Learning Spatio-Temporal Structure from RGB-D Videos for Human Activity Detection and Anticipation

Hema S. Koppula HEMA@CS.CORNELL.EDU
Ashutosh Saxena ASAXENA@CS.CORNELL.EDU
Computer Science Department, Cornell University, Ithaca, NY 14853 USA

Abstract

We consider the problem of detecting past activities as well as anticipating which activity will happen in the future and how. We start by modeling the rich spatio-temporal relations between human poses and objects (called affordances) using a conditional random field (CRF). However, because of the ambiguity in the temporal segmentation of the sub-activities that constitute an activity, in the past as well as in the future, multiple graph structures are possible. In this paper, we reason about these alternate possibilities by reasoning over multiple possible graph structures. We obtain them by approximating the graph with only additive features, which lends to efficient dynamic programming. Starting with this proposal graph structure, we then design moves to obtain several other likely graph structures. We then show that our approach improves the state-of-the-art significantly for detecting past activities as well as for anticipating future activities, on a dataset of 120 activity videos collected from four subjects.

1. Introduction

Being able to detect which activity is being performed as well as being able to *anticipate* what is going to happen next in an environment is important for application domains such as robotics and surveillance. In a typical environment, we have humans interacting with the objects and performing a sequence of activities. Recently, inexpensive RGB-D cameras (such as Microsoft Kinect, see Figure 1) have enabled researchers to model such rich spatio-temporal interactions in the 3D scene for learning complex human activities. For example, Koppula, Gupta and Saxena (2013) (KGS) used a conditional random field (CRF), trained with max-margin methods, to model the rich spatio-temporal relations between the objects and humans in the scene.

However, in previous works, emphasis has been on modeling the relations between components in the scene (human pose, objects, etc.), and performing learning and inference *given* the spatio-temporal structure of the model (i.e., for a given CRF structure in the case of KGS). However, it is quite challenging to estimate this structure because of two reasons. First, an activity comprises several sub-activities, of varying temporal length, performed in a sequence. This results in an ambiguity in the temporal segmentation and thus a single graph structure may not explain the activity well. Second, there can be several possible graph structures when we are reasoning about activities in the future (i.e., when the goal is to *anticipate* future activities, different from just detecting the past activities). Multiple spatio-temporal graphs are possible in these cases and we need to reason over them in our learning algorithm.

In our work, we start by using a CRF to model the sub-activities and affordances of the objects, how they change over time, and how they relate to each other. In detail, we have two kinds of nodes: object and sub-activity nodes. The edges in the graph model the pairwise relations among interacting nodes, namely the object–object interactions, object–sub-activity interactions, and the temporal interactions (see Figure 1). This model is built with each spatio-temporal segment being a node. Figure 2 shows two possible graph structures for an activity with two objects. We then reason about the possible graph structures for both past and future activities. The key idea is to first sample a few segmentations that are close to the ground-truth segmentation using our CRF model instantiated with a

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. Copyright 2013 by the author(s).

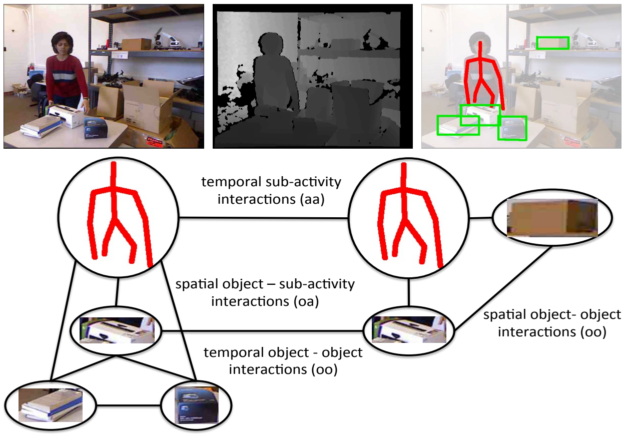
temporal sub-activity interactions (aa)
spatial object – sub-activity interactions (oa)
spatial object- object interactions (oo)
temporal object - object interactions (oo)

Figure 1. An example activity from the CAD-120 dataset (top row) and one possible graph structure (bottom row). Top row shows the RGB image (left), depths (middle), and the extracted skeleton and object information (right). (Graph in the bottom row shows the nodes at only the temporal segment level, the frame level nodes are not shown.)

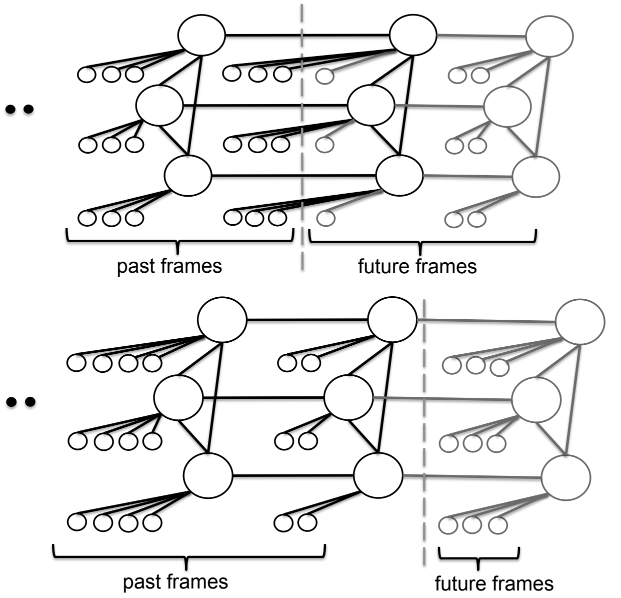

Figure 2. Figure illustrating two possible graph structures (top and bottom), with six observed frames in the past and three anticipated frames in the future. This example has one sub-activity node and two object nodes in each temporal segment.

subset of features, and then explore the space of segmentation by making merge and split moves to create new segmentations. We do so by approximating the graph with only additive features, which lends to efficient dynamic programming.

In extensive experiments over 120 activity videos collected from four subjects, we showed that our approach outperforms the state-of-the-art results both in the tasks of activity and affordance detection. We achieved an accuracy of 85.4% for affordance, 70.3% for sub-activity labeling and 83.1% for high-level activities respectively for detection. Furthermore, we obtain an accuracy of 67.2% and 49.6% for anticipating affordances and sub-activities respectively in future time-frames.

2. Related Work

There has been a considerable amount of previous work on detection of human activities from still images as well as videos (e.g., Maji et al., 2011; Yang et al., 2010; Xing et al., 2008; Ryoo, 2011; Hoai & De la Torre, 2012a). Similar to our setting some recent works have shown that modeling the mutual context between human poses and objects (either the category label or affordance label, Jiang et al. 2012a) is useful for activity detection (Gupta et al., 2009; Yao & Fei-Fei, 2010; Delaitre et al., 2011; Prest et al., 2012; Koppula et al., 2013).

Recent availability of inexpensive RGB-D sensors has enabled significant improvement in scene modeling (Koppula et al., 2011; Anand et al., 2012; Jiang et al., 2012b; 2013; Jia et al., 2013; Jiang & Saxena, 2013) and estimation of human poses (Shotton et al., 2012; Ly et al., 2012). This, together with depth information, has enabled some recent works (Sung et al., 2011; Zhang & Parker, 2011; Ni et al., 2011; Sung et al., 2012) to obtain good action recognition performance. However, these methods only address detection over small periods of time, where temporal segmentation (and thus knowledge of the spatio-temporal graph structure) is not a big problem. KGS (Koppula et al., 2013) proposed a model to jointly predict sub-activities and object affordances by taking into account both spatio-temporal interactions between human poses and objects over longer time periods. However, KGS found that not knowing the graph structure (i.e., the correct temporal segmentation) decreased the performance significantly. This is because the boundary between two sub-activities is often not very clear, as people often start performing the next sub-activity before finishing the current sub-activity. We compare our proposed method with theirs and show considerable improvement over their state-of-the-art results.

In activity detection from 2D videos, much previous work has focussed on short video clips, assuming that temporal segmentation has been done apriori. Some recent effort in recognizing actions from longer video sequences take an event detection approach (Ke et al., 2007; Simon et al., 2010; Nguyen et al., 2009), where they evaluate a classifier function at many different segments of the video and then predict the event presence in segments. Similarly, change point detection methods (Xuan & Murphy, 2007; Harchaoui et al.,

2008) work by performing a sequence of change-point analysis in a sliding window along the time dimension. However, these methods only detect *local* boundaries and tend to over-segment complex actions which often contain many changes in local motion statistics.

Some previous works consider joint segmentation and recognition by defining dynamical models based on kinematics (Oh et al., 2008; Fox et al., 2009), but these works do not model the complex human-object interactions. Hoai et al. (2011) and Hoai & De la Torre (2012b) do not consider temporal context and only perform activity classification and clustering respectively. In related work, Ion et al. (2011) consider the problem 2D image segmentation. They sample segmentations of images for labeling using an Incremental Saddle Point estimation procedure which require good initial samples. In contrast, our application requires modeling of the temporal context (as compared to just spatial). This work is closer to KGS, where they also sample the segmentation space. However, in our approach we use a discriminative approach, where we model the energy function as composed of an additive and a non-additive term. This allows us to efficiently sample the potential graph structures.

One important application of our approach is in anticipating future activities, where reasoning over future possible graph structures becomes important. Anticipating future activities has gained attention only recently (Kitani et al., 2012; Koppula & Saxena, 2013). Kitani et al. (2012) proposed a Markov decision process to obtain a distribution over possible human navigation trajectories in 2D from visual data. Koppula & Saxena (2013) addressed the problem of anticipating human activities at a fine-grained level of how humans interact with objects in more complex activities such as *microwaving food* or *taking medicine*. They represent the distribution of the possible futures with a set of particles that are obtained by augmenting the CRF structure of KGS with sampled future nodes. However, they do not reason about the possible graph structures for the past. In our work, we show that sampling the spatio-temporal structure, in addition to sampling the future nodes, results in better anticipation.

In terms of learning algorithms, probabilistic graphical models are a workhorse of machine learning and have been applied to a variety of applications. Frameworks such as HMMs (Hongeng & Nevatia, 2003; Natarajan & Nevatia, 2007), DBNs (Gong & Xiang, 2003), CRFs (Quattoni et al., 2007; Sminchisescu et al., 2005; Koppula et al., 2013), and semi-CRFs (Sarawagi & Cohen, 2004) have been previously used to model the temporal structure of videos and text. While most previous works maintain their template graph structure over time, in our work, new graph structures are possible. Works on semi-Markov models (Sarawagi & Cohen, 2004; Shi et al., 2011) are quite related to our work as they address the problem of finding the segmentation along with labeling. However, these methods are limited since they are only efficient for feature maps that are additive in nature. We build upon these ideas where we use the additive feature map as only a close approximation to the graph structure and then explore the space of likely graph structure by designing moves. We show that this improves performance while being computationally efficient.

3. Modeling Spatio-Temporal Relations for a Given Graph Structure

In our setting, our algorithm observes a scene containing a human and objects for time t in the past, and our goal is to detect activities in the observed past and also anticipate the future activities for time d. Following KGS, we discretize time to the frames of the video[1] and group the frames into temporal segments, where each temporal segment spans a set of contiguous frames corresponding to a single sub-activity. Therefore, at time 't' we have observed 't' frames of the activity that are grouped into 'k' temporal segments. (Figure 2 shows two temporal segments for the past.)

We model the spatio-temporal structure of an activity using a conditional random field, illustrated in Figure 1. For the past t frames, we know the nodes of the CRF but we do not know the temporal segmentation, i.e., which frame level nodes are connected to each of the segment level node. The node labels are also unknown. For the future d frames, we do not even know the nodes in the graph—there maybe different number of objects being interacted with depending on which sub-activity is performed in the future. Our goal is to explore different possible past and future graph structures (i.e., sub-activities, human poses, object locations and affordances). We will do so by augmenting the graph in time with potential object nodes, and sampling several possible graph structures.

We first describe the CRF modeling *given* a fixed graph structure for t observed frames. Let, $s \in \{1, .., N\}$ denote a temporal segment and $\mathbf{y} = (\mathbf{y}_1, ..., \mathbf{y}_N)$ denote the labeling we are interested in finding, where $\mathbf{y}_s$ is the set of sub-activity and object affordance labels for the temporal segment s. Our input is a set of features $\Phi(\mathbf{x})$ extracted from the seg-

[1] In the following, we will use the number of videos frames as a unit of time, where 1 unit of time ≈ 71ms (=1/14, for a frame-rate of about 14Hz).

mented 3D video. The prediction $\hat{\mathbf{y}}$ is computed as the argmax of an energy function $E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w})$ that is parameterized by weights $\mathbf{w}$.

$$\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y}} E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w}) \tag{1}$$

This energy is expressed over a graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ (as illustrated in Figure 1), and consists of two terms—node terms and edge terms. Each of these terms comprises the label, appropriate features, and weights. Let y_i^k be a binary variable representing the node i having label k, where K is the set of labels. Let $\phi_n(i)$ and $\phi_e(i,j)$ be the node and edge feature maps respectively. (Depending on the node and edge being used, the appropriate subset of features and class labels are used.) We thus write the energy function as:

$$\begin{aligned} E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w}) = & \sum_{i\in\mathcal{V}} \sum_{k\in K} y_i^k \left[w_n^k \cdot \phi_n(i) \right], \\ & + \sum_{(i,j)\in\mathcal{E}} \sum_{(l,k)\in K\times K} y_i^l y_j^k \left[w_e^{lk} \cdot \phi_e(i,j) \right] \end{aligned} \tag{2}$$

Note that the graph has two types of nodes: sub-activity and object nodes, and it has four types of edges: object–object edges, object–sub-activity edges, object–object temporal edges, and sub-activity–sub-activity temporal edges. Our energy function is a sum of six types of potentials that define the energy of a particular assignment of sub-activity and object affordance labels to the sequence of segments in the given video: $E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w}) = E_o + E_a + E_{oo} + E_{oa} + E_{oo}^t + E_{aa}^t$. However, we have written it compactly in Eq. (2).

Learning and Inference. As the structure of the graph is fully known, we learn the parameters of the energy function in Eq. (2) by using the cutting plane method (Joachims et al., 2009).

However, during inference we only know the nodes in the graph but not the temporal segmentation, i.e., the structure of the graph in terms of the edges connecting frame level nodes to the segment level label nodes. We could search for the best labeling over *all* possible segmentations, but this is very intractable because our feature maps contain non-additive features (that are important and are described in the next sub-section).

3.1. Features: Additive and Non-Additive

We categorize the features into two sets: additive features, $\Phi^A(\mathbf{x})$, and non-additive features, $\Phi^{NA}(\mathbf{x})$. We compute the additive features for a set of frames corresponding to a temporal segment by adding the feature values for the frames belonging to the temporal segment. Examples of the additive features include distance moved and vertical displacement of an object within a temporal segment. The features that do not satisfy this property are referred to as the non-additive features, for example, maximum and minimum distances between two objects. As we discuss in the next section, additive features allow efficient joint segmentation and labeling by using dynamic programming, but may not be expressive enough.

Non-additive features sometimes provide very useful cues for discriminating the sub-activity and affordance classes. For example, consider discriminating *cleaning* sub-activity from a *moving* sub-activity: here the total distance moved could be similar (an additive feature), however, the minimum and maximum distance moved being small may be strong indicator of the activity being *cleaning*. In fact, when compared to our model learned using only the additive features, the model learned with both additive and non-additive features improves macro precision and recall by 5% and 10.1% for labeling object affordance respectively and by 3.7% and 6.2% for labeling sub-activities respectively.

In detail, we use the same features as described by KGS. These features include the node feature maps $\phi_o(i)$ and $\phi_a(j)$ for object node i and sub-activity node j respectively, and edge feature maps $\phi_t(i,j)$ capturing the relations between various nodes. The object node feature map, $\phi_o(i)$, includes the (x, y, z) coordinates of the object's centroid, the coordinates of the object's bounding box and transformation matrix w.r.t. to the previous frame computed at the middle frame of the temporal segment, the total displacement and the total distance moved by the object's centroid in the set of frames belonging to the temporal segment. The sub-activity node feature map, $\phi_a(j)$, gives a vector of features computed using the noisy human skeleton poses obtained from running Openni's skeleton tracker on the RGBD video. We compute the above described location (relative to the subject's head location) and distance features for each the upper-skeleton joints excluding the elbow joints (neck, torso, left shoulder, left palm, right shoulder and right palm).

The edge feature maps, $\phi_t(i,j)$, include relative geometric features such as the difference in (x, y, z) coordinates of the object centroids and skeleton joint locations and the distance between them. In addition to computing these values at the first, middle and last frames of the temporal segment, we also consider the *min* and *max* of their values across all frames in the temporal segment to capture the relative motion information. The temporal relational features capture the change across temporal segments and we use the vertical change in position and the distance between corresponding object and joint locations. We perform cumulative binning of all the feature values into 10 bins for each feature.

4. Sampling Spatio-Temporal Graphs

Efficient Inference with Additive Features. We express the feature set, $\Phi(\mathbf{x})$, as the concatenation of the additive and non-additive feature sets, $\Phi^A(\mathbf{x})$ and $\Phi^{NA}(\mathbf{x})$ respectively. Therefore, by rearranging the terms in Eq. (2), the energy function can written as:

$$E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w}) = E(\mathbf{y}|\Phi^A(\mathbf{x});\mathbf{w}) + E(\mathbf{y}|\Phi^{NA}(\mathbf{x});\mathbf{w})$$

We perform efficient inference for the energy term $E(\mathbf{y}|\Phi^A(\mathbf{x});\mathbf{w})$ by formulating it as a dynamic program (see Eq. (3)). In detail, let L denote the max length of a temporal segment, i denote the frame index, s denote the temporal segment spanning frames $(i-l)$ to i, and $(s-1)$ denote the previous segment. We write the energy function in a recursive form as:

$$\begin{aligned} V(i,k) = \max_{k',l=1\ldots L} V(i-l,k') + \sum_{k\in K} y_s^k \left[w_n^k \cdot \phi_n^A(s) \right] \\ + \sum_{k\in K} y_s^k \left[w_e^{lk} \cdot \phi_e^A(s-1,s) \right] \end{aligned} \tag{3}$$

Here, $\phi_n^A(s)$ and $\phi_e^A(s-1,s)$ denote the additive feature maps and can be efficiently computed by using the concept of integral images.[2] The best segmentation then corresponds to the path traced by $\max_a V(t,a)$, where t is the number of video frames.

Using $E(\mathbf{y}|\Phi^A(\mathbf{x});\mathbf{w})$, we find the top-k scored segmentations and then evaluate them using the full model $E(\mathbf{y}|\Phi(\mathbf{x});\mathbf{w})$ in order to obtain more accurate labelings.

Merge and Split Moves. The segmentations generated by the approximate energy function, $E(\mathbf{y}|\Phi^A(\mathbf{x});\mathbf{w})$, are often very close to the given ground-truth segmentations. However, since the energy function used is only approximate, it sometimes tends to over-segment or miss the boundary by a few frames. In order to obtain a representative set of segmentation samples, we also perform random merge and split moves over these segmentations, and consider them for evaluating with the full model as well. A merge move randomly selects a boundary and removes it, and a split move randomly chooses a frame in a segment and creates a boundary.

Heuristic Segmentations. There is a lot of information present in the video which can be utilized for the purpose of temporal segmentation. For example, smooth movement of the skeleton joints usually represent a single sub-activity and the sudden changes in the direction or speed of motion indicate sub-activity boundaries. Therefore, we incorporate such information in performing temporal segmentation of the activities. In detail, we use the multiple segmentation hypotheses proposed by KGS. These include graph based segmentation method proposed by (Felzenszwalb & Huttenlocher, 2004) adapted to temporally segment the videos. The sum of the Euclidean distances between the skeleton joints and the rate of change of the Euclidean distance are used as the edge weights for two heuristic segmentations respectively. By varying the thresholds, different temporal segmentations of the given activity can be obtained. In addition to the graph based segmentation methods, we also use the uniform segmentation method which considers a set of continuous frames of fixed size as the temporal segment. There are two parameters for this method: the segment size and the offset (the size of the first segment). However, these methods often over-segment a sub-activity, and each segmentation would result in a different graph structure for our CRF modeling.

We generate multiple graph structures for various values of the parameters for the above mentioned methods and obtain the predicted labels for each using Eq. (1). We obtain the final labeling over the segments by either using the second-step learning method presented in KGS, or by performing voting and taking the label predicted by majority of the sampled graph structures (our experiments follow the latter).

4.1. Anticipating and Sampling Future Nodes

For anticipating the next d time, we augment the CRF structure with d frame nodes as proposed in (Koppula & Saxena, 2013). Sampling this graph comprises three steps. First, we need to sample the possible sub-activity and the object nodes involved. Second, for this segment-level graph, we sample the frame-level nodes, starting with possible end-points of the physical location of the objects and human hands. Third, given the end locations, we sample a variety of trajectories. This sampling procedure gives us various possible future nodes and their spatial relations. We add these to the observed nodes and sample various possible temporal segmentations as described above.

In the first step, we sample the object affordances and activities based on a discrete distribution generated from the training data. This distribution is based on

[2] The additive features for temporal segments starting at the first frame and ending at frame l, for $l = 1..t$ are pre-computed, i.e., the segment features for a total of t temporal segments are computed. This needs $(n \times t)$ summations, where n is the number of features. Now the segment features for a temporal temporal segment starting and ending at any frame can be computed by n subtractions. Therefore, the total feature computation cost is linear in the number of possible segmentations.

the object type (e.g., cup, bowl, etc.) and object's current position with respect to the human in the scene (e.g., in contact with the human hand, etc.). For example, if a human is holding an object of type 'cup' placed on the table, then the affordances *drinkable* and *movable* with their corresponding sub-activities (*drinking* and *moving* respectively) have equal probability, with all others being 0.

In the second step, given the sampled affordances and sub-activity, we sample the corresponding object locations and human poses for the d anticipated frames. In order to have meaningful object locations and human poses we take the following approach. We model object affordances using a potential function based on how the object is being interacted with, when the corresponding affordance is active. The affordance potential has the form $\psi_o = \prod_i \psi_{dist_i} \prod_j \psi_{ori_j}$, where ψ_{dist_i} is the i^{th} distance potential and ψ_{ori_j} is the j^{th} relative angular potential. We model each distance potential with a Gaussian distribution and each relative angular potential with a von Mises distribution and find the parameters from the training data. We can now score the points in the 3D space using the potential function, whose value represents the strength of the particular affordance at that location. Therefore, for the chosen affordance, we sample the object's most likely future location using the affordance potentials.

In the third step, for every sampled target object location, we generate a set possible trajectories following which the object can be moved form its current location to the sampled target location. We use parametrized cubic equations, in particular Bézier curves, to generate human hand like motions (Faraway et al., 2007). We estimate the control points of the Bézier curves from the training data.

Sampling Augmented Graphs. In order to sample the future graphs, we augment our observed frames with these sampled future frames, and then sample the augmented graph structures (for different temporal segmentations) as described in Section 4. We then use the energy function in Eq. (2) to obtain the best labeling for each sampled graph. The node potentials score how well the features extracted from the anticipated object locations and human poses match the affordance and sub-activity labels respectively, and the edge potentials score how likely are the anticipated sub-activities and affordances likely to follow the observed ones. Therefore, the value of the energy function provides a ranking over the sampled augmented graphs for the future time. For obtaining anticipation metrics, we report the highest ranked one (and compare it with what was actually performed in the future).

5. Experiments

Data. We test our model on the Cornell Activity Dataset-120 (CAD-120) (Koppula et al., 2013). It contains 120 3D videos of four different subjects performing 10 high-level activities, where each high-level activity was performed three times with *different* objects. It contains a total of 61,585 total 3D video frames. The activities have a long sequence of sub-activities, which vary from subject to subject significantly in terms of length of the sub-activities, order of the sub-activities as well as in the way they executed the task. The dataset is labeled with both sub-activity and object affordance labels. The high-level activities are: {*making cereal*, *taking medicine*, *stacking objects*, *unstacking objects*, *microwaving food*, *picking objects*, *cleaning objects*, *taking food*, *arranging objects*, *having a meal*}. The sub-activity labels are: {*reaching*, *moving*, *pouring*, *eating*, *drinking*, *opening*, *placing*, *closing*, *scrubbing*, *null*} and affordance labels are: {*reachable*, *movable*, *pourable*, *pourto*, *containable*, *drinkable*, *openable*, *placeable*, *closable*, *scrubbable*, *scrubber*, *stationary*}.

Evaluation: For comparison, we follow the same train-test split described in KGS and train our model on activities performed by three subjects and test on activities of a *new subject*. We report the results obtained by 4-fold cross validation by averaging across the folds. We consider the overall micro accuracy (P/R), macro precision and macro recall of the detected sub-activities, affordances and overall activity. Micro accuracy is the percentage of correctly classified labels. Macro precision and recall are the averages of precision and recall respectively for all classes. We also computed these metrics for the anticipated sub-activity and affordances.

Detection Results. Table 1 shows the performance of our proposed approach on object affordance, sub-activity and high-level activity labeling for past activities. Rows 3-5 show the performance for the case where ground-truth temporal segmentation is provided and rows 6-9 show the performance for the different methods when no temporal segmentation is provided. With known graph structure, the model using the the full set of features (row 4) outperforms the model which uses only the additive features (row 5): macro precision and recall improve by 5% and 10.1% for labeling object affordance respectively and by 3.7% and 6.2% for labeling sub-activities respectively. This shows that additive features bring us close, but not quite, to the optimal graph structure.

When the graph structure is not known, the performance drops significantly. Our graph sampling approach based on the additive energy function (row 6)

Table 1. **Results on CAD-120 dataset for *detection***, showing average micro precision/recall, and average macro precision and recall for affordances, sub-activities and high-level activities. Computed from 4-fold cross validation with testing on a new human subject in each fold. Standard error is also reported.

With ground-truth segmentation.

	Object Affordance			Sub-activity			High-level Activity		
	micro	macro		micro	macro		micro	macro	
method	P/R	Prec.	Recall	P/R	Prec.	Recall	P/R	Prec.	Recall
chance	8.3 (0.0)	8.3 (0.0)	8.3 (0.0)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)
max class	65.7 (1.0)	65.7 (1.0)	8.3 (0.0)	29.2 (0.2)	29.2 (0.2)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)	10.0 (0.0)
KGS (Koppula et al., 2013)	91.8 (0.4)	**90.4** (2.5)	74.2 (3.1)	86.0 (0.9)	84.2 (1.3)	76.9 (2.6)	84.7 (2.4)	85.3 (2.0)	84.2 (2.5)
Our model: all features	**93.9** (0.4)	89.2 (1.3)	**82.5** (2.0)	**89.3** (0.9)	**87.9** (1.8)	**84.9** (1.5)	**93.5** (3.0)	**95.0** (2.3)	**93.3** (3.1)
Our model: only additive features	92.0 (0.5)	84.2 (2.2)	72.4 (1.2)	86.5 (0.6)	84.2 (1.3)	78.7 (1.9)	90.3 (3.8)	92.8 (2.7)	90.0 (3.9)

Without ground-truth segmentation.

Our DP seg.	83.6 (1.1)	70.5 (2.3)	53.6 (4.0)	**71.5** (1.4)	71.0 (3.2)	60.1 (3.7)	80.6 (4.1)	86.1 (2.5)	80.0 (4.2)
Our DP seg. + moves	84.2 (0.9)	72.6 (2.3)	58.4 (5.3)	71.2 (1.1)	70.6 (3.7)	61.5 (4.5)	83.1 (5.2)	**88.0** (3.4)	**82.5** (5.4)
heuristic seg. (KGS)	83.9 (1.5)	75.9 (4.6)	64.2 (4.0)	68.2 (0.3)	71.1 (1.9)	62.2 (4.1)	80.6 (1.1)	81.8 (2.2)	80.0 (1.2)
Our DP seg. + moves + heuristic seg.	**85.4** (0.7)	**77.0** (2.9)	**67.4** (3.3)	70.3 (0.6)	**74.8** (1.6)	**66.2** (3.4)	83.1 (3.0)	87.0 (3.6)	82.7 (3.1)

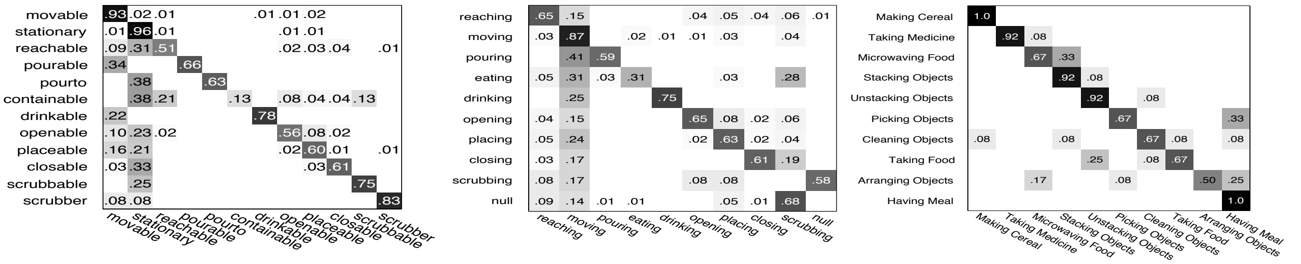

Figure 3. **Confusion matrix** for affordance labeling (left), sub-activity labeling (middle) and high-level activity labeling (right) of the test RGB-D videos.

achieves 83.6% and 71.5% micro precision for labeling object affordance and sub-activities, respectively. This is improved by sampling additional graph structures based on the Split and Merge moves (row 7). Finally, when combining these segmentations with the other heuristically generated segmentations presented by KGS, our method obtains the best performance (row 9) and significantly improves the previous state-of-the-art (KGS, row 8).

Figure 3 shows the confusion matrix for labeling affordances, sub-activities and high-level activities using our method (row 9). We can see that there is a strong diagonal with a few errors such as *pouring* misclassified as *moving*, and *picking objects* misclassified as *having a meal*. Figure 4 shows the labeling output of the different methods. The bottom-most row show the ground-truth segmentation, top-most row is the labeling obtained when the graph structure is provided, followed by three heuristically generated segmentations. The fifth row is the segmentation generated by our sampling approach and the sixth and seventh rows are the labeling obtained by combining the multiple segmentations using a simple max-voting and by the multi-segmentation learning of KGS. Note that some sub-activity boundaries are more ambiguous (high variance among different methods) than the others. Our method has an end-to-end (including feature computation cost) frame rate of 4.3 frames/sec compared to 16.0 frames/sec of KGS.

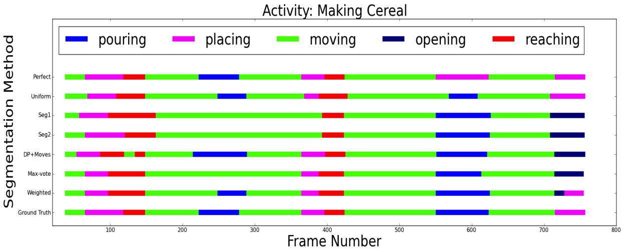

Figure 4. **Illustration of the ambiguity in temporal segmentation.** We compare the sub-activity labeling of various segmentations. Here, *making cereal* activity comprises the sub-activities: *reaching*, *moving*, *pouring* and *placing* as colored in red, green, blue and magenta respectively. The x-axis denotes the time axis numbered with frame numbers. It can be seen that the various individual segmentation methods are not perfect.

Anticipation Results. Table 2 shows the frame-level metrics for anticipating the sub-activity and object affordance labels for 3 seconds in the future on the CAD-120 dataset. We compare our anticipation method against the following baselines:

1. *Chance.* Labels are chosen at random.
2. *Nearest Neighbor.* It first finds an example from the training data which is the most similar (based on feature distance) to the activity observed in the last temporal segment. The sub-activity and object affordance labels of the frames following the matched frames from the exemplar are predicted as the anticipations.

Table 2. **Results for Anticipating Future Sub-activities and Affordances**, computed over 3 seconds in the future (similar trends hold for other anticipation times). Computed from 4-fold cross validation with a new human subject in each fold. Standard error is also reported.

model	Anticipated Object Affordance			Anticipated Sub-activity		
	micro	macro		micro	macro	
	P/R	Precision	Recall	*P/R*	Precision	Recall
chance	8.3 (0.1)	8.3 (0.1)	8.3 (0.1)	10.0 (0.1)	10.0 (0.1)	10.0 (0.1)
Nearest neighbor	48.3 (1.5)	18.4 (1.2)	16.2 (0.9)	22.0 (0.9)	10.7 (0.7)	10.5 (0.5)
KGS+ co-occurance	55.9 (1.7)	10.5 (0.4)	12.9 (0.4)	28.6 (1.8)	9.9 (0.2)	12.8 (0.8)
Ours-segment	59.5 (1.5)	11.3 (0.3)	13.6 (0.4)	34.3 (0.8)	10.4 (0.2)	14.7 (0.2)
Koppula & Saxena (2013)	66.1 (1.9)	71.5 (5.6)	24.8 (1.6)	47.7 (1.6)	61.1 (4.1)	27.6 (2.0)
Ours-full	**67.2** (1.1)	**73.4** (1.8)	**29.1** (1.7)	**49.6** (1.4)	**63.6** (1.2)	**29.9** (1.7)

3. *Co-occurrence Method.* The transition probabilities for sub-activities and affordances are computed from the training data. The observed frames are first labelled using the CRF model proposed by KGS. The anticipated sub-activity and affordances of the objects for the future frames are predicted based on the transition probabilities given the inferred labeling of the last frame.
4. *Our Method - segment.* Our method that only samples the segment level nodes for future sub-activities and object affordances and uses the ground-truth segmentation.
5. *Koppula & Saxena (2013).* Our method that samples the future nodes (both segment and frame level) as described in Section 4, and uses a fixed temporal structure, which in this case is the segmentation output of KGS.

Our method outperforms all the baseline algorithms. Sampling future frame level nodes in addition to just sampling the segment level nodes (row 4) increases the accuracy of affordance and sub-activity anticipation by 6.6% and 13.4% respectively. Our method of sampling the whole graph structure (row 6) achieves the best performance with an increase in macro precision and recall over the best baseline results (row 5) – 1.9% and 4.3% for anticipating object affordances and 2.5% and 2.3% for anticipating sub-activities, respectively. This shows that sampling the graph structures enables us to reason about the spatial and temporal interactions that can happen in the future, which is essential to obtain good anticipation performance.

Figure 5 shows how the macro F1 scores changes with the anticipated time. The average duration of a sub-activity in the CAD-120 dataset is around 3.6 seconds, therefore, an anticipation duration of 10 seconds is over two to three sub-activities. With the increase in anticipation time, the performance of the others approach that of a random chance baseline. Our method outperforms the baselines for all anticipation times and its performance declines gracefully with increase in the anticipation time. The code and videos are available at: `http://pr.cs.cornell.edu/anticipation/`.

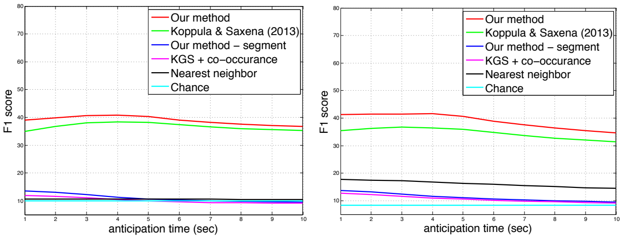

Figure 5. Plots showing how macro F1 score changes with the anticipation time for anticipating sub-activities (left) and object affordances (right).

6. Conclusion

In this paper, we considered the task of detecting the past human activities as well as anticipating the future human activities using object affordances. In the task of detection, most previous works assume ground-truth temporal segmentation is known or simply use some heuristic methods. Since modeling human activities requires rich modeling of the spatio-temporal relations, fixing the temporal segmentation limited the expressive power of CRF-based model in the previous works. In this work, we proposed a method to first obtain potential graph structures that are close to the ground-truth ones by approximating the graph with only additive features. We then designed moves to explore the space of likely graph structures. Our approach also enabled us to anticipate the future activities where considering multiple possible graphs is critical. In the recently growing field of RGB-D vision, our work thus shows a considerable advance by improving the state-of-the-art results on both the detection and anticipation tasks.

Acknowledgements

We thank Jerry Yeh and Yun Jiang for useful discussions. This research was funded in part by ARO award W911NF-12-1-0267, and by Microsoft Faculty Fellowship and NSF Career Award to one of us (Saxena).

References

Anand, A., Koppula, H. S., Joachims, T., and Saxena, A. Contextually guided semantic labeling and search for 3d point clouds. *IJRR*, 2012.

Delaitre, V., Sivic, J., and Laptev, I. Learning person-object interactions for action recognition in still images. In *NIPS*, 2011.

Faraway, J. J., Reed, M. P., and Wang, J. Modelling three-dimensional trajectories by using bezier curves with application to hand motion. *J Royal Stats Soc Series C-Applied Statistics*, 56:571–585, 2007.

Felzenszwalb, P. F. and Huttenlocher, D.P. Efficient graph-based image segmentation. *IJCV*, 59(2), 2004.

Fox, E. B., Sudderth, E. B., Jordan, M. I., and Willsky, A. S. Nonparametric Bayesian learning of switching linear dynamical systems. In *NIPS 21*, 2009.

Gong, S. and Xiang, T. Recognition of group activities using dynamic probabilistic networks. In *ICCV*, 2003.

Gupta, A., Kembhavi, A., and Davis, L.S. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE TPAMI*, 2009.

Harchaoui, Z., Bach, F., and Moulines, E. Kernel change-point analysis. In *NIPS*, 2008.

Hoai, M. and De la Torre, F. Max-margin early event detectors. In *CVPR*, 2012a.

Hoai, M. and De la Torre, F. Maximum margin temporal clustering. In *AISTATS*, 2012b.

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

Hongeng, S. and Nevatia, R. Large-scale event detection using semi-hidden markov models. In *ICCV*, 2003.

Ion, A., Carreira, J., and Sminchisescu, C. Probabilistic Joint Image Segmentation and Labeling. In *NIPS*, 2011.

Jia, Z., Gallagher, A., Saxena, A., and Chen, T. 3d-based reasoning with blocks, support, and stability. In *CVPR*, 2013.

Jiang, Y. and Saxena, A. Infinite latent conditional random fields for modeling environments through humans. In *RSS*, 2013.

Jiang, Y., Lim, M., and Saxena, A. Learning object arrangements in 3d scenes using human context. In *ICML*, 2012a.

Jiang, Y., Lim, M., Zheng, C., and Saxena, A. Learning to place new objects in a scene. *IJRR*, 31(9), 2012b.

Jiang, Y., Koppula, H. S., and Saxena, A. Hallucinated humans as the hidden context for labeling 3d scenes. In *CVPR*, 2013.

Joachims, T., Finley, T., and Yu, C. Cutting-plane training of structural SVMs. *Mach. Learn.*, 77(1), 2009.

Ke, Y., Sukthankar, R., and Hebert, M. Event detection in crowded videos. In *ICCV*, October 2007.

Kitani, K., Ziebart, B. D., Bagnell, J. A., and Hebert, M. Activity forecasting. In *ECCV*, 2012.

Koppula, H. S., Anand, A., Joachims, T., and Saxena, A. Semantic labeling of 3d point clouds for indoor scenes. In *NIPS*, 2011.

Koppula, H. S., Gupta, R., and Saxena, A. Learning human activities and object affordances from rgb-d videos. *IJRR*, 2013.

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

Ly, D. L., Saxena, A., and Lipson, H. Co-evolutionary predictors for kinematic pose inference from rgbd images. In *GECCO*, 2012.

Maji, S., Bourdev, L., and Malik, J. Action recognition from a distributed representation of pose and appearance. In *CVPR*, 2011.

Natarajan, P. and Nevatia, R. Coupled hidden semi markov models for activity recognition. In *WMVC*, 2007.

Nguyen, M. H., Torresani, L., De la Torre, F., and Rother, C. Weakly supervised discriminative localization and classification: a joint learning process. In *ICCV*, 2009.

Ni, B., Wang, G., and Moulin, P. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *ICCV Workshop Consumer Depth Cameras Computer Vision*, 2011.

Oh, S., Rehg, J.M., Balch, T., and Dellaert, F. Learning and inferring motion patterns using parametric segmental switching linear dynamic systems. *IJCV*, 2008.

Prest, A., Schmid, C., and Ferrari, V. Weakly supervised learning of interactions between humans and objects. *IEEE TPAMI*, 34(3):601–614, 2012.

Quattoni, A., Wang, S., Morency, L.-P., Collins, M., and Darrell, T. Hidden-state conditional random fields. *IEEE TPAMI*, 2007.

Ryoo, M.S. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *ICCV*, 2011.

Sarawagi, S. and Cohen, W. W. Semi-markov conditional random fields for information extraction. In *NIPS*, 2004.

Shi, Q., Wang, L., Cheng, L., and Smola, A. Human action segmentation and recognition using discriminative semi-markov models. *IJCV*, 2011.

Shotton, J., Girshick, R., Fitzgibbon, A., Sharp, T., Cook, M., Finocchio, M., Moore, R., Kohli, P., Criminisi, A., Kipman, A., and Blake, A. Efficient human pose estimation from single depth images. *IEEE TPAMI*, 2012.

Simon, T., Nguyen, M. H., De la Torre, F., and Cohn, J. F. Action unit detection with segment-based svms. In *CVPR*, 2010.

Sminchisescu, C., Kanaujia, A., Li, Z., and Metaxas, D. Conditional models for contextual human motion recognition. In *ICCV*, 2005.

Sung, J., Ponce, C., Selman, B., and Saxena, A. Human activity detection from rgbd images. In *AAAI PAIR workshop*, 2011.

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

Xing, Z., Pei, J., Dong, G., and Yu, P. S. Mining Sequence Classifiers for Early Prediction. In *SIAM ICDM*, 2008.

Xuan, X. and Murphy, K. Modeling changing dependency structure in multivariate time series. In *ICML*, 2007.

Yang, W., Wang, Y., and Mori, G. Recognizing human actions from still images with latent poses. In *CVPR*, 2010.

Yao, B. and Fei-Fei, L. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010.

Zhang, H. and Parker, L. E. 4-dimensional local spatio-temporal features for human activity recognition. In *IROS*, 2011.