Watch-n-Patch: Unsupervised Understanding of Actions and Relations

Chenxia Wu[1,2], Jiemi Zhang, Silvio Savarese[2], and Ashutosh Saxena[1,2]

[1]Department of Computer Science, Cornell University
[2]Department of Computer Science, Stanford University
{chenxiawu,asaxena}@cs.cornell.edu, jmzhang10@gmail.com, ssilvio@stanford.edu

Abstract

We focus on modeling human activities comprising multiple actions in a completely unsupervised setting. Our model learns the high-level action co-occurrence and temporal relations between the actions in the activity video. We consider the video as a sequence of short-term action clips, called action-words, and an activity is about a set of action-topics indicating which actions are present in the video. Then we propose a new probabilistic model relating the action-words and the action-topics. It allows us to model long-range action relations that commonly exist in the complex activity, which is challenging to capture in the previous works.

We apply our model to unsupervised action segmentation and recognition, and also to a novel application that detects forgotten actions, which we call action patching. For evaluation, we also 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 interacted with different objects. The extensive experiments show the effectiveness of our model.

1. Introduction

We consider modeling human activities containing a sequence of actions (see an example in Fig. 1), as perceived by an RGB-D sensor in home and office environments. In the complex human activity such as *warming milk* in the example, there are not only short-range action relations, *e.g.*, *microwaving* is often followed by *fetch-bowl-from-oven*, but there are also long-range action relations, *e.g.*, *fetch-milk-from-fridge* is strongly related to *put-milk-back-to-fridge* even though several other actions occur between them.

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

Most previous works focus on action detection in a supervised learning setting. In the training, they are given fully labeled actions in videos [23, 29, 31], or weakly supervised action labels [9, 7], or locations of human/their interactive objects [21, 35, 25]. Among them, the temporal structure of actions is often discovered by Markov models such as Hidden Markov Model (HMM) [34] and semi-Markov [12, 32], or by linear dynamical systems [3], or by hierarchical grammars [27, 37, 20, 39, 2], or by other spatio-temporal representations [15, 26, 17, 19]. Most of these works are based on RGB features and only model the short-range relations between actions (see Section 2 for details).

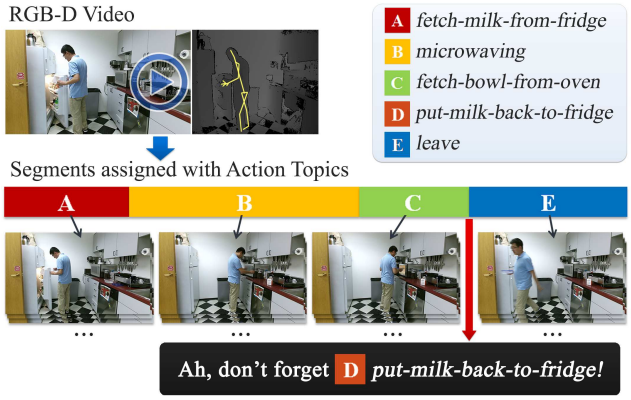


Figure 1: Our goal is to automatically segment RGB-D videos and assign action-topics to each segment. We propose a completely unsupervised approach to modeling the human skeleton and RGB-D features to actions, as well as the pairwise action co-occurrence and temporal relations. We then show that our model can be used to detect which action people forgot, a new application which we call *action patching*.

Different from 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 *fetch-milk-from-fridge*. We also use the more informative human skeleton and RGB-D features, which show higher performance over RGB only features for action recognition [18, 42, 22].

In order to capture the rich structure in the activity, we draw strong parallels with the work done on document modeling from natural language (e.g., [6]). We consider an activity video as a document, which consists of a sequence of short-term action clips as *action-words*. And 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. Action-words are drawn from these action-topics and has a distribution for each topic. Then we model the following (see Fig. 2):

- *Action co-occurrence.* Some actions often co-occur in the same activity. We model the co-occurrence by adding correlated topic priors to the occurrence of action-topics, *e.g.*, *fetch-milk-from-fridge* and *put-milk-back-to-fridge* has strong correlations.
- *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 activity videos. We use the model to temporally segment videos to segments assigned with action-topics. We show that these action-topics are semantically meaningful by mapping them to ground-truth action classes and evaluating the labeling performance.

We then also show that our model can be used to detect forgotten actions in the activity, a new application that we call *action patching*. We show that the learned co-occurrence and temporal relations are very helpful to infer the forgotten actions by evaluating the patching accuracy.

We also provide a new challenging RGB-D activity video dataset recorded by the new Kinect v2 (see examples in Fig. 8), in which the human skeletons and the audio are also recorded. It contains 458 videos of human daily activities as compositions of multiple actions interacted with different objects, in which people forget actions in 222 videos. They are performed by different subjects in different environments with complex backgrounds.

In summary, the main contributions of this work are:

- Our model is completely unsupervised and non-parametric, 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 recognition, as well as in a new application action patching.
- We provide a new challenging RGB-D activity dataset recorded by the new Kinect v2, which contains videos of multiple actions interacted with different objects.

2. Related Work

Most previous works on action recognition are supervised [21, 9, 26, 23, 29, 35, 7, 24]. Among them, the linear models [34, 12, 32, 3] are the most popular, which focus on modeling the action transitions in the activities. More complex hierarchical relations [27, 37, 20, 39] or graph relations [2] are considered in modeling actions in the complex activity. Although they 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.

There also exist some unsupervised approaches on action recognition. Yang *et al*. [43] develop a meaningful representation by discovering local motion primitives in an unsupervised way, then a HMM is learned over these primitives. Jones *et al*. [13] propose an unsupervised dual assignment clustering on the dataset recorded from two views.

Different from these approaches, we use the richer human skeleton and RGB-D features rather than the RGB action features [38, 14]. 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.

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 [31, 33, 36, 42, 22]. Besides of the human skeletons, we also detect the human interactive objects in an unsupervised way to provide more discriminate features. Object-in-use contextual information has been commonly used for recognizing actions [18, 19, 25, 39]. Most of them depend on correct object tracking or local motion changes. They lost the high-level action relations which can be captured in our model.

Our work is also related to the topic models. LDA [6] was the first hierarchical Bayesian topic model and widely used in different applications. The correlated topic models [4, 16] add the priors over topics to capture topic correlations. A topic model over absolute timestamps of words is proposed in [40] and has been applied to action recognition [10]. However, the independence assumption of different topics would lead to non smooth temporal segmentations. 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.

3. Overview

We outline our approach in this section (see approach pipline in 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 features and the human interactive object features from the clips (introduced in Section. 3.1), which show higher performance over RGB only features for action recognition [18, 42, 22].

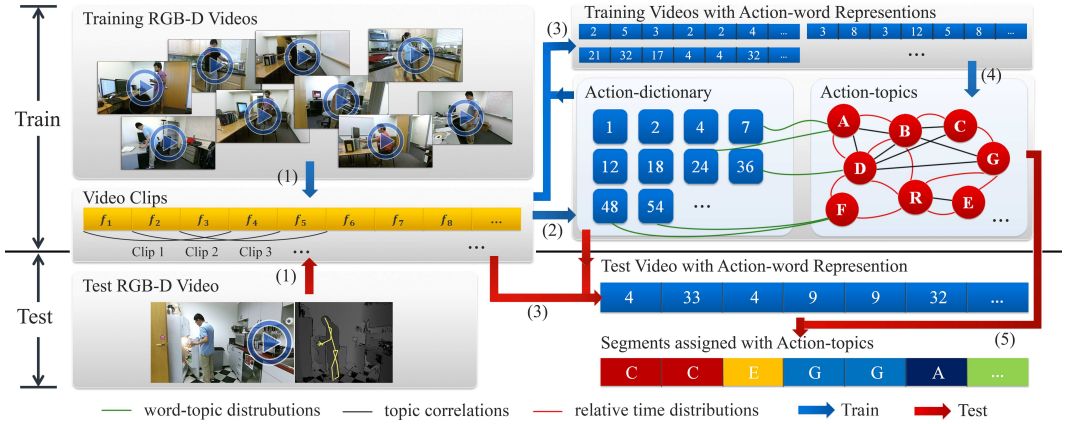


Figure 2: **The pipeline of our approach.** Training (blue arrows) follows steps (1), (2), (3), (4). Testing (red arrows) follows steps (1), (3), (5). The steps are: (1) Decompose the video into a sequence of overlapping fixed-length temporal clips. (2) Learn the action-dictionary by clustering the clips, where the cluster centers are action-words. (3) Map the clips to the action-words in the action-dictionary to get the action-word representation of the video. (4) Learn the model from the action-word representations of training videos. (5) Assign action-words in the video with action-topics using the learned model.

In order to build a compact representation of the action video, we draw parallels to document modeling in the natural language [6] to represent a video as a sequence of words. We use k-means to cluster the clips to form an *action-dictionary*, where we use the cluster centers as *action-words* (step (2)). Then, the video can be represented as a sequence of action-word indices by mapping its clips to the nearest action-words in the dictionary (step (3)). And an activity video is about a set of *action-topics* indicating which actions are present in the video.

We then build an unsupervised learning model (step (4)) that models the mapping of action-words to the action-topics, as well as the co-occurrence and the temporal relations between the action-topics. Using the learned model, we can assign the action-topic to each clip (step (5)), so that we can get the action segments, the continuous clips with the same assigned topic.

The unsupervised action-topic assignments of action-words are challenging because there is no annotations during the training stage. Besides extracting rich visual features, we well consider the relations between action-topics. Different from previous works, our model can capture long-range relations between actions *e.g.*, *put-milk-back-to-fridge* is strongly related to *fetch-milk-from-fridge* with *pouring* and *drinking* between them. We model all pairwise co-occurrence and temporal casual relations between occurring action-topics in the video, using a new probabilistic model (introduced in Section 4). Specifically, we use a joint distribution as the correlated topic priors. They estimate which actions are most likely to co-occur in a video. And we use a relative time distributions of topics to capture the temporal causal relations. They estimate the possible temporal ordering of the occurring actions in the video.

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

3.1. Visual Features

We describe how we extract the visual features of a clip in this sub-section. We extract both skeleton and object features from the output by the Kinect v2 [1], which has an improved body tracker and higher resolution of RGB-D frame than the Kinect v1. The skeleton has 25 joints in total. Let $X_c = \{x_1^c, x_2^c, \cdots, x_{25}^c\}$ be the 3D coordinates of 25 joints of a skeleton in the current frame. We first compute the cosine of the angles between the connected parts in each frame: $\alpha_i = p_{i+1} \cdot p_i / |p_{i+1}| \cdot |p_i|$, where $p_i = x_{i+1} - x_i$ is the body part. The change of the joint coordinates and angles can well capture the human body movements. So we extract the motion features and off-set features [42] by computing their Euclidean distances $\mathbb{D}(,)$ to previous frame $f_{c,c-1}^x, f_{c,c-1}^\alpha$ and the first frame $f_{c,1}^x, f_{c,1}^\alpha$ in the clip:

$$f_{c,c-1}^x = \{\mathbb{D}(x_i^c, x_i^{c-1})\}_{i=1}^{25},\ f_{c,c-1}^\alpha = \{\mathbb{D}(\alpha_i^c, \alpha_i^{c-1})\}_{i=1}^{25};$$

$$f_{c,1}^x = \{\mathbb{D}(x_i^c, x_i^1)\}_{i=1}^{25},\ f_{c,1}^\alpha = \{\mathbb{D}(\alpha_i^c, \alpha_i^1)\}_{i=1}^{25}.$$

Then we concatenate all $f_{c,c-1}^x, f_{c,c-1}^\alpha, f_{c,1}^x, f_{c,1}^\alpha$ as the human features of the clip.

We also extract the human interactive objects based on the human hands, motion detection and edge detection. The interactive objects can help discriminate the different human actions with similar body motions such as *fetch-book* and *turn-on-monitor*. To detect the interactive objects, first we segment each frame into super-pixels using a fast edge detection approach [8] on both RGB and depth images. The image segmentation provides richer candidate super-pixels

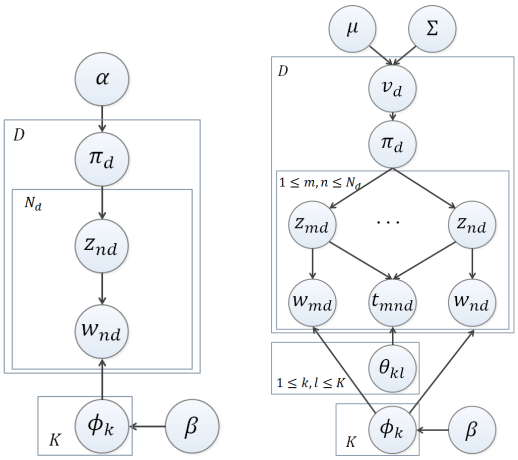


Figure 4: The graphic model of LDA (left) and our model (right).

rather than pixels to further extracting objects. Second we use a moving targets detection approach [28] to detecting foreground mask by removing the unnecessary steady backgrounds. Third we consider the interactive objects should be close to tracked human hands. Combining above three facts, we extract features from the image segments with more than 50% in the foreground mask and within a distance to the human hand joints in both 3D points and 2D pixels (see examples in Fig. 3). Then we extract six kernel descriptors from these image segments: 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 scene labeling [41]. We concatenate the human features and the object features as the final feature vector of a clip.

4. Learning Model

In order to incorporate the aforementioned properties of activities for patching, we present a new generative model (see the graphic model in Fig. 4-right and the notations in Fig. 5 and Table 1). The novelty of our model is the ability to infer the probability of forgotten actions in a complex activity video.

Consider a collection of D videos (documents in the topic model). Each video consists of N_d action-words $\{w_{nd}\}_{n=1}^{N_d}$ mapped to the action-dictionary. Consider K latent action-topics, z_{nd} is the topic assignment of each word, indicating which action-topic the action-word w_{nd} belongs to in the video. Then continuous action-words with the same topic in a video consist an action segment, and the segments assigned with the same topic from different videos consist an action-topic segment cluster.

The topic model such as LDA [6] has been very common for document modeling from language (see graphic model in Fig. 4-left), which generates a document using a mixture of topics. To model human actions in the video, our model introduces co-occurrence and temporal structure of topics instead of the topic independency assumption in LDA.

Table 1: Notations in our model.

Symbols	Meaning
D	number of videos in the training database;
K	number of action-topics;
N_d	number of words in a video;
w_{nd}	n-th word in d-th document;
z_{nd}	topic-word assignment of w_{nd};
t_{nd}	the normalized timestamp of of w_{nd};
t_{mnd}	$= t_{md} - t_{nd}$ the relative time between w_{md} and w_{nd};
$\pi_{:d}$	the probabilities of topics in d-th document;
$v_{:d}$	the priors of $\pi_{:d}$ in d-th document;
ϕ_k	the multinomial distribution of the word from topic k;
μ, Σ	the mutivariate normal distribution of $v_{:d}$;
θ_{kl}	the relative time distribution of t_{mnd}, between topic k, l;

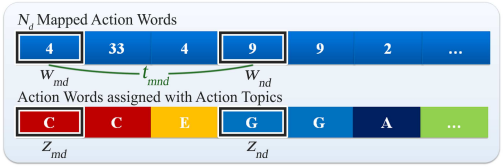


Figure 5: Notations in a video.

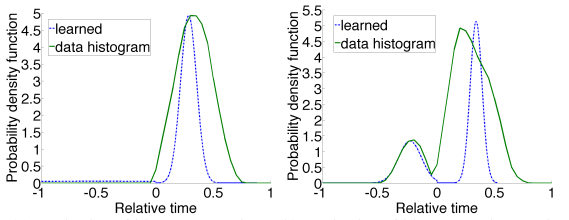
Basic generative process. In a document d, the topic assignment z_{nd} is chosen from a multinomial distribution with parameter $\pi_{:d}$, $z_{dn} \sim Mult(\pi_{:d})$, where $\pi_{:d}$ is sampled from a prior. And the word w_{nd} is generated by a topic-specific multinomial distribution $\phi_{z_{nd}}$, $w_{dn} \sim Mult(\phi_{z_{dn}})$, where $\phi_k \sim Dir(\beta)$ is the word distribution of topic k, sampled from a Dirichlet prior with the hyperparameter β.

Topic correlations. First we consider correlations between topics to model the probabilities of co-occurrence of actions. Let π_{kd} be the probability of topic k occurring in document d, where $\sum_{k=1}^{K} \pi_{kd} = 1$. Instead of sampling it from a fix Dirichlet prior with parameter α in LDA, we construct the probabilities by a stick-breaking process:

$$\pi_{kd} = \Psi(v_{kd}) \prod_{l=1}^{k-1} \Psi(v_{ld}),\ \Psi(v_{kd}) = \frac{1}{1+\exp(-v_{kd})},$$

where $0 < \Psi(v_{kd}) < 1$ is a classic logistic function, which satisfies $\Psi(-v_{kd}) = 1 - \Psi(v_{kd})$, and v_{kd} serves as the prior of π_{kd}. The vector $v_{:d}$ in a video are drawn from a mutivariate normal distribution $N(\mu, \Sigma)$, which captures the correlations between topics. In practice, $v_{:d} = [v_{1d}, \cdots, v_{K-1,d}]$ is a truncated vector for $K-1$ topics, then we can set $\pi_{Kd} = 1 - \sum_{k=1}^{K-1} \pi_{kd} = \prod_{k=1}^{K-1} \Psi(-v_{kd})$ as the probability of the final topic for a valid distribution of $\pi_{:d}$.

Relative time distributions. Second 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 word relative to n-th word (the green line in Fig. 5). Then t_{mnd} is drawn from a certain distribution, $t_{mnd} \sim \Omega(\theta_{z_{md}, z_{nd}})$, where $\theta_{z_{md}, z_{nd}}$ are the parameters. $\Omega(\theta_{k,l})$ are K^2 pairwise topic-specific relative



(a) *put-back-to-fridge* w.r.t. *pouring* (b) *put-back-to-fridge* w.r.t. *microwaving*

Figure 6: 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).

time distributions defined as follows:

$$\Omega(t|\theta_{k,l}) = \begin{cases} b_{k,l} \cdot N(t|\theta^+_{k,l}) & \text{if } t \geq 0, \\ 1 - b_{k,l} \cdot N(t|\theta^-_{k,l}) & \text{if } t < 0, \end{cases} \quad (1)$$

An illustration of the learned relative time distributions are shown in Fig. 6. We can see that the distributions we learned can correctly reflect the order of the actions, *e.g.*, *put-back-to-fridge* is after *pouring* and can be before/after *microwaving*, and the shape is mostly 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 $N(t|\theta^+_{k,l})/N(t|\theta^-_{k,l})$ estimate how long the action k is after/before the action l[1]. Then the order and the length of the actions will be captured by all these pairwise relative time distributions.

5. Gibbs Sampling for 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 [5, 16]. Given a video, the word w_{nd} and the relative time t_{mnd} are observed. In the training stage, given a set of training videos, we use Gibbs sampling to approximately sample other hidden variables from the posterior distribution of our model. Since we adopt conjugate prior $Dir(\beta)$ for the multinomial distributions Φ_k, we can easily integrate out Φ_k and need not to sample them. For simplicity and efficiency, we estimate the standard distributions including the mutivariate normal distribution $N(\mu, \Sigma)$ and the time distribution $\Omega(\theta_{kl})$ by the method of moments, once per iteration of Gibbs sampling. And as in many applications using the topic model, we use fixed symmetric Dirichlet distributions by setting $\beta = 0.01$.

In the Gibbs sampling updates, then we need to sample the topic assignment z_{nd} and the topic prior $v_{:d}$. We can do a collapsed sampling as in LDA by calculating the posterior distribution of z_{nd}:

$$p(z_{nd} = k|\pi_{:d}, z_{-nd}, t_{nd}) \propto \pi_{kd}\omega(k, w_{nd})p(t_{nd}|z_{:d}, \theta),$$
$$\omega(k, w_{nd}) = \frac{N^{-nd}_{kw} + \beta}{N^{-nd}_{k} + N\beta}, \quad (2)$$
$$p(t_{nd}|z_{:d}, \theta) = \prod_m \Omega(t_{mnd}|\theta_{z_{md},k})\Omega(t_{nmd}|\theta_{k,z_{md}}),$$

where N is the number of unique word types in dictionary, N^{-nd}_{kw} denotes the number of instances of word w_{nd} assigned with topic k, excluding n-th word in d-th document, and N^{-nd}_{k} denotes the number of total words assigned with topic k. z_{-nd} denotes the topic assignments for all words except z_{nd}.

Note that, in Eq. (2), π_{kd} is the topic prior generated by a joint distribution giving which actions are more likely to co-occur in the video. $\omega(k, w_{nd})$ is the word distribution for topic k giving which topic the word is more likely from. And $p(t_{nd}|z_{:d}, \theta)$ is the time distribution giving which topic-assignment of the word is more causally consistent to other topic-assignments.

Due to the logistic stick-breaking transformation, the posterior distribution of $v_{:d}$ does not have a closed form. So we instead use a Metropolis-Hastings independence sampler [11]. Let the proposals $q(v^*_{:d}|v_{:d}, \mu, \Sigma) = N(v^*_{:d}|\mu, \Sigma)$ be drawn from the prior. The proposal is accepted with probability $\min(\mathbb{A}(v^*_{:d}, v_{:d}), 1)$, where

$$\mathbb{A}(v^*_{:d}, v_{:d}) = \frac{p(v^*_{:d}|\mu, \Sigma)\prod_{n=1}^{M_d} p(z_{nd}|v^*_{:d})q(v_{:d}|v^*_{:d}, \mu, \Sigma)}{p(v_{:d}|\mu, \Sigma)\prod_{n=1}^{M_d} p(z_{nd}|v_{:d})q(v^*_{:d}|v_{:d}, \mu, \Sigma)}$$
$$= \prod_{n=1}^{M_d} \frac{p(z_{nd}|v^*_{:d})}{p(z_{nd}|v_{:d})} = \prod_{k=1}^{K} (\frac{\pi^*_{kd}}{\pi_{kd}})^{\sum_{n=1}^{N_d} \delta(z_{nd},k)},$$

which can be easily calculated by counting the number of words assigned with each topic by z_{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 $O(N^2_d KD)$.

Given a test video, we fix all parameters learned in the training stage and only sample the topic assignments z_{nd} and the topic priors $v_{:d}$.

5.1. Action Segmentation and Recognition

After we learn the topic-assignment of each action-word, we can easily get the action segments by merging the continuous clips with the same assigned topic. Also the assigned topic of the segment indicate which action it is and these segments with the same assigned topic consist an action-topic segment cluster.

5.2. Action Patching

We also apply our model in a new significant application, called *action patching*. It reminds people of forgotten actions by output a segment containing the forgotten action from the training set (illustrated in Fig. 7). It is more challenging than conventional similarity search, since the

[1]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, it also follows Eq. (1) indicating the relative time between two same actions. We also use functions $\tan(-\pi/2 + \pi t)(0 < t < 1), \tan(\pi/2 + \pi t)(-1 < t < 0)$ to feed t to the normal distribution so that the probability is valid, that summits to one through the domain of t.

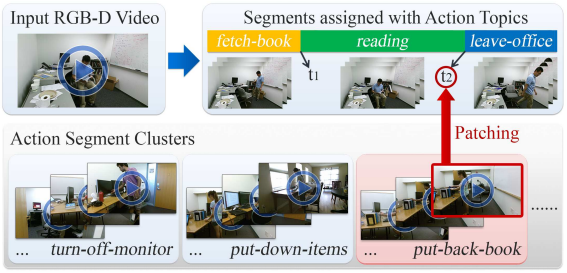


Figure 7: **Illustration of action patching using our model.** Given a test video, we infer the forgotten topic from all missing topics in each segmentation point (t_1, t_2 as above) using the learned co-occurrence and temporal relations of the topics. Then we select the top segment from the inferred action-topic's segment cluster by ranking them using a frame-wise similarity score.

retrieved target is not shown in the query video. Therefore, learning the action co-occurrence and the temporal relations is important in this application.

Different from existing models on action relations learning, our model learns all the pairwise relations rather than only the local and the past-to-future transitions. This is very useful to patching, since those actions occurred with a relatively large time interval with or actions occurred after the forgotten actions are also helpful to detect it, *e.g.*, a *put-back-book* might be forgotten as previously seen a *fetch-book* action before a long *reading*, and seen a *leaving* action indicates he really forgot to *put-back-book*.

Our model infers the forgotten action using the probability inference based on the dependencies. After assigning the topics to the action-words of a query video q, we consider adding one additional action-word $\hat{w}$ into the video in each segmentation point t_s. Then the probabilities of the missing topics k_m in each segmentation point can be compared following the posterior distribution in Eq. (2):

$$p(z_{\hat{w}} = k_m, t_{\hat{w}} = t_s | other) \propto \pi_{k_m d} p(t_s | z_{:d}, \theta) \sum_w \omega(k_m, w),$$

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

where T_s is the set of segmentation points (t_1, t_2 in Fig. 7) and K_e is the set of existing topics in the video (*fetch-book*,*etc.* in Fig. 7). Thus $[1:K] - K_e$ are the missing topics in the video (*turn-off-monitor*,*etc.* in Fig. 7). $p(t_s|z_{:d}, \theta), \omega(k_m, w)$ can be computed as in Eq. (2). Here we marginized $\hat{w}$ to avoid the effect of a specific action-word. Note that, π_{kd} gives the probability of a missing 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 $p(t_s|z_{:d}, \theta)$ measures the casual consistency of adding a new topic.

Then we rank the pair (k_m, t_s) using the above score and select the top ones (three in the experiments). The segments with the selected topics k_m in the training set consist a candidate patching segment set. Finally, we select the top one from the candidates to output by comparing their frame-wise distances. In detail, we consider that the front and the tail of the patching 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 patching segment f_{pm} should be different to f_{qt}, f_{qf}, as it is a different action forgotten in the video.[2] So we choose the patching segment with the maximum score: $ave(\mathbb{D}(f_{pm}, f_{qf}), \mathbb{D}(f_{pm}, f_{qt})) - max(\mathbb{D}(f_{pf}, f_{qt}), \mathbb{D}(f_{pt}, f_{qf}))$, where $\mathbb{D}(,)$ is the average pairwise distances between frames, $ave(,), max(,)$ are the average and max value. If the maximum score is below a threshold or there is no missing topics (*i.e.*, $K_e = [1:K]$) in the query video, we claim there is no forgotten actions.

6. Experiments

6.1. Dataset

We collect a new challenging RGB-D activity dataset recorded by the new Kinect v2 camera [3]. Each video in the dataset contains 2-7 actions interacted with different objects (see examples in Fig. 8). The new Kinect v2 has higher resolution of RGB-D frames (RGB: 1920×1080, depth: 512×424) and improved body tracking of human skeletons (25 body joints). We record 458 videos with a total length of about 230 minutes. We ask 7 subjects to perform human daily activities in 8 offices and 5 kitchens with complex backgrounds. And in each environment the activities are recorded in different views. It composed of fully annotated 21 types of actions (10 in the office, 11 in the kitchen) interacted with 23 types of objects. We also record the audio, though it is not used in this paper.

In order to get a variation in activities, we ask participants to finish task with different combinations of actions and ordering naturally. Some actions occur together often such as *fetch-from-fridge* and *put-back-to-fridge* while some are not always in the same video (see more examples on our website). Some actions are in fix ordering such as *fetch-book* and *put-back-book* while some occur in random order. Moreover, to evaluate the action patching performance, 222 videos in the dataset has action forgotten by people naturally and the forgotten actions are annotated.

6.2. 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). We compare seven unsupervised approaches in our experiments. They are Hidden

[2] 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.

[3] The dataset and tools are released at `http://watchnpatch.cs.cornell.edu`

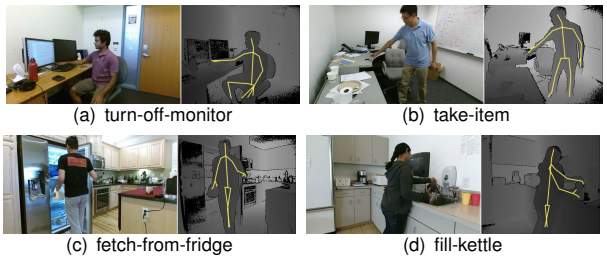


Figure 8: Action examples in our dataset. The left is RGB frame and the right is depth frame with human skeleton (yellow). The full action classes are shown on our website.

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 (CaTM), that is the correlated topic model over relative time. All these methods use the same human skeleton and RGB-D features introduced in Section 3.1. We also evaluate HMM and our model CaTM using the popular features for action recognition, dense trajectories feature (DTF) [38], extracted only in RGB videos[4], named as HMM-DTF and CaTM-DTF.

In the experiments, we set the number of topics (states of HMM) equal to or more than ground-truth action 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 ranking score as in Section 5.2 to select the top one patched segments.

6.3. Evaluation Metrics

We want to evaluate if the unsupervised learned action-topics (states for HMM) are semantically meaningful. We first map the assigned topics to the ground-truth labels for evaluation. This could be done by counting the mapped frames 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: $m_{kc} = \frac{\sum_i \delta(k_i,k)\delta(c_i,c)}{\sum_i \delta(c_i,c)}$, 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:

$$\max_x \sum_{k,c} x_{kc} m_{kc},$$

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

where $x_{kc} = 1$ indicates mapping topic k to class c, otherwise $x_{kc} = 0$. And $\sum_c x_{kc} = 1$ constrain that each topic must be mapped to exact one class, $\sum_k x_{kc} \geq 1$ 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. Segmentation: we consider a true positive if the overlap (union/intersection) between the detected and the ground-truth segments is more than a default threshold 40% as in [27]. Then we compute *segmentation accuracy (Seg-Acc)*, the ratio of the ground-truth segments that are correctly detected, and *segmentation average precision (Seg-AP)* by sorting all action segments output by the approach using the average probability of their words' topic assignments. All above three metrics are computed by taking the average of each action class.

We also evaluate the *patching accuracy (P-Acc)* by the portion of correct patched video, including correctly output the forgotten action segments or correctly claiming no forgotten actions. We consider the output action segments by the algorithm containing over 50% ground-truth forgotten actions as correctly output the forgotten action segments.

6.4. Results

Table 2 and Fig. 9 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 recognition, 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

[4]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.

Table 2: Results using the same number of topics as the ground-truth action classes. HMM-DTF, CaTM-DTF use DTF RGB features and others use our human skeleton and RGB-D features.

'office' (%)	Seg-Acc		Seg-AP		Frame-Acc		P-Acc
	Offline	Online	Offline	Online	Offline	Online	
HMM-DTF	15.2	9.4	21.4	20.7	20.2	15.9	23.6
HMM	18.0	14.0	25.9	24.8	24.7	21.3	33.3
TM	9.3	9.2	20.9	19.6	20.3	13.0	13.3
CTM	10.0	5.9	18.1	15.8	20.2	16.4	13.3
TM-AT	8.9	3.7	25.4	19.0	18.6	13.8	12.0
CTM-AT	9.6	6.8	25.3	19.8	19.6	15.5	10.8
TM-RT	**30.8**	30.9	29.0	30.2	38.1	36.4	39.5
CaTM-DTF	28.2	27.0	28.3	27.4	37.4	34.0	33.7
CaTM	30.6	**32.9**	**33.1**	**34.6**	**39.9**	**38.5**	**41.5**

'kitchen' (%)	Seg-Acc		Seg-AP		Frame-Acc		P-Acc
	Offline	Online	Offline	Online	Offline	Online	
HMM-DTF	4.9	3.6	18.8	5.6	12.3	9.8	2.3
HMM	20.3	15.2	20.7	13.8	21.0	18.3	7.4
TM	7.9	4.7	21.5	14.7	20.9	11.5	9.6
CTM	10.5	9.2	20.5	14.9	18.9	15.7	6.4
TM-AT	8.0	4.8	21.5	21.6	20.9	14.0	7.4
CTM-AT	9.7	10.0	19.1	22.6	20.1	16.7	10.7
TM-RT	32.3	26.9	23.4	23.0	35.0	31.2	18.3
CaTM-DTF	26.9	23.6	18.4	17.4	33.3	29.9	16.5
CaTM	**33.2**	**29.0**	**26.4**	**25.5**	**37.5**	**34.0**	**20.5**

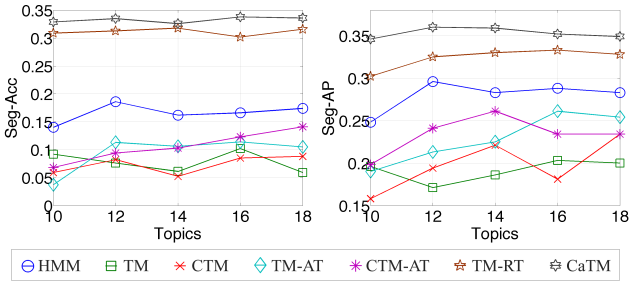


Figure 9: Online segmentation Acc/AP varied with the number of topics in 'office' dataset.

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 can 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. 10. 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. 9. 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-DTF and

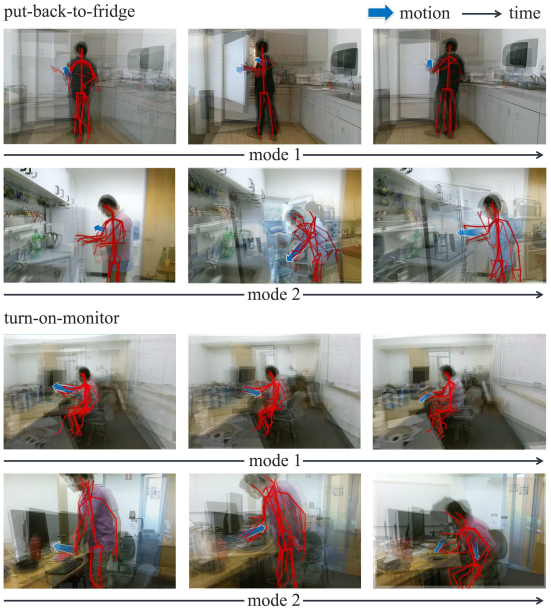


Figure 10: **Visualization of the learned topics using our model.** For better illustration, we decompose the segments with the same topic into different modes (shown two) and divide a segment into three stages in time. The clips from different segments in the same stage are merged by scaling to the similar size of human skeletons.

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 performs? 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.

6.5. Sharing the Learned Topics

In order to make our learned knowledge useful to robots, we also share the learned topics to RoboBrain [30], a large-scale knowledge engine for robots. Our learned action topics are represented as nodes in the knowledge graph for robots and these nodes are connected with edges of our learned co-occurrence and temporal relations.

7. Conclusion

In this paper, we presented an algorithm that models the human activities in a completely unsupervised setting. We showed that it is important to model the long-range relations between the actions. To achieve this, we considered the video as a sequence of action-words, and an activity as a set of action-topics. Then we modeled the word-topic distributions, the topic correlations and the topic relative time distributions. We then showed the effectiveness of our model in the unsupervised action segmentation and recognition, as well as the action patching. For evaluation, we also contributed a new challenging RGB-D activity video dataset.

Acknowledgement

We acknowledge the support of onr sparse(N00014-13-1-0761) and ONR award N000141110389.

References

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

[2] S. M. Assari, A. R. Zamir, and M. Shah. Video classification using semantic concept co-occurrences. In *CVPR*, 2014.

[3] S. Bhattacharya, M. M. Kalayeh, R. Sukthankar, and M. Shah. Recognition of complex events: Exploiting temporal dynamics between underlying concepts. In *CVPR*, 2014.

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

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

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[7] 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 *ECCV*, 2014.

[8] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013.

[9] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In *ECCV*, 2009.

[10] T. A. Faruquie, P. K. Kalra, and S. Banerjee. Time based activity inference using latent dirichlet allocation. In *BMVC*, 2009.

[11] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian data analysis*. CRC press, 2013.

[12] M. Hoai, Z. zhong Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. In *CVPR*, 2011.

[13] S. Jones and L. Shao. Unsupervised spectral dual assignment clustering of human actions in context. In *CVPR*, 2014.

[14] V. Kantorov and I. Laptev. Efficient feature extraction, encoding and classification for action recognition. In *CVPR*, 2014.

[15] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *ECCV*, 2007.

[16] D. I. Kim and E. B. Sudderth. The doubly correlated nonparametric topic model. In *NIPS*, 2011.

[17] A. Kläser, M. Marszałek, 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.

[18] H. S. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. In *RSS*, 2013.

[19] H. S. Koppula and A. Saxena. Learning spatio-temporal structure from RGB-D videos for human activity detection and anticipation. In *ICML*, 2013.

[20] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, 2014.

[21] I. Laptev and P. Perez. Retrieving actions in movies. In *ICCV*, 2007.

[22] Y.-Y. Lin, J.-H. Hua, N. C. Tang, M.-H. Chen, and H.-Y. Mark Liao. Depth and skeleton associated action recognition without online accessible rgb-d cameras. In *CVPR*, 2014.

[23] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[24] S. Mathe and C. Sminchisescu. Actions in the Eye: Dynamic Gaze Datasets and Learnt Saliency Models for Visual Recognition. *TPAMI*, 2014.

[25] B. Ni, V. R. Paramathayalan, and P. Moulin. Multiple granularity analysis for fine-grained action detection. In *CVPR*, 2014.

[26] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010.

[27] H. Pirsiavash and D. Ramanan. Parsing videos of actions with segmental grammars. In *CVPR*, 2014.

[28] N. Reljin, S. McDaniel, D. Pokrajac, N. Pejcic, T. Vance, A. Lazarevic, and L. J. Latecki. Small moving targets detection using outlier detection algorithms. *Proc. SPIE*, 7698:769804–769804–12, 2010.

[29] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012.

[30] A. Saxena, A. Jain, O. Sener, A. Jami, D. K. Misra, and H. S. Koppula. Robo brain: Large-scale knowledge engine for robots. *Tech Report arxiv*, 2014.

[31] B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, 2012.

[32] Q. Shi, L. Cheng, L. Wang, and A. Smola. Human action segmentation and recognition using discriminative semi-markov models. *IJCV*, 93(1):22–32, 2011.

[33] J. Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *ICRA*, 2012.

[34] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

[35] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *CVPR*, 2013.

[36] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *CVPR*, 2014.

[37] N. N. Vo and A. F. Bobick. From stochastic grammar to bayes network: Probabilistic parsing of complex activity. In *CVPR*, 2014.

[38] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action Recognition by Dense Trajectories. In *CVPR*, 2011.

[39] X. Wang and Q. Ji. A hierarchical context model for event recognition in surveillance video. In *CVPR*, 2014.

[40] X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *KDD*, 2006.

[41] C. Wu, I. Lenz, and A. Saxena. Hierarchical semantic labeling for task-relevant rgb-d perception. In *RSS*, 2014.

[42] D. Wu and L. Shao. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In *CVPR*, 2014.

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