Watch-n-Patch: Unsupervised Learning of Actions and Relations
Chenxia Wu, Jiemi Zhang, Ozan Sener, Bart Selman, Silvio Savarese, and Ashutosh Saxena

Abstract—There is a large variation in the activities that humans perform in their everyday lives. We consider modeling these composite human activities which comprises multiple basic level actions in a completely unsupervised setting. Our model learns high-level co-occurrence and temporal relations between the actions. We consider the video as a sequence of short-term action clips, which contains human-words and object-words. An activity is about a set of action-topics and object-topics indicating which actions are present and which objects are interacting with. We then propose a new probabilistic model relating the words and the topics. It allows us to model long-range action relations that commonly exist in the composite activities, which is challenging in previous works. We apply our model to the unsupervised action segmentation and clustering, and to a novel application that detects forgotten actions, which we call action patching. For evaluation, we contribute a new challenging RGB-D activity video dataset recorded by the new Kinect v2, which contains several human daily activities as compositions of multiple actions interacting with different objects. Moreover, we develop a robotic system that watches people and reminds people by applying our action patching algorithm. Our robotic setup can be easily deployed on any assistive robot.

Index Terms—Unsupervised Learning, Activity Discovery, Robot Application.

1 INTRODUCTION

The average adult forgets three key facts, chores or events every day [2]. Hence it is important for a vision system to be able to detect not only what a human is currently doing but also what he forgot to do. For example in Fig. 1, someone fetches milk from the fridge, pours the milk to the cup, takes the cup and leaves without putting back the milk, then the milk would go bad. In this paper, we focus on modeling these composite human activities then detecting the forgotten actions for a robot, which learns from a completely unlabeled set of RGB-D videos.

A human activity is composite, i.e., it is composed of several basic level actions. For example, a composite activity warming milk contains a sequence of actions: fetch-milk-from-fridge, microwave-milk, put-milk-back-to-fridge, fetch-milk-from-microwave, and leave. Modeling this poses several challenges. First, some actions often co-occur in a composite activity but some may not. Second, co-occurring actions have variations in temporal orderings, e.g., people can first put-milk-back-to-fridge then microwave-milk instead of the inverse order in the above example, as its ordering is more relevant to the action fetch-milk-from-fridge. Moreover, these ordering relations could exist in both short-range and long-range, e.g., pouring is followed by drink while sometimes fetch-book is related to put-back-book with a long read between them. Third, the objects the human interacting with are also important to modeling the actions and their relations, as same actions often have common objects in interaction.

The challenge that we undertake in this paper is: Can an algorithm learn about the aforementioned relations in the composite activities when just given a completely unlabeled set of RGB-D

Fig. 1: Our Watch-Bot understands what human is currently doing by automatically segmenting the composite activity into basic level actions. We propose a completely unsupervised approach to modeling the human skeleton and object features to the actions, as well as the pairwise action co-occurrence and temporal relations. Using the learned model, our robot detects humans’ forgotten actions and reminds them by pointing out the related object using the laser spot.
Most previous works focus on action detection in a supervised learning setting. In the training, they are given fully labeled actions in videos [31], [42], [43], or weakly supervised action labels [9], [13], or locations of human/their interacting objects [29], [37], [50]. Among them, the temporal structure of actions is often discovered by Markov models such as Hidden Markov Model (HMM) [49] and semi-Markov [16], [45], or by linear dynamical systems [8], or by hierarchical grammars [4], [25], [59], [52], [54], or by other spatio-temporal representations [22], [24], [27], [38].

Object-in-use contextual information has also been commonly used for recognizing actions [26], [27], [37], [54]. Besides relying on the manually labeling, most of these works are based on RGB features and only model the short-range relations between actions (see Section 2 for details).

Unlike these approaches, we consider a completely unsupervised setting. The novelty of our approach is the ability to model the long-range action relations in the temporal sequence, by considering pairwise action co-occurrence and temporal relations, e.g., put-milk-back-to-fridge often co-occurs with and temporally after (but not necessarily follows) fetch-milk-from-fridge. We also use the more informative human skeleton features and RGB-D object features, which have shown higher performance over RGB only features for action recognition [26], [30], [59].

In order to capture the rich structure in the composite activity, we draw strong parallels with the work done on document modeling from natural language (e.g., [8]) and proposed a Casual Topic Model (CaTM). We consider an activity video as a document, which consists of a sequence of short-term action clips containing human-skeleton-trajectories as human-words and interacting-object-trajectories as object-words. An activity is about a set of action-topics indicating which actions are present in the video, such as fetch-milk-from-fridge in the warming milk activity, and a set of object-topics indicating which object types are interacting. We draw human-words from the action-topics, and object-words from both action-topics and object-topics. Then we model the following (see Fig. 2):

- **Action co-occurrence.** Some actions often co-occur in the same activity and may have the same objects. We model the co-occurrence by adding correlated topic priors to the occurrence of action-topics and object-topics, e.g., action-topics fetch-book and put-back-book have strong correlations and are also strongly correlated to object-topic book.

- **Action temporal relations.** Some actions often causally follow each other, and actions change over time during the activity execution. We model the relative time distributions between every action-topic pair to capture the temporal relations.

We first show that our model is able to learn meaningful representations from the unlabeled composite activity videos. We use the model to temporally segment videos to action segments by assigning action-topics. We show that these action-topics are promising to be semantically meaningful by mapping them to ground-truth action classes and evaluating the labeling performance.

We then show that our model can be used to detect forgotten actions in the composite activity, a new application that we call action patching. We enable a robot, which we call Watch-Bot, to detect humans’ forgotten actions as well as to localize the related object in the scene. The setup of the robot can be easily deployed on any assistive robot and applied to different areas such as industry, medical work and home use. We evaluate the action patching accuracy to show that the learned co-occurrence and temporal relations are very helpful to inferring the forgotten actions. We also show that our Watch-Bot is able to remind humans of forgotten actions in the real-world robotic experiments.

We also provide a new challenging RGB-D activity video dataset recorded by the new Kinect v2 (see examples in Fig. 12), in which the human skeletons are also recorded. It contains 458 videos of human daily activities as compositions of multiple actions interacting with different objects, in which people forget actions in 222 videos. They are performed by different subjects in different environments with complex backgrounds. In robotic experiments, we show that our Watch-Bot is able to remind humans of forgotten actions in the real-world experiments.

In summary, the main contributions of this work are:

- Our model is completely unsupervised thus being more useful and scalable.
- Our model considers both the short-range and the long-range action relations, showing the effectiveness in the action segmentation and clustering.
- We show a new application by enabling a robot to remind humans of forgotten actions in the real scenes.
- We provide a new challenging RGB-D activity dataset recorded by the new Kinect v2, which contains videos of multiple actions interacting with different objects.

The paper is organized as follows. Section 2 introduces the related works. Section 3 outlines our approach to modeling the composite activity. We present the visual features of the activity video clip in Section 4. Section 5 gives the detailed description of our learning model as well as its learning and inference. Section 6 introduces our watch-bot system to reminding of forgotten actions using our learned model. We give an extensive evaluation and discussion in the experiments in Section 7. Sections 8 concludes the paper.

## 2 Related Work

### Action Recognition

Our work is related to the works on action recognition in computer vision. There is a large number of works on action recognition, which can be referred in recent surveys [3]. In this section, we cover the most related approaches. Most previous works on action recognition are supervised [9], [13], [29], [31], [34], [38], [42], [50]. Among them, the most popular are linear-chain models such as hidden markov model (HMM) [49], semi-Markov [16], [45] and the linear dynamic system [5]. They focus on modeling the local transitions (between frames, temporal segment, or sub-actions) in the activities. More complex hierarchical relations [28], [39], [52], [54] or graph relations [4], [46] are considered in modeling actions in the complex activity. There are also some works focusing on detecting local action patches, primitives, trajectories or spatio-temporal features [19], [33], [35], [62] without considering the high-level action relations. There also exist some unsupervised approaches on action recognition. Yang et al. [62] develop a meaningful representation by discovering local motion primitives in an unsupervised way, then a HMM is learned over these primitives. Jones et al. [20] propose an

---

1. Here we consider the same object type like book can be variant in appearance in different actions such as close book in the fetch-book action and open book in the reading action.

2. The dataset and tools are released at http://watchnpatch.cs.cornell.edu.
unsupervised dual assignment clustering on the dataset recorded from two views.

Although these approaches have performed well in different areas, most of them rely on local relations between adjacent clips or actions that ignore the long-term action relations and use RGB visual features. Unlike these approaches, we use the richer human skeleton and RGB-D features rather than the RGB action features [21], [55]. We model the pairwise action co-occurrence and temporal relations in the whole video, thus relations are considered globally and completely with the uncertainty. We also use the learned relations to infer the forgotten actions without any manual annotations.

**RGB-D and Human Skeleton Features.** Action recognition using human skeletons and RGB-D camera have shown the advantages over RGB videos in many works. Skeleton-based approach focus on proposing good skeletal representations [30], [43], [48], [51], [59]. Besides of the human skeletons, we also detect the object-in-use contextual information [21], [53], [54]. Moreover, Hu et al. [17] propose a joint learning model to simultaneously learn heterogenous features from RGB-D activity videos. Most of them focus on designing or learning good action features. They lost the high-level action relations which can be captured in our model.

**Bayesian Models.** Our work is also related to the Bayesian models. LDA [8] was the first hierarchical Bayesian topic model and widely used in different applications. The correlated topic models [6], [23] add the priors over topics to capture topic correlations. A topic model over absolute timestamps of words is proposed in [55] and has been applied to action recognition [14]. However, the independence assumption of different topics would lead to non smooth temporal segmentations. Recently, a multi-feature max-margin hierarchical Bayesian model [60] is proposed to jointly learn a high-level representation by combining a hierarchical generative model and discriminative maxmargin classifiers in a unified Bayesian framework. Differently, our model considers both correlations and the relative time distributions between topics rather than the absolute time, which captures richer information of action structures in the complex human activity.

**Perception of Human Activities for Robotics.** Our work is also related to the works on recognizing human actions for robotics [10], [25], [32]. Yang et al. [61] presented a system that learns manipulation action plans for robot from unconstrained youtube videos. Hu et al. [18] proposed an activity recognition system trained from soft labeled data for the assistant robot. Chrungoo et al. [11] introduced a human-like stylized gestures for better human-robot interaction. Piyathilaka et al. [40] used 3D skeleton features and trained dynamic bayesian networks for domestic service robots. The output filter spot on object is also related to the work ‘a clickable world’ [36], which selects the appropriate behavior to execute for an assistive object-fetching robot using the 3D location of the click by the laser pointer. However, it is challenging to directly use these approaches to detecting the forgotten actions and remind people.

### 3 Overview

We outline our approach in this section (see Fig. 2). The input to our system is RGB-D videos with the 3D joints of human skeletons from Kinect v2. We first decompose a video into a sequence of overlapping fixed-length temporal clips (step (1)). We then extract the human-skeleton-trajectory features and the interacting-object-trajectory features from the clips (introduced in Section 4). The human skeleton features and RGB-D object features have shown higher performance over RGB only features for the human action modeling [26], [30], [59].

![Fig. 2: Video representation. (1) A video frames (f_j) is first decomposed into a sequence of overlapping fixed-length temporal clips. (2) The human-skeleton-trajectories/interactive-object-trajectories are extracted from each clip, and we cluster them to form the human-dictionary/object-dictionary. (3) Then the video is represented as a sequence of human-word and object-word indices by mapping its human-skeleton-trajectories/interactive-object-trajectories to the nearest human-words/object-words in the dictionary. (4) An activity video is about a set of action-topics/object-topics indicating which actions are present and which types of objects are interacting with. (5) We learn the mapping of action-words/object-words to the action-topics/object-topics, as well as the co-occurrence and the temporal relations between the topics. (6) We assign the topics to clips using the learned model.](image)
topics, as well as the co-occurrence and the temporal relations between the topics \((5)\) in Fig. 2. Using the learned model, we can assign the action-topic/object-topic to each clip. So the continuous clips with the same assigned action-topic form an action segment \((6)\) in Fig. 2.

The unsupervised action assignments of the clips are challenging because there is no annotation during the training stage. Besides extracting rich visual features, we further consider the relations among actions and objects. Unlike previous works, our model captures long-range relations between actions e.g., put-milk-back-to-fridge is strongly related to fetch-milk-from-fridge even with pour and drink between them. We model all pairwise co-occurrence and temporal causal relations between topics in a video, using a new probabilistic model (introduced in Section 5).

Specifically, we use a joint distribution as the correlated topic priors. They estimate which actions and objects are most likely to co-occur in a video. And we use a relative time distributions of topics to capture the temporal causal relations between actions, which estimate the possible temporal ordering of the occurring actions in the video.

4 Visual Features

We describe how we extract the visual features of a clip in this section. We both extract human-skeleton-trajectory features and the interacting-object-trajectory features from the output by the Kinect v2 [1], which has an improved body tracker and the higher resolution of RGB-D frame than the Kinect v1. The tracked human skeleton has 25 joints in total. Let \(X_u = \{x_u(1), \ldots, x_u(25)\}\) be the 3D coordinates of 25 joints of a skeleton in the current frame \(u\). We first compute the cosine of the angles between the connected body parts in each frame: \(\alpha_{pq} = \langle p^{(v)} \cdot q^{(v)} \rangle / \langle p^{(v)} \cdot p^{(v)} \rangle\), where the vector \(p^{(v)} = x^{(i)} - x^{(j)}\) represents the body part. The transition between the joint coordinates and angles in different frames can well capture the human body movements. So we extract the motion features and off-set features [59] by computing their Euclidean distances \(\mathcal{D}(\cdot, \cdot)\) to previous frame \(f_{u-1,u-1}\) and the first frame \(f_{u,1,u-1}\) in the clip:

\[
\begin{align*}
  f_{u-1,u-1}^m &= \{\mathcal{D}(x_u^{(1)}, x_{u-1}^{(1)})\}_{i=1}^{25}, \quad f_{u,u-1}^m = \{\mathcal{D}(\alpha_{pq}^{(1)}, \alpha_{pq}^{(1)})\}_{pq}; \\
  f_{u,1,u-1}^m &= \{\mathcal{D}(x_u^{(1)}, x_{1}^{(1)})\}_{i=1}^{25}, \quad f_{u,u-1}^m = \{\mathcal{D}(\alpha_{pq}^{(1)}, \alpha_{pq}^{(1)})\}_{pq}.
\end{align*}
\]

Then we concatenate all \(f_{u-1,u-1}^m, f_{u,u-1}^m, f_{u,1,u-1}^m, f_{u,u-1}^m\) as the human features of the clip.

Fig. 3: Examples of the human skeletons (red line) and the extracted interacting objects (green mask, left: fridge, right: book).

We also extract the human interacting-object-trajectory based on the human hands, image segmentation, motion detection and tracking. To detect the interacting objects, first we segment each frame into super-pixels using a fast edge detection approach [12] on both RGB and depth images. The RGB-D edge detection provides richer candidate super-pixels rather than pixels to further extract objects. We then apply the moving foreground mask [47] to remove the unnecessary steady backgrounds and select those super-pixels within a distance to the human hands in both 3D points and 2D pixels. Finally, we collect the bounding boxes enclosing these super-pixels as the potential interested objects (see examples in Fig. 5).

We then track the bounding box in the segmented clip using SIFT matching and RANSAC to get the trajectories. We use the closest trajectory to the human hands for the clip. Finally, we extract six kernel descriptors [41] from the bounding box of each frame in the trajectory: gradient, color, local binary pattern, depth gradient, spin, surface normals, and KPCA/self-similarity, which have been proven to be useful features for RGB-D data [56]. We concatenate the object features of each frame as the interacting-object-trajectory feature of the clip.

5 Learning Model

In order to incorporate the aforementioned properties of activities, we present a new generative model (see the graphic model in Fig. 3 and the notations in Table 1). The novelty of our model is the ability to capture both short-range and long-range relations between actions in the compose activity videos in an unsupervised way. Using these relations, we can simultaneously segment the video and assign the action-topics as well as infer forgotten actions.

Consider a collection of \(D\) videos (documents in the topic model). Each video as a document \(d\) consists of \(N_d\) continuous clips \(\{c_{nd}\}_{n=1}^{N_d}\), each of which consists of a human-word \(w_{nd}^{h}\) mapped to the human-dictionary and an object-word \(w_{od}^{o}\) mapped to the object-dictionary. We assign action-topic to each clip \(c_{nd}\) from \(K\) latent action-topics, indicating which action-topic they belong to. We assign object-topic to each object-word \(w_{od}^{o}\) from \(P\) latent object-topics, indicating which object-topic is interacting within the clip. The assignments are denoted as \(z_{nd}^{(1)}\) and \(z_{nd}^{(2)}\).

We use superscripts \((1), (2)\) to denote action-topics and object-topics respectively. After assignments, continuous clips with the same action-topic compose an action segment in a video. All the segments assigned with the same action-topic from the training set compose an action cluster.

The topic model such as LDA [8] has been very common for document modeling from language. We use it to generate a video document using a mixture of topics. Enable to model human actions in the video, our model introduces co-occurrence and temporal structure of topics instead of the topic independence assumption in LDA.

Basic generative process. In a document \(d\), we choose \(z_{dn}^{(1)} \sim \text{Mult}(\pi_d^{(1)}), z_{dn}^{(2)} \sim \text{Mult}(\pi_d^{(2)})\), where \(\text{Mult}(\pi)\) is a multinomial distribution with parameter \(\pi\). The human-word \(w_{nd}^{h}\) is drawn from an action-topic specific multinomial distribution \(\phi_{z_{dn}^{(1)}}^{(1)}, w_{od}^{o} \sim \text{Mult}(\phi_{z_{dn}^{(2)}}^{(2)})\), where \(\phi_{z_{dn}^{(1)}}^{(1)} \sim \text{Dir}(\beta^{(1)})\) is the human-word distribution of action-topic \(k\), sampled from a Dirichlet prior with the hyperparameter \(\beta^{(1)}\). While the object-word \(w_{od}^{o}\) is drawn from an action-topic and object-topic specific multinomial distribution \(\phi_{z_{dn}^{(1)},z_{dn}^{(2)}}^{(12)}, w_{od}^{o} \sim \text{Mult}(\phi_{z_{dn}^{(12)}}^{(12)})\), where \(\phi_{z_{dn}^{(12)}}^{(12)} \sim \text{Dir}(\beta^{(12)})\) is the object-word distribution of action-topic \(k\) and object-topic \(p\). Here we consider the same object type like book can be variant in appearance in different actions such as a close book in fetch-book and a open book in read action. So we consider the object-word distribution for different combinations of the action topic and the object topic.
Topic correlations. The co-occurrence such as action pour and action drink, object book and action read, is useful to recognizing the co-occurring actions/objects and also gives a strong evidence for detecting forgotten actions. We model the co-occurrence by drawing their priors from a mixture distribution. Let \( \pi_{kd}^{(1)} = \pi_{pd}^{(2)} \) be the probability of action-topics \( k \) and object-topics \( p \) occurring in document \( d \), where \( \sum_{k=1}^{K} \pi_{kd}^{(1)} = 1 \), \( \sum_{p=1}^{P} \pi_{pd}^{(2)} = 1 \). Instead of sampling it from a fix Dirichlet prior with parameter \( \alpha \) that models them independently, we construct the probabilities by a stick-breaking process as follows. The stick-breaking notion from a multivariate normal distribution, which satisfies

\[
\pi_{kd}^{(1)} = \Psi(v_{kd}^{(1)}) \prod_{l=1}^{K-1} \Psi(v_{kl}^{(1)}) = \frac{1}{1 + \exp(-v_{kd}^{(1)})}, \\
\pi_{pd}^{(2)} = \Psi(v_{pd}^{(2)}) \prod_{l=1}^{K-1} \Psi(v_{pl}^{(2)}) = \frac{1}{1 + \exp(-v_{pd}^{(2)})},
\]

where \( 0 < \Psi(v_{kd}^{(1)}) , \Psi(v_{pd}^{(2)}) < 1 \) is a classic logistic function, which satisfies \( \Psi(-v_{kd}^{(1)}) = 1 - \Psi(v_{kd}^{(1)}) \), \( \Psi(-v_{pd}^{(2)}) = 1 - \Psi(v_{pd}^{(2)}) \), and \( v_{kd}^{(1)}, v_{pd}^{(2)} \) serves as the prior of \( \pi_{kd}^{(1)}, \pi_{pd}^{(2)} \).

In order to capture the correlations between action-topics and object-topics, we draw the packed vector \( v_{d} = [v_{d1}, ..., v_{dK}] \) in the stick-breaking notion from a multivariate normal distribution \( N(\mu, \Sigma) \). In practice, we use a truncated vector \( v_{d} = [v_{d1}, ..., v_{dK-1}] \) for (K-1) topics, and set \( \pi_{kd}^{(1)} = 1 - \sum_{k=1}^{K-1} \pi_{kd}^{(1)} = \prod_{k=1}^{K-1} \Psi(-v_{kd}^{(1)}) \) as the probability of the final topic for a valid distribution. The same for \( v_{pd}^{(2)} \).

Relative time distributions. The temporal relations between actions are also useful to discriminating the actions using temporal ordering and inferring the forgotten actions using the temporal context. We model the relative time of occurring actions by taking their time stamps into account. We consider that the relative time between two words are drawn from a certain distribution according to their topic assignments. In detail, let \( t_{nd} \), \( t_{md} \in (0, 1) \) be the absolute time stamp of \( n \)-th word and \( m \)-th word, which is normalized by the video length. \( t_{mnd} = t_{md} - t_{nd} \) is the relative time of \( m \)-th clip relative to \( n \)-th clip. Then \( t_{mnd} \) is drawn from a certain distribution, \( t_{mnd} \sim \Omega(t|\theta_{k,l}) \), where \( \theta_{k,l} \) are the parameters. \( \Omega(t|\theta_{k,l}) \) is \( K \times K \) pairwise action-topic specific relative time distributions defined as follows:

\[
\Omega(t|\theta_{k,l}) = \begin{cases} 
 b_{k,l} \cdot N(t|\theta_{k,l}^{(1)}) & \text{if } t \geq 0, \\
 1 - b_{k,l} \cdot N(t|\theta_{k,l}^{(2)}) & \text{if } t < 0,
\end{cases}
\]

An illustration of the learned relative time distributions are shown in Fig. 5. We can see that the distributions we learned correctly reflect the order of the actions, e.g., put-back-to-fridge is after pour and can be before/after microwave, and the shape is almost similar to the real distributions. Here the Bernoulli distribution \( b_{k,l} / 1 - b_{k,l} \) gives the probability of action \( k \) after/before the action \( l \). And two independent normal distributions \( N(t|\theta_{k,l}^{(1)}) / N(t|\theta_{k,l}^{(2)}) \) estimate how long the action \( k \) is after/before the action \( l \). Then the order and the length of the actions will be captured by all these pairwise relative time distributions.

3 Specially, when \( k = l \), If two words are in the same segments, we draw \( t \) from a normal distribution which is centered on zero, and the variance models the length of the action. If not, if it also follows Eq. (4) indicating the relative time between two same actions. We also use functions \( \tan((\pi/2 + \pi)t)(0 < t < 1) \), \( \tan((\pi/2 + \pi))(1 < t < 0) \) to feed \( t \) to the normal distribution so that the probability is valid, that sums to one through the domain of \( t \).

![Fig. 4: The graphic model of our causal topic model.](image)

![Fig. 5: The relative time distributions learned by our model on training set (the blue dashed line) and the ground-truth histogram of the relative time over the whole dataset (the green solid line).](image)

| Symbols | Meaning |
|---------|---------|
| \( D \) | number of videos in the training database; |
| \( K \) | number of action-topics; |
| \( P \) | number of object-topics; |
| \( N \) | number of human-words/object-words in a video; |
| \( c_{nd} \) | \( n \)-th clip in \( d \)-th video; |
| \( w_{nd} \) | \( n \)-th human-word in \( d \)-th video; |
| \( w_{o} \) | \( n \)-th object-word in \( d \)-th video; |
| \( z_{v}^{(1)} \) | action-topic assignment of \( v_{d} \); |
| \( z_{o}^{(2)} \) | object-topic assignment of \( o_{d} \); |
| \( t_{md} \) | normalized timestamp of \( m \)-th clip relative to \( n \)-th clip; |
| \( \pi_{kd}^{(1)} \) | the probabilities of action-topics in \( d \)-th document; |
| \( \pi_{pd}^{(2)} \) | the priors of action-topics in \( d \)-th document; |
| \( \beta_{k}^{(1)} \) | multinomial human-word distribution from action-topic \( k \); |
| \( \phi_{k,l}^{(12)} \) | multinomial object-word distribution from action-topic \( k \) and object-topic \( l \); |
| \( \mu, \Sigma \) | multivariate normal distribution of \( v_{d} = [v_{d1}, ..., v_{dK}] \); |
| \( \theta_{k,l} \) | relative time distribution of \( t_{mnd} \) between action-topic \( k, l \); |

TABLE 1: Notations in our model.
System Pipeline.

(a) Our Watch-Bot system. It consists of a Kinect v2 sensor that inputs RGB-D frames of human actions, a laptop that infers the forgotten action and the related object, a pan/tilt camera that localizes the object, mounted with a fixed laser pointer that points out the object.

(b) The system pipeline. The robot first uses the learned model to infer the forgotten action and the related object based on the Kinect’s input. Then it maps the view from the Kinect to the pan/tilt camera so that the bounding box of the object is mapped in the camera’s view. Finally, the camera moves until the laser spot lies in the bounding box of the target object.

5.1 Learning and Inference

Gibbs sampling is commonly used as a means of statistical inference to approximate the distributions of variables when direct sampling is difficult \cite{7, 23}. Given a video, the word \( w_{nd}^h, w_{nd}^o \) and the relative time \( t_{mn} \) are observed. We can integrate out \( \Phi^{(1)}, \Phi^{(2)} \) since \( \text{Dir}(\beta^{(1)}), \text{Dir}(\beta^{(2)}) \) are conjugate priors for the multinomial distributions \( \Phi^{(1)}, \Phi^{(2)} \). We also estimate the standard distributions including the multivariate normal distribution \( N(\mu, \Sigma) \) and the time distribution \( \Omega(\theta) \) using the method of moments, once per iteration of Gibbs sampling. Following the convention, we use the fixed symmetric Dirichlet distributions by setting \( \beta^{(1)}, \beta^{(2)} = 0.01 \).

Then we introduce how we sample the topic assignment \( z_{nd}^{(1)}, z_{nd}^{(2)} \). We do a collapsed sampling as in LDA by calculating the posterior distribution of \( z_{nd}^{(1)}, z_{nd}^{(2)} \):

\[
p(z_{nd}^{(1)} = k | z_{nd}^{(2)}, z_{nd}^{(1)}, z_{nd}^{(2)}) = \frac{\pi_k^{(1)}}{\sum_k \pi_k^{(1)} + \beta^{(1)}}
\]

\[
p(z_{nd}^{(2)} = p | z_{nd}^{(2)}, z_{nd}^{(1)}, z_{nd}^{(2)}, k, w_{nd}^{o}) = \frac{\pi_p^{(2)} \omega(k, w_{nd}^{o}) p(t_{mn} | z_{nd}^{(2)}, \theta)}{\sum_p \pi_p^{(2)} \omega(k, w_{nd}^{o})},
\]

\[
\omega(k, w_{nd}^{o}) = \frac{N_{kw}^{o}}{N_{k}} \times \left[ N_{k} - N_{kw}^{o} + \beta^{(1)}\right],
\]

\[
\omega(k, p, w_{nd}^{o}) = \frac{N_{kpw}^{o}}{N_{kp}} \times \left[ N_{kp} - N_{kpw}^{o} + \beta^{(2)}\right],
\]

\[
p(t_{mn} | z_{nd}^{(2)}, \theta) = \prod_m \Omega(t_{mn}) \Omega(t_{mn} | \theta),
\]

where \( N_{kw}^{o} / N_{kp}^{o} \) denotes the number of unique word types in dictionary, \( N_{kw}^{o} / N_{kp}^{o} \) denotes the number of instances of word \( w_{nd}^{o} / w_{nd}^{o} \) assigned with action-topic \( k \) and object-topic \( p, \) excluding \( n \)-th word in \( d \)-th document and \( N_{k} / N_{kp} \) denotes the number of total words assigned with action-topic \( k \) and object-topic \( p. \) The detailed derivation of Eq. \( 2 \) is in the Appendix A.

In Eq. \( 2 \), note that the topic assignments are decided by which actions/objects are more likely to co-occur in the video (the occurrence probabilities \( \pi_k^{(1)} / \pi_k^{(2)} \), the visual appearance of the word (the word distributions \( \omega(k, w_{nd}^{o}) \) and the temporal relations (the relative time distributions \( p(t_{mn} | z_{nd}^{(1)}), \theta) \)).

Due to the logistic stick-breaking transformation, the posterior distribution of the topic priors \( v_{d} = [v_{d}^{(1)}, v_{d}^{(2)}] \) does not have a closed form. So we instead use a Metropolis-Hastings independence sampler \cite{15}. Let the proposals \( q(v_{d}^{(1)} | v_{d}^{(1)}, \mu, \Sigma) = N(v_{d}^{(1)} | \mu, \Sigma) \) be drawn from the prior. The proposal is accepted with probability \( \min(A(v_{d}^{(1)}, v_{d}^{(1)}), 1) \), where

\[
A(v_{d}^{(1)}, v_{d}^{(1)}) = \frac{p(v_{d}^{(1)} | v_{d}^{(1)}, \mu, \Sigma) \prod_{k=1}^{N_{d}} p(z_{nd}^{(1)} | v_{d}^{(1)}) \prod_{k=1}^{N_{d}} p(z_{nd}^{(2)} | v_{d}^{(2)}) \prod_{k=1}^{N_{d}} q(v_{d}^{(2)} | v_{d}^{(1)}) \prod_{k=1}^{N_{d}} q(v_{d}^{(1)} | v_{d}^{(1)})}{\prod_{k=1}^{N_{d}} p(z_{nd}^{(1)} | v_{d}^{(1)}) \prod_{k=1}^{N_{d}} p(z_{nd}^{(2)} | v_{d}^{(2)}) \prod_{k=1}^{N_{d}} q(v_{d}^{(2)} | v_{d}^{(1)}) \prod_{k=1}^{N_{d}} q(v_{d}^{(1)} | v_{d}^{(1)})},
\]

which can be easily calculated by counting the number of words assigned with each topic by \( z^{(1)}_{nd}, z^{(2)}_{nd} \). Here the function \( \delta(x, y) = 1 \) if only if \( x = y \), otherwise equal to 0. The time complexity of the sampling per iteration is \( O(N_{d}D(\max(N_{d}K, P))) \).

For inference of a test video, we sample the unknown topic assignments \( z_{nd}^{(1)}, z_{nd}^{(2)} \) and the topic priors \( v_{d}^{(1)}, v_{d}^{(2)} \) using the learned parameters in the training stage.

6 Watch-Bot to Reminding of Forgotten Actions

The average adult forgets three key facts, chores or events every day \cite{2}. So it is important for a personal robot to be able to detect not only what a human is currently doing but also what he forgot to do. In this section, we describe a new robot system (see Fig. 6).
to detect the forgotten actions and remind people, which we called action patching, using our learning model.

Note that detecting forgotten action is more challenging than conventional action recognition, since what to infer is not shown in the query video. Also, our model does not necessarily know the semantic class of the actions. Instead it learns action clusters and relations from the unlabeled action videos and use them to detect forgotten actions and remind people. Therefore, modeling rich relations from videos is important to providing evidence for detecting forgotten actions. Our model models pairwise co-occurrence and long-range temporal relations of actions/topics. As a result, rather than only modeling the single action or the local temporal transitions in the previous works, those actions occurred with a relatively large time interval, occurred after the forgotten actions, as well as the interacting objects can also be used to detect forgotten actions in our model. For example, a put-back-book might be forgotten as previously seen a fetch-book action before a long read action, and seen a book and a leave action indicates he really forgot it.

We enable a robot, that we call Watch-Bot, to detect humans’ forgotten actions as well as localize the related object in the current scene. The robot consists of a Kinect v2 sensor, a pan/tilt camera (which we call camera for brevity in this paper) mounted with a laser pointer, and a laptop (see Fig. 5). This setup can be easily deployed on any assistive robot. Taking the example in Fig. 1, if our robot sees a person fetch a milk from the fridge, pour the milk, and leave without putting the milk back to the fridge. Our robot would first detect the forgotten action and the related object (the milk), given the input RGB-D frames and human skeletons from the Kinect; then map the object from the Kinect’s view to the camera’s view; finally pan/tilt the camera till its mounted laser pointer pointing to the milk.

Our goal is to detect the forgotten action and then point out the related object in the forgotten action using our learned model (see Alg. 1). We first use our model to segment the query video into action segments (step 1.2 in Alg. 1), and then infer the most possible forgotten action-topic and the related object-topic (step 4 in Alg. 1). Next we retrieve a top forgotten action segment from the training database, containing the inferred forgotten action-topic and the object-topic (step 5.6 in Alg. 1). Using the extracted object in the retrieved segment, we detect the bounding box of the related object in the forgotten action segment candidate set $Q$, which contains segments with topics $(k_m, p_m)$; 5. Select the top forgotten action segment candidate set $Q$ which contains segments with topics $(k_m, p_m)$; 6. Select the top forgotten action segment $p$ from $Q$ with the maximum patch_score value; 7. If patch_score($p$) is smaller than a threshold, claim no action forgotten and return; 8. Segment the current frame to super-pixels using edge detection [12] as in Section 3; 9. Select the nearest super-pixels to both extracted object bounding box in $p$ and $q$. 11. Return the top forgotten action segment and the object bounding box.

**Algorithm 1 Forgotten Action and Object Detection.**

**Input:** RGB-D video $q$ with tracked human skeletons.

**Output:** Claim no action forgotten, or output an action segment with the forgotten action and a bounding box of the related object in the current scene.

1. Assign the action-topics to clips and the object-topics to object-words in $q$ as introduced in Section 5.1
2. Get the action segments by merging the continuous clips with the same assigned action-topic.
3. If the assigned action-topics $K_e$ in $q$ contains all modeled action-topics $[1 : K]$, claim no action forgotten and return;
4. For each action segmentation point $t_s$, each not assigned action-topic $k_m \in [1 : K] - K_e$, and each object-topic $p_m \in [1 : P]$;
5. For each action segmentation point $t_s$, each not assigned action-topic $k_m \in [1 : K] - K_e$, and each object-topic $p_m \in [1 : P]$:

Compute the probability defined in Eq. 3.

- $K_e$ is the set of segmentation points ($t_1$, $t_2$ in Fig. 7) and $K_e$ is the set of existing action-topics in the video (fetch-book, etc. in Fig. 7). Thus $[1 : K] - K_e$ are the missing topics in the video (put-down-items, etc. in Fig. 7).
- $p(t_s)_{\{1\}, \theta}$, $\omega(k_m, w^h), \omega(k_m, p_m, w^o)$ can be computed as in Eq. 2. Here we marginalized $w^h, w^o$ to avoid the effect of a specific human-word or object-word. Note that, $\bar{v}_{kd}, \bar{v}_{pd}$ gives the probability of a missing action-topic with an object-topic in the video decided by the correlation we learned in the joint distribution prior, i.e., the close topics have higher probabilities to occur in this query video. And $p(t_s)_{\{1\}, \theta}$ measures the temporal consistency of adding a new action-topic. And the marginalized word-topic distribution $\sum_{w^h, w^o} \omega(k_m, w^h) \omega(k_m, p_m, w^o)$ give the likelihood of the topic learned from training data.

**Patched Action and Object Inference.** Our model infers the forgotten action using the probability inference based on the dependencies between actions and objects. After assigning the action-topics and object-topics to a query video $q$, we consider adding one additional clip $\hat{c}$ consisting of $w^h, \hat{w}^o$ into $q$ in each action segmentation point $t_s$ (see Fig. 2). Then the probabilities of the missing action-topics $k_m$ with object-topics $p_m$ in each segmentation point $t_s$ can be computed following the posterior distribution in Eq. 2:

$$p(\bar{v}_{kd}^{(1)}, \bar{v}_{pd}^{(2)}) \propto \pi_{k_m; p_m} \pi_{p_m} \omega(k_m, w^h) \omega(k_m, p_m, w^o),$$

$$s.t. \quad t_s \in T_s, \ k_m \in [1 : K] - K_e,$$

where $T_s$ is the set of segmentation points ($t_1$, $t_2$ in Fig. 7) and $K_e$ is the set of existing action-topics in the video (fetch-book, etc. in Fig. 7). Thus $[1 : K] - K_e$ are the missing topics in the video (put-down-items, etc. in Fig. 7). $p(t_s)_{\{1\}, \theta}$, $\omega(k_m, w^h), \omega(k_m, p_m, w^o)$ can be computed as in Eq. 2. Here we marginalized $w^h, w^o$ to avoid the effect of a specific human-word or object-word. Note that, $\bar{v}_{kd}, \bar{v}_{pd}$ gives the probability of a missing action-topic with an object-topic in the video decided by the correlation we learned in the joint distribution prior, i.e., the close topics have higher probabilities to occur in this query video. And $p(t_s)_{\{1\}, \theta}$ measures the temporal consistency of adding a new action-topic. And the marginalized word-topic distribution $\sum_{w^h, w^o} \omega(k_m, w^h) \omega(k_m, p_m, w^o)$ give the likelihood of the topic learned from training data.

**Patched Action and Object Detection.** Then we select the top three tuples $(k_m, p_m, t_s)$ using the above probability. The action segments of action-topic $k_m$ containing object-topic $p_m$ in the training set consist a patched action candidate segment set $Q$. We then select the patched action segment from $Q$ with the maximum patch_score defined in Eq. 3. In detail, we consider that the front and the tail of the patched action segment $f_{pf}, f_{pt}$ should be similar to the tail of the adjacent segment in $q$ before $t_s$ and the front of the adjacent segment in $q$ after $t_s$: $f_{qt}, f_{qf}$. At the same time, the middle of the patched action segment $f_{pm}$ should be different to $f_{pt}, f_{qf}$, as it is a different action forgotten in the video:

$$\text{patch_score}(p) = \text{ave}(D(f_{pf}, f_{qf}), \ D(f_{pm}, f_{qf})) - \text{max}(D(f_{pf}, f_{qf}), \ D(f_{pm}, f_{qf})),$$

where $D(\cdot)$ is the average pairwise distances between frames, $\text{ave}(\cdot), \text{max}(\cdot)$ are the average and max value. If the maximum distance is the average pairwise distances between frames, $\text{ave}(\cdot), \text{max}(\cdot)$ are the average and max value. If the maximum distance is too large, we consider the action forgotten.

4. Here the middle, front, tail frames are 20%-length of segment centering on the middle frame, starting from the first frame, and ending at the last frame in the segment respectively.
human skeletons (25 × actions interacting with different objects (see examples in Fig. 12). We collect a new challenging RGB-D activity dataset recorded by the new Kinect v2 camera. Each video in the dataset contains 2-7 actions interacting with different objects.

### 7.1 Watch-n-Patch Dataset

In order to get a variation in activities, we ask participants to finish task with different combinations of actions and ordering. Some actions occur together often such as fetch-from-fridge and put-back-to-fridge while some are not always in the same video such as take-item and read. Some actions are in fixed ordering such as fetch-book and put-back-book while some occur in random order such as put-back-to-fridge and microwave. Moreover, to evaluate the action patching performance, 222 videos in the dataset has action forgotten by people and the forgotten actions are annotated. We give the examples of action classes in Fig. 12 and action sequences in Table 4.

#### 7.2 Experimental Setting and Compared Baselines

We evaluate in two environments ‘office’ and ‘kitchen’. In each environment, we split the data into a train set with most full videos (office: 87, kitchen 119) and a few forgotten videos (office: 10, kitchen 10), and a test set with a few full videos (office: 10, kitchen 20) and most forgotten videos (office: 89, kitchen 113). In our experiments, we compare seven unsupervised approaches with only action-topics. They are Hidden Markov Model (HMM), topic model LDA (TM), correlated topic model (CTM), topic model over absolute time (TM-AT), correlated topic model over absolute time (CTM-AT), topic model over relative time (TM-RT) and our causal topic model with only action-topics (CaTM-A). We compare three methods with both action-topics and object-topics. They are HMM with the object-topics (HMM-O), LDA with the object-topics (TM-O) and our causal topic model with the object-topics (CaTM-AO). All these methods use the same human skeleton and RGB-D features introduced in Section 4. We also evaluate HMM and our model CaTM using the popular features for action recognition, dense trajectories feature (DTF) named as HMM-DTF and CaTM-A-DTF.

In the experiments, we set the number of topics and states of HMM equal to or more than ground-truth classes. For correlated topic models, we use the same topic prior in our model. For models over absolute time, we consider the absolute time of each word is drawn from a topic-specific normal distribution. For models over relative time, we use the same relative time distribution as in our model (Eq. 1). The clip length of the action-words is set to 20 frames, densely sampled by step one and the size of action dictionary is set to 500. For patching, the candidate set for different approaches consist of the segments with the inferred missing topics by transition probabilities for HMM, the topic priors for TM and CTM, and both the topic priors and the time distributions for TM-AT, TM-RT, CTM-AT and our CaTM. Then we use the same patch_score as in Section 6 to select the top one patched segments, and the average of the patch_score computed in a set of the segmented videos after training is set as the threshold of claiming forgotten action.

#### 7.3 Evaluation Metrics

**Action Segmentation and Cluster Assignment.** We want to evaluate if the unsupervised learned action-topics and states of HMM are semantically meaningful. In the unsupervised setting, we need to map the assigned topics to the ground-truth labels for evaluation. This could be done by counting the mapped frames in a set of the segmented videos after training is set as the threshold of claiming forgotten action.

5. We train a codebook with the size of 2000 and encode the extracted DTF features in each clip as the bag of features using the codebook.
between topics and ground-truth classes. Let \(k_i, c_i\) be the assigned topic and ground-truth class of frame \(i\). The count of a mapping is:

\[
m_{kc} = \sum_i \delta(k_i, k) \delta(c_i, c)
\]

where \(\sum_i \delta(k_i, k)\delta(c_i, c)\) is the number of frames assigned with topic \(k\) as the ground-truth class \(c\) and normalized by the number of frames as the ground-truth class \(c\). \(\sum_i \delta(c_i, c)\). Then we can solve the following binary linear programming to get the best mapping:

\[
\max \sum_{k,c} x_{kc} m_{kc},
\]

s.t. \(\forall k, \sum_c x_{kc} = 1\), \(\forall c, \sum_k x_{kc} \geq 1, \quad x_{kc} \in \{0, 1\}\),

where \(x_{kc} = 1\) indicates mapping topic \(k\) to class \(c\) otherwise \(x_{kc} = 0\). And \(\sum_k x_{kc} = 1\) constrain that each topic must be mapped to exact one class, \(\sum_c x_{kc} \geq 1\) constrain that each class must be mapped by at least one topic.

We then measure the performance in two ways. Per frame: we compute frame-wise accuracy (Frame-Acc), the ratio of correctly labeled frames. We consider a true positive if the overlap rate (union/intersection) between the detected and the ground-truth object bounding box is greater than 40%.

### 7.4 Results

Table 2 and Fig. 8 show the main results of our experiments. We first perform evaluation in the offline setting to see if actions can be well segmented and clustered in the train set. We then perform testing in an online setting to see if the new video from the test set can be correctly segmented and the segments can be correctly assigned to the action cluster. We can see that our approach performs better than the state-of-the-art in unsupervised action segmentation and clustering, as well as action patching. We discuss our results in the light of the following questions.

**Did modeling the long-range relations help?** We studied whether modeling the correlations and the temporal relations between topics was useful. The approaches considering the temporal relations, HMM, TM-RT, and our CaTM, outperform other approaches which assume actions are temporal independent. This demonstrates that understanding temporal structure is critical to recognizing and patching actions. The approaches, TM-RT and CaTM, which model both the short-range and the long-range relations perform better than HMM only modeling local relations. Also, the approaches considering the topic correlations CTM, CTM-AT, and our CaTM perform better than the corresponding non-correlated topic models TM, TM-AT, and TM-RT. Our CaTM, which considers both the action correlation priors and the temporal relations, shows the best performance.

**How successful was our unsupervised approach in learning meaningful action-topics?** From Table 2 we can see that the unsupervised learned action-topics is promising to be semantically
meaningful even though ground-truth semantic labels are not provided in the training. In order to qualitatively estimate the performance, we give a visualization of our learned topics in Fig. 11. It shows that the actions with the same semantic meaning are clustered together though they are in different views and motions. In addition to the one-to-one correspondence between topics and semantic action classes, we also plot the performance curves varied with the topic number in Fig. 8. It shows that if we set the topics a bit more than ground-truth classes, the performance increases since a certain action might be divided into multiple action-topics. But as topics increase, more variations are also introduced so that performance saturates.

**RGB videos vs. RGB-D videos.** In order to compare the effect of using information from RGB-D videos, we also evaluate our model CaTM and HMM using the popular RGB features for action recognition (CaTM-A-DTF, CaTM-AO-DTF and HMM-DTF in Table 2). Clearly, the proposed human skeleton and RGB-D features outperform the DTF features as more accurate human motion and object are extracted.

**How well did our new application of action patching perform?** From Table 2, we find that the approaches learning the action relations mostly give better patching performance. This is because the learned co-occurrence and temporal structure strongly help indicate which actions are forgotten. Our model capturing both the short-range and long-range action relations shows the best results.

**How important is it to consider relations between actions and objects?** From the results, we can see that the model which did well in forgotten action detection also performed well in detecting forgotten object. Since our model CaTM-AO well considers the relations between the action and the object, it shows better performance in both forgotten action and forgotten object detection than those which models action and object independently as well as CaTM-A which only models the actions.

### 7.5 Robotic Experiments

In this section, we show how our Watch-Bot reminds people of the forgotten actions in the real-world scenarios. We test each two forgotten scenarios in ‘office’ and ‘kitchen’ respectively (put-back-book, turn-off-monitor, put-milk-back-to-fridge and fetch-food-from-microwave). We use a subset of the dataset to train the model in each activity type separately. In each scenario, we ask 3 subjects to perform the activity twice. Therefore, we test 24 trials in total. We evaluate three aspects. One is objective, the success rate (Succ-Rate): the laser spot lying within the object as correct. The other two are subjective, the average Subjective Accuracy Score (Subj-AccScore): we ask the participant if he thinks the pointed object is correct; and the average Subjective Helpfulness Score (Subj-HelpScore): we ask the participant if the output of the robot is helpful. Both of them are in $1-5$ scale, the higher the better.

**Table 3** gives the results of our robotic experiments. We can see that our robot can achieve over $60\%$ success rate and gives the best performance. In most cases people think our robot is able to help them understand what is forgotten. Fig. 10 gives an example of our experiment, in which our robot observed what a human is currently doing, realized he forgot to fetch food from microwave and then correctly pointed out the microwave in the scene.

### 8 Conclusion and Future Work

In this paper, we presented an algorithm that models the human activities in a completely unsupervised setting. We showed that it is important to modeling the long-range relations between...
where we denote the Beta function as $\Gamma(\theta) / \Gamma(\sum_{k=1}^K \theta_k)$. The topic-pair relative time distribution:

$$p(t|z^{(1)}, \theta) = \frac{D}{d=1} \frac{N_d}{n=1} p(t_{nd}|z^{(1)}_d, \theta)$$

$$= \frac{D}{d=1} \frac{N_d}{n=1} \prod_{m=1}^{N_d} p(t_{mnd}|\theta^{(1)}_{z^{(1)}_{mnd}}, \theta^{(1)}_{z^{(1)}_{mnd}}).$$

topic priors:

$$p(z^{(1)}|\pi^{(1)}) = \frac{D}{d=1} \frac{N_d}{n=1} \prod_{m=1}^{N_d} \pi_{z^{(1)}_{mnd}}^{(1)}$$

$$p(z^{(2)}|\pi^{(2)}) = \frac{D}{d=1} \frac{N_d}{n=1} \prod_{m=1}^{N_d} \pi_{z^{(2)}_{mnd}}^{(2)}.$$
Fig. 12: Examples of every action class in our dataset. The left is RGB frame and the right is depth frame with human skeleton (yellow).

REFERENCES

[1] Kinect v2 sensor. http://www.microsoft.com/en-us/kinectforwindows/develop/

[2] Adults forget three things a day, research finds. http://www.telegraph.co.uk/news/uknews/5891701/Adults-forget-three-things-a-day-research-finds.html, 2009. The Daily Telegraph.

[3] J. Aggarwal and M. Ryoo. Human activity analysis: A review. ACM Comput. Surv., 43(3):16:1–16:43, 2011.

[4] S. M. Assari, A. R. Zamir, and M. Shah. Video classification using semantic concept co-occurrences. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[5] S. Bhattacharya, M. M. Kalayeh, R. Sukthankar, and M. Shah. Recognition of complex events: Exploiting temporal dynamics between underlying concepts. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014.

[6] D. M. Blei and J. D. Lafferty. A correlated topic model of science. The Annals of Applied Statistics, 1(1):17–35, 2007.

[7] D. M. Blei and J. D. Lafferty. Topic models. Text mining: classification, clustering, and applications, 10:71, 2009.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, 2003.

[9] P. Bojanowski, R. Lajugie, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic. Weakly supervised action labeling in videos under ordering constraints. In European Conference on Computer Vision (ECCV), 2014.

[10] G. Chen, M. Giuliani, D. S. Clarke, A. K. Gaschler, and A. Knoll. Action recognition using ensemble weighted multi-instance learning. In International Conference on Robotics and Automation (ICRA), 2014.

[11] A. Chirungoo, S. Manimaran, and B. Ravindran. Activity recognition for natural human robot interaction. In Social Robotics, volume 8755, pages 84–94, 2014.

[12] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In International Conference on Computer Vision (ICCV), 2013.

[13] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In European Conference on Computer Vision (ECCV), 2009.

[14] T. A. Faruquie, P. K. Kalra, and S. Banerjee. Time based activity inference using latent dirichlet allocation. In British Machine Vision Conference (BMVC), 2009.

[15] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. Bayesian data analysis. CRC press, 2013.
TABLE 4: Action sequence examples in our dataset. The action ‘walk’ is omitted as it can be between any actions. Possible forgotten actions are in the brackets.

**office:**
- read → leave-office
- fetch-book → read → [put-back-book] → leave-office
- put-down-item → read → [take-item] → leave-office
- put-down-item → fetch-book → read → [put-back-book] → [take-item] → leave-office
- put-down-item → fetch-book → read → [put-back-book] → [take-item] → leave-office
- put-down-item → fetch-book → read → [put-back-book] → [take-item] → leave-office
- turn-on-monitor → play-computer → [turn-off-monitor] → leave-office
- put-down-item → turn-on-monitor → play-computer → [turn-off-monitor] → [take-item] → leave-office
- turn-on-monitor → put-down-item → play-computer → [turn-item] → [turn-off-monitor] → leave-office
- turn-on-monitor → put-down-item → play-computer → [turn-item] → [turn-off-monitor] → leave-office

**kitchen:**
- pour → drink
- pour → [drink] → leave-kitchen
- fetch-from-fridge → pour → put-back-to-fridge → drink
- fetch-from-fridge → pour → [drink] → leave-office
- fetch-from-fridge → pour → [drink] → [take-item] → leave-office
- fetch-from-fridge → prepare-food → microwave → [fetch-from-microwave] → leave-office
- fetch-from-fridge → prepare-food → microwave → [put-back-to-fridge] → [fetch-from-microwave] → leave-office
- move-kettle → fill-kettle → move-kettle → [plug-in-kettle]

[16] M. Hoai, Z. zhong Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
[17] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for rgb-d activity recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[18] N. Hu, Z. Lou, G. Englebienne, and B. Krse. Learning to recognize human activities from soft labeled data. In Proceedings of Robotics: Science and Systems (RSS), 2014.
[19] A. Jain, A. Gupta, M. Rodriguez, and L. Davis. Representing videos using mid-level discriminative patches. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[20] S. Jones and L. Shao. Unsupervised spectral dual assignment clustering of human actions in context. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[21] V. Kantorov and I. Laptev. Efficient feature extraction, encoding and classification for action recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[22] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In European Conference on Computer Vision (ECCV), 2005.
[23] D. I. Kim and E. B. Sudderth. The doubly correlated nonparametric topic model. In Advances in Neural Information Processing Systems (NIPS), 2011.
[24] A. Kläser, M. Marszalek, C. Schmid, and A. Zisserman. Human focused action localization in video. In International Workshop on Sign, Gesture, and Activity (SGA) in Conjunction with ECCV, 2010.
[25] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from RGB-D videos. I. J. Robotic Res., 32(8):951–970, 2013.
[26] H. S. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. In Robotics: Science and Systems (RSS), 2013.
[27] H. S. Koppula and A. Saxena. Learning spatio-temporal structure from RGB-D videos for human activity detection and anticipation. In International Conference on Machine Learning (ICML), 2013.
[28] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering complex event detection. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[29] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering complex event detection. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[30] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering complex event detection. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[31] H. S. Koppula and A. Saxena. Learning human activities by attributes. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
[32] M. Losch, S. Schmidt-Rohr, S. Knoop, S. Vacek, and R. Dillmann. Feature set selection and optimal classifier for human activity recognition. In Robot and Human Interactive Communication, 2007.
[33] S. Ma, L. Sigal, and S. Sclaroff. Space-time tree ensemble for action recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
by representing 3d skeletons as points in a lie group. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[52] N. N. Vo and A. F. Bobick. From stochastic grammar to bayes network: Probabilistic parsing of complex activity. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[53] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action Recognition by Dense Trajectories. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[54] X. Wang and Q. Ji. A hierarchical context model for event recognition in surveillance video. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[55] X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2006.

[56] C. Wu, I. Lenz, and A. Saxena. Hierarchical semantic labeling for task-relevant rgb-d perception. In Robotics: Science and Systems (RSS), 2014.

[57] C. Wu, J. Zhang, S. Savarese, and A. Saxena. Watch-n-patch: Unsupervised understanding of actions and relations. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[58] C. Wu, J. Zhang, B. Selman, S. Savarese, and A. Saxena. Watch-bot: Unsupervised learning for reminding humans of forgotten actions. In International Conference on Robotics and Automation (ICRA), 2016.

[59] D. Wu and L. Shao. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[60] S. Yang, C. Yuan, B. Wu, W. Hu, and F. Wang. Multi-feature max-margin hierarchical bayesian model for action recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[61] Y. Yang, Y. Li, C. Fermüller, and Y. Aloimonos. Robot learning manipulation action plans by watching unconstrained videos from the world wide web. In AAAI, 2015.

[62] Y. Yang, I. Saleemi, and M. Shah. Discovering motion primitives for unsupervised grouping and one-shot learning of human actions, gestures, and expressions. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI), 35(7):1635–1648, 2013.