (l_t^{c_t^k}, c_t^k), c_t^k \neq k$.

Blocked. In such cases, an object is occluded due to another object sitting in between the object and the camera from certain camera views, as shown in Figure 3 (a) and 3 (b), where an apple is occluded by a bowl and a person. The dashed box is the proposal for the apple's present location, which is the same as the location in the last frame before occlusion happened. Formally, suppose such occlusion happens to the kth object at time t, the algorithm proposes the object state as the same in previous state: $x_t^k = x_{t-1}^k = (l_{t-1}^k, c_{t-1}^k)$, where the location and containment relation remain the same.

Temporal Hypotheses by Human Actions

Across different frames, we consider human as the cause and the only cause of object state changes, assuming no other external disturbance in a scene. In other words, if there is no human action occurring, the objects should remain the same location and the containment relations will not change. As a result, human actions impose a hard constraint to rule out the implausible sudden jumps from the object detection, resulting in a smooth and plausible trajectory.

In this paper, we represent the human action as a skeleton sequence $H = \{H_1, H_2, \cdots, H_\tau\}$, where τ is the length of the sequence. At time t, 25 joints of human skeleton captured by a Kinect sensor were used: $H_t = (h_t^1, h_t^2, \cdots, h_t^{25})$.

Recovering Incomplete Trajectories

We recover incomplete trajectories using MAP by reasoning about containment relations and human actions:

$$T^* = \arg\max_T P(T|I) = \arg\max_T P(T|\mathcal{X}, H) \quad (1)$$

$$\propto \arg\max_T P(\mathcal{X}|T)P(T|H) \quad (2)$$

$$= \arg\max_T \prod_k P(\mathcal{X}|T^k)P(T^k|H), \quad (3)$$

where $\mathcal{X}$ and H are the object detection in space and human action in time, respectively. $P(\mathcal{X}|T^k) = \prod_{t=1}^{\tau} P(\mathcal{X}_t|x_t^k)$ models the likelihood for object detector response $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_\tau\}$ $P(T^k|H)$ is a dynamic model which is a smoothness term for trajectory, and can be decomposed as

$$P(T^k|H) = P(\{x_1^k, x_2^k, \cdots, x_\tau^k\}|H) \quad (4)$$

$$= P_S(x_0^k) \prod_{t=1}^{\tau} P(x_t^k|x_{t-1}^k, H_{t-1})\, P_E(x_\tau^k), \quad (5)$$

where $P_S(x_0^k)$ and $P_E(x_\tau^k)$ are the probability for initialization and termination, respectively, and $P(x_t^k|x_{t-1}^k, H_{t-1}^k)$ is the transition probability of two consecutive frames, which models the probability that the object status changes from time $t-1$ to t based on the observation of human action H_{t-1}. Intuitively, this probability evaluates the consistency between the location of an object and human actions. As we discuss in previous section, the location changes of an object can be interpreted by the occurrence of human actions. Figure 4 illustrates three examples of the object location changes and the corresponding human actions, including (a) a person taking an apple from a bowl, (b) a person throwing a frisbee, and (c) the object keeps the same location without human action.

The transition probability of two consecutive states is

$$-\log P(x_t^k|x_{t-1}^k, H_{t-1}^k) = \langle \omega\,, \theta_{[\epsilon]} \rangle, \quad (6)$$

where ω is the template parameter. $\theta_{[\epsilon]}$ is the extracted human action feature in a time interval $[t-1-\epsilon, \quad t-1+\epsilon]$.

For θ, we consider three types of features in a sliding window on the time axis: human pose, relative movements between the human and the object, and the object movements. Suppose that the sliding window size is 2ϵ, the feature vector sequence at time t is $\mathcal{F}_m = (\mathcal{F}_m^h, \mathcal{F}_m^r, \mathcal{F}_m^o)$, $m \in [t-1-\epsilon \quad t-1+\epsilon]$. Specifically,

- $\mathcal{F}_m^h$ is the relative distance of all the skeletons to three base points (two shoulders and one spine point), which encodes human action. In Figure 4, we show one component of $\mathcal{F}_m^h$ in red lines: the distance between the hand and the spine point.
- $\mathcal{F}_m^r$ is the distance between human hand and the location of the object, which is denoted in yellow lines in Figure 4.
- $\mathcal{F}_m^o$ is the distance between the locations of the object at time m and t, depicted by green lines in Figure 4.

A sequence clip is first interpolated to a certain length. The wavelet transform is then applied to $\mathcal{F}_m$. The coefficients at the low frequency are kept as the action feature. The window sizes and sliding steps are both in multiple scales.

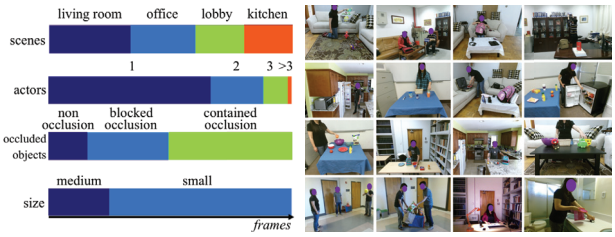
Substituting Eq. 4 into Eq. 1, we then have

$$T^* = \arg\max_T \prod_{k=1}^{K} \prod_{t=1}^{\tau} [P(\mathcal{X}_t|x_t^k) \cdot \quad (7)$$

$$P_S(x_0^k) \cdot P(x_t^k|x_{t-1}^k, H_{t-1}) \cdot P_E(x_\tau^k)].$$

We can reformulate Eq. 7 as an Integer Linear Programming problem:

$$f^* = \arg\min_f C(f), \quad (8)$$



(a) (b)

Figure 5: Our occluded object tracking dataset. (a) Statistic of the dataset. (b) Some examples of the activities.

where

$$C(f) = \sum_i c_i^s f_i^s + \sum_{i,j} c_{ij} f_{ij} + \sum_i c_i f_i + \sum_i c_i^e f_i^e \quad (9)$$

$$c_{ij} = -\log P(x_j|x_i, H_i) \quad (10)$$

$$c_i = -\log P(x_i|T^k) \quad (11)$$

$$c_i^e = -\log P_E(x_i) \quad (12)$$

$$c_i^s = -\log P_S(x_i) \quad (13)$$

$$\text{s.t.} \quad f_{ij}, f_i, f_i^s, f_i^e \in \{0, 1\}. \quad (14)$$

This is equivalent to finding a min-cost path in network flow with source S and sink E as shown in Figure 2: the red arrows denotes the detection on input RGB-D images with cost on the edge c_i, the dashed red arrows indicates that the object is hidden at the present state and there is no observation from current frame, and each transition between successive frames is denoted by blue lines with cost c_{ij} given by human actions, serving as a smoothness term.

Dynamic programming is applied to optimize Eq. 9. By assuming objects will not affect each other's trajectory, we optimize the trajectory for each object individually. Firstly, we run K-Shortest Paths Algorithm (Berclaz et al. 2011), which generates a set of tracklets. Then we use the Viterbi algorithm to connect these tracklets, which yields continuous trajectories for each object.

Experiments

Dataset

We collected a 3D dataset with diverse scenes, multiple actions and various objects to evaluate the proposed method (Figure 5). 1346 video clips in 10 scene categories were captured by Kinect sensors. RGB and depth images, 3D human skeletons as well as point cloud data were recorded in each video clip. Compared with existing dataset, the proposed dataset focuses on occluded objects for visual tracking, which consists of a large variety of human actions causing object location changes in different scenarios, such as throwing, catching, picking up, putting down, fetching, lifting, *etc*.

Each frame in the dataset was manually annotated with ground truth by drawing bounding boxes for each object. When an object is occluded, we annotate the ground truth based on two types of causes for the occlusions. i) Contained. The object shares the location with its container,

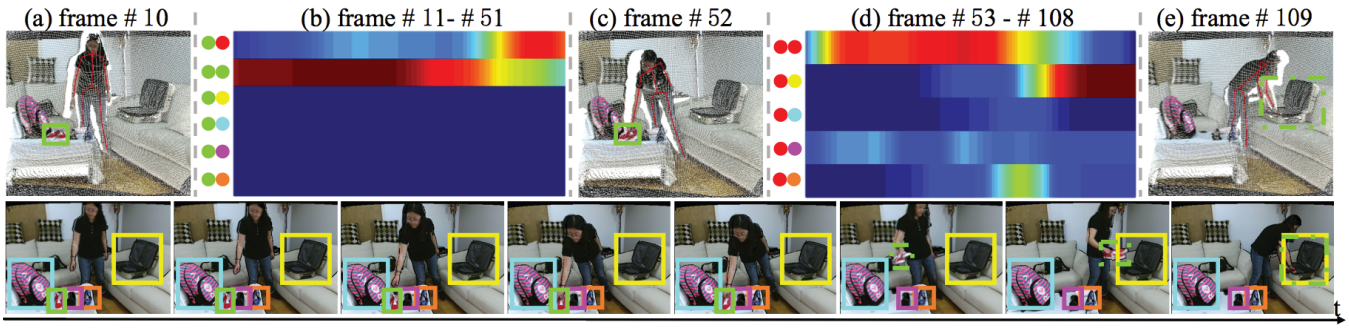


Figure 6: Transition probability of the object location in the green bounding box. The solid boxes depict that the object is tracked by object detectors. The dashed boxes depict that the object is recovered by inference. (a), (c) and and (e) show detected bounding boxes and human skeletons on point cloud. (b) and (d) are the transition probabilities between two possible locations. In (b), the bottom four bars with low probability keep the same since we constrain the impossible object moving that are not caused by human actions.

Figure 7: Confusion matrix of HOI. ① denotes that the object movement is consistent with HOI, whereas② denotes that the object movement is not consistent with HOI. (a) Human pose sequence only. (b) Human pose sequence with objects context. (c) Joint inference in our method.

forming a new containment relation. ii) Blocked. The object is stationary, and the containment relation remains the same. For the situation that a person serves as a container, we draw a bounding box on the person's hand.

Transitions in DP: an In-depth Example

Figure 6 shows an example of the trajectory inference process of an object bounded by a green box. The tracking results are visualized in the bottom panel, where the solid boxes denote the detected location, and the dashed boxes denote the inferred results. Specifically, we employed the state-of-the-art RGB-D based detectors (Song and Xiao 2013) on a RGB-D image sequence. The detected objects are bounded by boxes with different colors shown in Figure 6 (a), (c) and (e). The human skeletons from Kinect are in red color.

Figure 6 (b) and (d) illustrate the partial transition probabilities changes between two consecutive states in an interval (frame 11 to frame 51, frame 53 to frame 108), equivalent to the probability of human actions and calculated by Eq. 6. The left panel of (b) and (d) are some possible transitions. Take the first bar in (b) as an example. The green and red dot represent the location of the object bounded by green bounding box and the person, respectively. The bar depicts the probabilities of the transition from the green box location to the human hand location over time. We can see that the probability increases from frame 11 to frame 51. At frame 51, the person picked up the object. From frame 59 to frame 108, the object was held by the person. The first bar of Figure 6 (d) shows the probability of the object being carried by this person.

It is worth noting that the bottom four bars in Figure 6 (b) have low transition probabilities which are close to zero. Take the last bar in Figure 6 (b) as an example. It shows the probability of the object bounded by the green box moving to the location of the object bounded by the orange box. This movement was not caused by human action and violated our assumption, which was ruled out during the inference.

From frame 51 to 109, the object was contained and thus cannot be visually detected. Human action provided a strong cue for the object location: a person picked up this object and moved it to a container bounded by a yellow bounding box.

Ablative Analysis: Roles of Interactions in HOI

In this section, we evaluate the roles and importance of HOI quantitatively by turning on and off certain components in the proposed method.

We consider the HOI as a binary classification problem: if the object movement is consistent with human action, it should be classified as positive; otherwise it is negative. We define whether the object movement is consistent with human action using two criteria: i) if no human action, the object should remain stationary, and vice versa; ii) if there is an object location change, the object should follow the trajectory of human action.

We first consider the simplest method using human pose only, *i.e.*, Eq. 6 with feature vector $\mathcal{F}_m = (\mathcal{F}_m^h)$. As showed in Figure 7a, using human pose only is not sufficient to achieve reasonable performance. This was mainly caused by the lack of object context, disallowing a good classification between certain actions, *e.g.*, putting down and picking up.

Next, we consider the method using both human pose and object context, *i.e.*, Eq. 6 with feature vector $\mathcal{F}_m = (\mathcal{F}_m^h, \mathcal{F}_m^r, \mathcal{F}_m^o)$. Although achieving reasonable results as shown in Figure 7b, this method only looks at local window $m \in [t-1-\epsilon, t-1+\epsilon]$, thus lacking of global optimization.

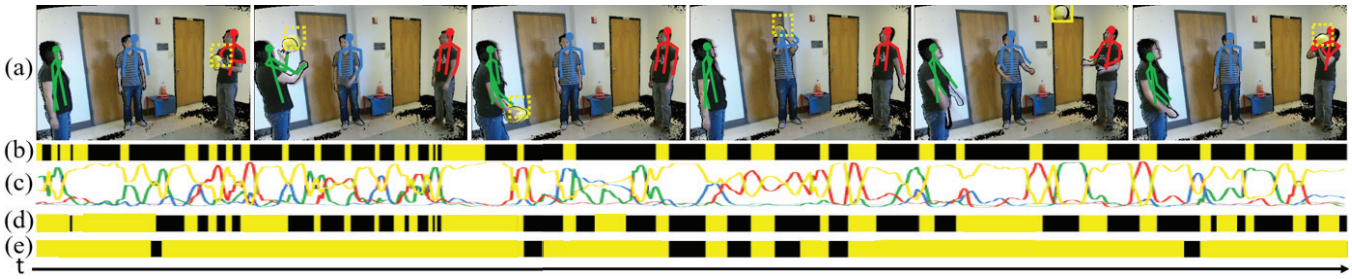


Figure 8: An example of the experiment results. The goal is to track the yellow ball. In each bar, the yellow represents the correct results, and the black represents the wrong results. The overlap ratio of bounding boxes were set to 0.5. Different colors denote different objects: actor 1 (green), actor 2 (blue), actor 3 (red) and ball (yellow). (a) Examples of tracking results. The dashed boxes depict the object is occluded. (b) Temporal-suppression results. (c) The scores of consistency between object movement and human action. (d) Spatial-suppression results. (e) Full model results.

With back propagation using DP as described in Eq. 7 and Eq. 9, the proposed method globally adjust the inference, resulting in the best performance among three methods as shown in Figure 7c.

All the results report here were trained by SVM on the same training data. To address the problem of different scales of interaction, different step sizes and different sliding window sizes along time axis were used.

Ablative Analysis: Spatial/Temporal Suppression

In this section, we design two experiments (baseline1 and baseline2) to evaluate how spatial and temporal information influence the tracking. We compare the results of these two experiments with the approach of tracking with occlusion model (baseline3) and the proposed method (full model).

As an example, we show comparisons of results from different methods using a video of 530 frames (Figure 8). In this video, three actors threw and caught a ball highlighted by a yellow bounding box. The ball traveled fairly fast, appearing and then disappearing frequently. Directions, scales, and views of the ball also varied. Severe occlusions by hands or other body parts occurred.

Temporal Suppression (baseline1). In this setting, we do not consider the human actions, *i.e.*, set Eq. 10 to a constant. As a result, it is equivalent to an online tracking problem: the trajectory of an object is determined only by the response of detectors. Non-maximum suppression was applied on all detection candidates per frame. Figure 8 (b) shows the results.

Spatial Suppression (baseline2). In this setting, we set Eq. 11 to a constant, *i.e.*, not considering the detection score, but inferring object location only by human actions in time. In other words, the trajectory of an object is determined only by the transition probabilities modeled by human actions.

Results were shown in Figure 8 (d). Failure cases mostly fall into two categories: i) when human skeleton, the object and the container are occluded at the same time, and ii) when human skeleton or the object are partially occluded, it is difficult to distinguish the throwing action from the catching action as the lack of action cues or spatial context.

Tracking with occlusion model (baseline3). A related topic in computer vision is multi-object tracking. Some recent efforts were trying to infer and recover both short-term and long-term occluded objects by occlusion assumption (Zhang, Li, and Nevatia 2008; Andriyenko and Schindler 2011). In this paper, we use (Zhang, Li, and Nevatia 2008) as the baseline representing the state-of-the-art multi-object tracking algorithm with occlusion assumptions, which adopted an Explicit Occlusion Model (EOM) to track with long-term inter-object occlusions, adding occluded object hypothesis to model occlusions.

Full model. The results of full model are shown in Figure 8 (e). Benefit from both spatial and temporal terms with back propagation, most of the occlusions were successfully recovered. The failure cases happened when the object was transferred continuously between containers without any valid object detection in space. For example, from frame 265-320, the ball was passed from actor 1 to actor 2 and then passed to actor 3. Later, at frame 320, the ball was passed back to actor 1. In this case, the ball was not detected during the entire process. As the result, our method believed the ball was in the hand of actor 1 all the time.

Table 1: Tracking accuracy of full model compared with three baselines on different subsets of the proposed dataset.

	baseline1	baseline2	baseline3	full
all	0.57	0.32	0.59	0.69
blocked	0.21	0.08	0.25	0.47
contained	0.15	0.02	0.16	0.42

Results. To evaluate our method quantitatively, we extract two subsets of the video clips from the proposed dataset based on two causes of occlusions: contained by another object and the blocked camera views. We evaluate the accuracy on these two subsets as well as on the entire dataset.

Success rate was adopted for quantitative analysis, defined as the ratio between the number of frames with correct object localization and the number of all frames. Given an estimated bounding box of an object b_e and the ground truth bounding box b_g, the overlap score is defined as $r = \frac{b_e \cap b_g}{b_e \cup b_g}$, where $\cap$ and $\cup$ are the intersection and union of two regions. An object bounding box is considered correct if $r \geq r_0$. The

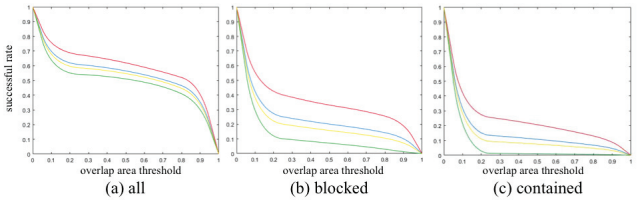


Figure 9: Different overlap ratios evaluated on different subsets. The red, yellow, green, and blue line represent the results of full model, baseline1, baseline2, and baseline3, respectively. The horizontal axis is the threshold axis, ranging from 0 to 1. The vertical axis is the success rate.

accuracy of the tracking results are shown in Table 1 with $r_0 = 0.5$. We further evaluate success rate when varying different overlap ratios r_0. Results are shown in in Figure 9.

Evaluations on Existing Datasets

In addition to our proposed dataset designed for tracking severe objects which are "contained" or "blocked", we further test our method on some existing datasets for modeling HOI: CAD-120 (Sung et al. 2012), CMU interaction dataset (Gupta, Kembhavi, and Davis 2009), MSR action recognition dataset (Yuan, Liu, and Wu 2009), and NW-UCLA Multiview Action 3D dataset (Wang et al. 2014). The major differences between these four datasets and other public available datasets (*e.g.*, the multiple objects tracking datasets) is: these four datasets focus on rich HOI, severe occlusions between human and objects, and large appearance variations of object, which is the main focus of this paper.

To evaluate our method on these datasets, we apply the RGB-D detectors (Song and Xiao 2013) for RGB-D datasets (Sung et al. 2012; Wang et al. 2014; Yuan, Liu, and Wu 2009), and RGB detectors (Kalal, Mikolajczyk, and Matas 2012) for RGB-only dataset (Gupta, Kembhavi, and Davis 2009). For action detection, we train a classifier on 2D data for CAD 120 and CMU interaction datasets which have no skeleton data. Examples of qualitative results are shown in Figure 10.

The quantitative tracking accuracy is shown in Table 2. The performance of our method on MSR action recognition and Northwestern-UCLA dataset is better than the results on CAD-120 and CMU dataset. We believe two reasons contributed to the performance differences: i) Some errors were caused by the unreliable action detections in 2D space compared to 3D space. ii) Small object detections are more challenging in 2D cases, such as pouring from a cup, lighting a flash light in the CMU dataset.

Table 2: Tracking accuracy on other datasets.

	baseline1	baeline2	baseline3	full
CAD-120	0.30	0.13	0.33	0.47
CMU	0.28	0.12	0.25	0.43
MSR	0.43	0.21	0.44	0.60
NW-UCLA	0.56	0.25	0.56	0.72

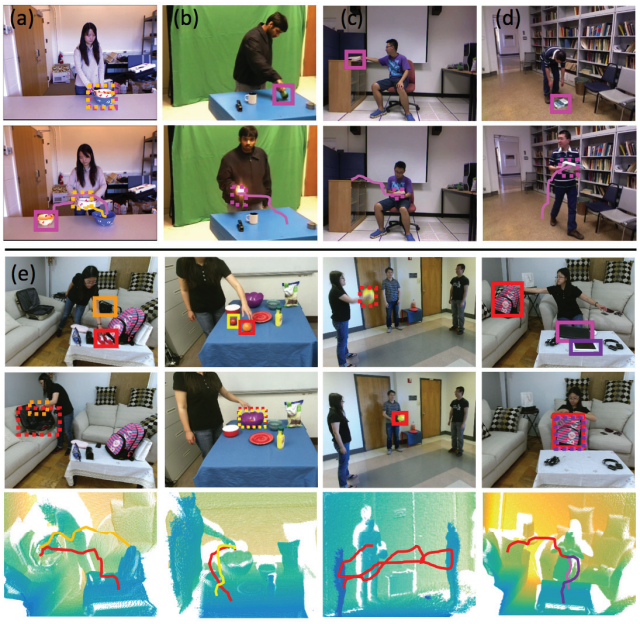


Figure 10: More qualitative results. Solid boxes are detected by tracking algorithm and the dashed boxes are inferred. Top two rows: (a) CAD-120, (b) CMU Dataset, (c) MSR Dataset, and (d) NW-UCLA Dataset. The bottom three rows (e) are the results on our proposed occluded objects dataset.

Conclusions and Discussions

We propose an algorithm to infer occluded objects and recover the incomplete trajectories for objects in a cluttered indoor scene by reasoning about containment relations and human actions. We assume that the movements of objects are only caused by human actions, and explicitly model occlusions from two causes: contained by others, or blocked camera views. A network flow representation is adopted to globally optimize trajectories based on two occlusion causes. In the experiment, we test our method on the collected occluded objects dataset and other four existing datasets, demonstrating the proposed method can provide better performance in challenging scenarios.

The current work is limited in the following aspects:

i) When the object detection is noisy, the performance of our method is likely to degenerate, especially when continuous transitions between occluded objects happen. High level knowledge may help to improve the results, *e.g.*, integrating an inference algorithm for the intention of the agent.

ii) We currently limit the scenarios where human is the only cause that leads to the object status changes, thus are unable to handle situations where objects move only by invisible force field, *e.g.*, gravity. Such challenging situations would require a much deeper understanding of the 3D scenes, particular the "dark matter" that is invisible (Shu et al. 2015), *e.g.*, functionality (Zheng et al. 2013; Zhu, Zhao, and Zhu 2015) and causality (Fire and Zhu 2013).

iii) The majority of computer vision community is focusing on rigid body. However, properly modeling fluid (*e.g.*,

water (Bates et al. 2015; Kubricht et al. 2016)) and granular material (*e.g.,* sand (Kubricht et al. 2017)) is important for inferring containment relations.

Acknowledgment

The work reported herein was supported by a Natural Science Foundation of China (NSFC) grant No.61472038 and No.61375044 (to Liang), DARPA XAI grant N66001-17-2-4029 and ONR MURI grant N00014-16-1-2007 (to Zhu).

References

Aksoy, E. E.; Abramov, A.; Dörr, J.; Ning, K.; Dellen, B.; and Wörgötter, F. 2011. Learning the semantics of object–action relations by observation. *The International Journal of Robotics Research* 30(10):1229–1249.

Andriyenko, A., and Schindler, K. 2011. Multi-target tracking by continuous energy minimization. In *CVPR*, 1265–1272.

Bates, C.; Battaglia, P.; Yildirim, I.; and Tenenbaum, J. B. 2015. Humans predict liquid dynamics using probabilistic simulation. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society*.

Berclaz, J.; Fleuret, F.; Turetken, E.; and Fua, P. 2011. Multiple object tracking using k-shortest paths optimization. *T-PAMI* 33(9):1806–1819.

Bredeweg, B., and Forbus, K. D. 2003. Qualitative modeling in education. *AI Magazine* 24(4):35.

Davis, E.; Marcus, G.; and Chen, A. 2013. Reasoning from radically incomplete information: The case of containers. In *Proceedings of the Second Annual Conference on Advances in Cognitive Systems (ACS)*, volume 273, 288.

Eichner, M., and Ferrari, V. 2010. We are family: Joint pose estimation of multiple persons. In *ECCV*, 228–242. Springer.

Enzweiler, M.; Eigenstetter, A.; Schiele, B.; and Gavrila, D. M. 2010. Multi-cue pedestrian classification with partial occlusion handling. In *CVPR*, 990–997.

Fire, A. S., and Zhu, S.-C. 2013. Learning perceptual causality from video. In *AAAI Workshop: Learning Rich Representations from Low-Level Sensors*.

Frank, A. U. 1996. Qualitative spatial reasoning: Cardinal directions as an example. *International Journal of Geographical Information Science* 10(3):269–290.

Gupta, A.; Kembhavi, A.; and Davis, L. S. 2009. Observing human-object interactions: Using spatial and functional compatibility for recognition. *T-PAMI* 31(10):1775–1789.

Kalal, Z.; Mikolajczyk, K.; and Matas, J. 2012. Tracking-learning-detection. *T-PAMI* 34(7):1409–1422.

Kubricht, J.; Jiang, C.; Zhu, Y.; Zhu, S.; Terzopoulos, D.; and Lu, H. 2016. Probabilistic simulation predicts human performance on viscous fluid-pouring task. In *Proceedings of the 38th Annual Meeting of the Cognitive Science Society*, 1805–1810.

Kubricht, J.; Zhu, Y.; Jiang, C.; Terzopoulos, D.; Zhu, S.-C.; and Lu, H. 2017. Consistent probabilistic simulation underlying human judgment in substance dynamics. In *Proceedings of the 39th Annual Meeting of the Cognitive Science Society*, 700–705.

Liang, W.; Zhao, Y.; Zhu, Y.; and Zhu, S.-C. 2015. Evaluating human cognition of containing relations with physical simulation. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society*, 782–787.

Liang, W.; Zhao, Y.; Zhu, Y.; and Zhu, S.-C. 2016. What is where: Inferring containment relations from videos. In *IJCAI*, 3418–3424.

Mottaghi, R.; Schenck, C.; Fox, D.; and Farhadi, A. 2017. See the glass half full: Reasoning about liquid containers, their volume and content. In *ICCV*.

Rohrbach, M.; Amin, S.; Andriluka, M.; and Schiele, B. 2012. A database for fine grained activity detection of cooking activities. In *CVPR*, 1194–1201.

Shu, T.; Xie, D.; Rothrock, B.; Todorovic, S.; and Zhu, S.-C. 2015. Joint inference of groups, events and human roles in aerial videos. In *CVPR*, 4576–4584.

Shukla, N.; He, Y.; Chen, F.; and Zhu, S.-C. 2017. Learning human utility from video demonstrations for deductive planning in robotics. In *Conference on Robot Learning*, 448–457.

Smeulders, A. W.; Chu, D. M.; Cucchiara, R.; Calderara, S.; Dehghan, A.; and Shah, M. 2014. Visual tracking: An experimental survey. *T-PAMI* 36(7):1442–1468.

Song, S., and Xiao, J. 2013. Tracking revisited using rgbd camera: Unified benchmark and baselines. In *ICCV*, 233–240.

Sung, J.; Ponce, C.; Selman, B.; and Saxena, A. 2012. Unstructured human activity detection from rgbd images. In *ICRA*, 842–849.

Wang, J.; Nie, X.; Xia, Y.; Wu, Y.; and Zhu, S.-C. 2014. Cross-view action modeling, learning and recognition. In *CVPR*, 2649–2656.

Wang, J.; Zhang, Z.; Xie, C.; Zhou, Y.; Premachandran, V.; Zhu, J.; Xie, L.; and Yuille, A. 2017. Visual concepts and compositional voting. *arXiv preprint arXiv:1711.04451*.

Wang, H.; Liang, W.; and Yu, L.-F. 2017. Transferring objects: Joint inference of container and human pose. In *ICCV*, 2933–2941.

Wang, J.; Liu, Z.; and Wu, Y. 2014. Learning actionlet ensemble for 3d human action recognition. In *Human Action Recognition with Depth Cameras*, 11–40. Springer.

Wei, P.; Zhao, Y.; Zheng, N.; and Zhu, S.-C. 2013. Modeling 4d human-object interactions for event and object recognition. In *ICCV*, 3272–3279.

Wojek, C.; Walk, S.; Roth, S.; and Schiele, B. 2011. Monocular 3d scene understanding with explicit occlusion reasoning. In *CVPR*, 1993–2000.

Wu, Y.; Lim, J.; and Yang, M.-H. 2015. Object tracking benchmark. *T-PAMI* 37(9):1834–1848.

Yang, M.; Wu, Y.; and Hua, G. 2009. Context-aware visual tracking. *T-PAMI* 31(7):1195–1209.

Yao, B. Z.; Nie, B. X.; Liu, Z.; and Zhu, S.-C. 2014. Animated pose templates for modeling and detecting human actions. *T-PAMI* 36(3):436–452.

Yu, L.-F.; Duncan, N.; and Yeung, S.-K. 2015. Fill and transfer: A simple physics-based approach for containability reasoning. In *CVPR*, 711–719.

Yuan, J.; Liu, Z.; and Wu, Y. 2009. Discriminative subvolume search for efficient action detection. In *CVPR*, 2442–2449.

Zhang, L.; Li, Y.; and Nevatia, R. 2008. Global data association for multi-object tracking using network flows. In *CVPR*, 1–8.

Zheng, B.; Zhao, Y.; Yu, J. C.; Ikeuchi, K.; and Zhu, S.-C. 2013. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *CVPR*, 3127–3134.

Zhu, Y.; Jiang, C.; Zhao, Y.; Terzopoulos, D.; and Zhu, S.-C. 2016. Inferring forces and learning human utilities from videos. In *CVPR*, 3823–3833.

Zhu, Y.; Zhao, Y.; and Zhu, S.-C. 2015. Understanding tools: Task-oriented object modeling, learning and recognition. In *CVPR*, 2855–2864.