Recognizing Primitive Interactions by Exploring Actor-Object States

Roman Filipovych Eraldo Ribeiro

Computer Vision and Bio-Inspired Computing Laboratory
Department of Computer Sciences
Florida Institute of Technology, Melbourne, FL 32901, USA
{rfilipov,eribeiro}@fit.edu

Abstract

In this paper, we present a solution to the novel problem of recognizing primitive actor-object interactions from videos. Here, we introduce the concept of actor-object states. *Our method is based on the observation that at the moment of physical contact, both the motion and the appearance of actors are constrained by the target object. We propose a probabilistic framework that automatically learns models in such constrained states. We use joint probability distributions to represent both actor and object appearances as well as their intrinsic spatio-temporal configurations. Finally, we demonstrate the applicability of our approach on series of human-object interaction classification experiments.*

1. Introduction

In this paper, we focus on the problem of recognizing human-object interactions. Experimental evidence suggests that motion information alone may not be sufficient to achieve higher-level reasoning about activities that involve interactions with objects. Indeed, approaches for recognizing human actor-object interactions usually rely on additional contextual information provided in the form of predefined labels or landmark points [13, 4], or a number of electronic sensors [19]. Moreover, actions in such cases are usually strongly constrained and described with a predefined set of semantic entities [16]. As an illustration of the main problem addressed in this paper, let us consider the "grasp a cup" activity in Figure 1. In the figure, the hands approach cups at different speeds and having different spatial properties (*e.g.*, clutched, in the first sequence, and slightly open, in the second). The motion of different actors performing the same interaction activity may differ considerably. However, at the instant of physical contact, actors' motions, appearances, and actor-object spatial configurations become constrained by the target object. These constrained motion and spatial configurations are descriptive of the specific actor-object interaction.

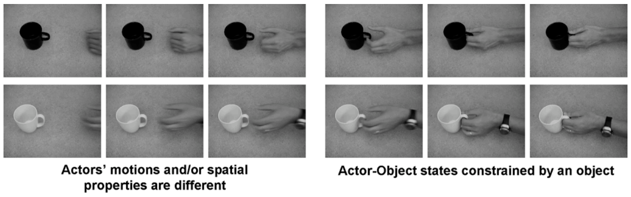

Actors' motions and/or spatial properties are different
Actor-Object states constrained by an object

Figure 1. Constrained actor-object states.

The literature on activity recognition is extensive. In general, activity analysis methods focus on high-level analysis of activities. The analysis is usually semantic-based and aims at recognizing complex activities such as "greeting" or "preparing a french toast". Semantic level description can be accomplished by means of context-free grammars [18], language-based models [16], and graphical models [13, 12, 19, 3, 14]. An overview of efforts made in the area of actions and interactions recognition from a high-level perspective can be found in [1].

However, the recognition of *primitive* human-object interactions is still an open and relatively unexplored problem. An effort in this direction was made by Gupta and Davis [10]. They presented a Bayesian approach that simultaneously estimates object type, location, movement segments, and the effect of movements on objects. However, interactions here are limited to a predefined sequence of motions (*i.e.* reaching, trajectory-like manipulation, and object reaction). Peursum *et al.* [17] suggest the importance of action understanding in object recognition tasks. They use human activity to infer both the location and identity of objects. This idea was consistent with the results obtained by Gupta and Davis [10].

Our actor-object interaction recognition method is inspired by recent developments of probabilistic constella-

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

tion models [15, 9]. We propose a probabilistic graphical model of primitive actor-object interactions that combines information about the interaction's dynamics, and actor-object static appearances and spatial configurations. We term these appearances and joint actor-object configurations as "actor-object states". In our method, a spatio-temporal part-based model of interaction's dynamics guides the process of discovering consistent actor-object states. Selected actor-object states are subsequently modeled within a static part-based framework. No manual input of object or contextual information are required. Video sequences of primitive interactions are the only input to the program. To the best of our knowledge, this is the first general vision-only method for learning primitive human-object interactions without manual input of object information.

2. Integrating Actor-Object Static States and Interaction Dynamics

We commence by defining the main components of our model. An interaction sequence $\boldsymbol{\mathcal{V}}$ can be considered to be the temporal variation of a specific actor-object static state. Let $\boldsymbol{\mathcal{S}} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$ be a set of K discrete static states sampled from the space of all representative static states of an interaction class. Let $\boldsymbol{\mathcal{M}}$ be the spatio-temporal information extracted from the video. In this paper, this information is obtained using spatio-temporal features [7, 11]. Let $\boldsymbol{\mathcal{X}}$ represent simultaneously a particular spatio-temporal configuration of static states and interaction dynamics. The term "actor-object states" will represent the specific joint appearance and spatial configuration of an actor and object in an video frame. Let $p(\boldsymbol{\mathcal{V}}|\boldsymbol{\mathcal{X}})$ be the likelihood of observing a particular video sequence given that an interaction is at some spatio-temporal location. From Bayes' theorem:

$$\begin{aligned} p(\boldsymbol{\mathcal{X}}|\boldsymbol{\mathcal{V}}) &\propto p(\boldsymbol{\mathcal{V}}|\boldsymbol{\mathcal{X}})\; p(\boldsymbol{\mathcal{X}}) \\ &\propto \underbrace{p(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}})}_{\substack{\text{static states'} \\ \text{appearance}}}\; \underbrace{p(\boldsymbol{\mathcal{M}}|\boldsymbol{\mathcal{X}})}_{\substack{\text{dynamics'} \\ \text{appearance}}}\; \underbrace{p(\boldsymbol{\mathcal{X}})}_{\substack{\text{spatio-temporal} \\ \text{configuration}}} \end{aligned} \qquad (1)$$

The appearance of both static states and dynamics are assumed to be statistically independent. As a result, the likelihood term in (1) can be factorized into two components. Following the part-based object factorization suggested by Crandall and Huttenlocher [5], we assume that the parts' spatio-temporal arrangement can be encoded into the prior probability distribution while the likelihood function encodes their appearance.

Spatio-Temporal Prior Model. We begin by assuming that a static-state $\mathcal{S}_i \in \boldsymbol{\mathcal{S}}$ can be subdivided into a number of non-overlapping subregions such that $\mathcal{S}_i = \{(\mathbf{a}_1^{(i)}, \mathbf{x}_1^{(i)}), \ldots, (\mathbf{a}_{N_{\mathcal{S}_i}}^{(i)}, \mathbf{x}_{N_{\mathcal{S}_i}}^{(i)})\}$, where the pair $(\mathbf{a}_j^{(i)}, \mathbf{x}_j^{(i)})$ consists of the appearance $\mathbf{a}$ and the spatio-temporal location $\mathbf{x}$ of the subregion j for the model of static-state $\mathcal{S}_i$, respectively. Here, $N_{\mathcal{S}_i}$ is the total number of subregions for the static-state $\mathcal{S}_i$. The temporal position of the static-state in the video sequence serves as the temporal coordinate of the parts' locations. Similarly, the dynamic information required by our model is represented by a sparse set of $N_{\mathcal{M}}$ spatio-temporal features [7, 11] given by $\mathcal{M} = \{(\mathbf{a}_1^{(\mathcal{M})}, \mathbf{x}_1^{(\mathcal{M})}), \ldots, (\mathbf{a}_{N_{\mathcal{M}}}^{(\mathcal{M})}, \mathbf{x}_{N_{\mathcal{M}}}^{(\mathcal{M})})\}$. For simplicity, we model both static-state and dynamic information using directed acyclic star graphs. This is similar to the part-based object model suggested by Fergus *et al.* [8]. Here, a particular vertex is assigned to be a landmark vertex $(\mathbf{a}_r^{(i)}, \mathbf{x}_r^{(i)})$ for the static-state $\mathcal{S}_i$. A similar landmark vertex assignment is done for the dynamics model, $(\mathbf{a}_r^{(\mathcal{M})}, \mathbf{x}_r^{(\mathcal{M})})$. The remaining vertices within each partial model are conditioned on the corresponding landmark vertex. Figure 2(b) shows an example of part-based models for two static states of the "grasp a cup" interaction. Finally, we obtain a multi-layered tree-structured actor-object interaction model by conditioning the landmark vertices of the static-state model graphs on the landmark vertex of the dynamics model graph. An example of the interaction model is shown in Figures 2(a,c). Graph arrows indicate vertices' conditional dependences. The joint distribution of the spatial interaction configuration can be derived from the graphical model in Figure 2(c):

$$p(\boldsymbol{\mathcal{X}}) = p(\mathbf{x}^{(\mathcal{M})}) \prod_{\mathcal{S}_i \in \boldsymbol{\mathcal{S}}} p(\mathbf{x}^{(i)}|\mathbf{x}^{(\mathcal{M})}) \qquad (2)$$

where $\mathbf{x}^{(i)}$ is the spatio-temporal configuration of the static-state $\mathcal{S}_i$, and $\mathbf{x}^{(\mathcal{M})}$ is the spatio-temporal configuration of the dynamics model. The probability distributions that compose Equation 2 are: $p(\mathbf{x}^{(\mathcal{M})}) = p(\mathbf{x}_r^{(\mathcal{M})}) \prod_{j \neq r} p(\mathbf{x}_j^{(\mathcal{M})}|\mathbf{x}_r^{(\mathcal{M})})$ and $p(\mathbf{x}^{(i)}|\mathbf{x}^{(\mathcal{M})}) = p(\mathbf{x}_r^{(i)}|\mathbf{x}_r^{(\mathcal{M})}) \prod_{j \neq r} p(\mathbf{x}_j^{(i)}|\mathbf{x}_r^{(i)})$. The partial models dependence is based solely on their spatio-temporal configuration within the global model (*i.e.*, partial models appearances are statistically independent).

Appearance Model. Under the appearance independence assumption, the appearance likelihood of the static-state $\mathcal{S}_i$ can be written as $p(\mathcal{S}_i|\boldsymbol{\mathcal{X}}) = \prod_j^{N_{\mathcal{S}_i}} p(\mathbf{a}_j^{(i)}|\mathbf{x}_j^{(i)})$. Similarly, the dynamics model appearance likelihood is $p(\mathcal{M}|\boldsymbol{\mathcal{X}}) = \prod_j^{N_{\mathcal{M}}} p(\mathbf{a}_j^{(\mathcal{M})}|\mathbf{x}_j^{(\mathcal{M})})$. The likelihood term in (1) becomes:

$$p(\boldsymbol{\mathcal{V}}|\boldsymbol{\mathcal{X}}) = \prod_i^K \prod_j^{N_{\mathcal{S}_i}} p(\mathbf{a}_j^{(i)}|\mathbf{x}_j^{(i)}) \times \prod_j^{N_{\mathcal{M}}} p(\mathbf{a}_j^{(\mathcal{M})}|\mathbf{x}_j^{(\mathcal{M})}) \qquad (3)$$

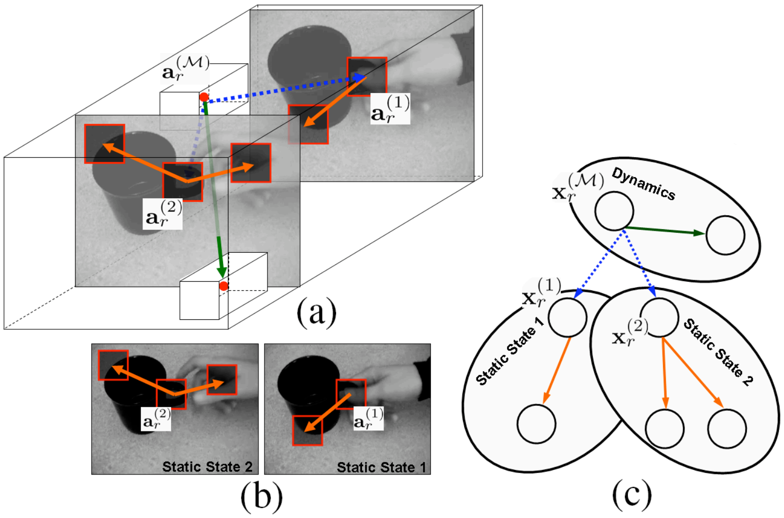


Figure 2. (a) Spatio-temporal part-based model of interaction. (b) Static states part-based models; (c) Interaction model graph.

3. Learning Interactions

The factorization in (2) and (3) allows for a modular learning procedure given a set of unsegmented training videos $\{\boldsymbol{\mathcal{V}}_1, \ldots, \boldsymbol{\mathcal{V}}_L\}$. The learning steps are the following (Figure 3).

Learning Step 1 - Learning the Dynamics Model. We begin by modeling the probabilities of subregion locations in $p(\mathbf{x}^{(\mathcal{M})})$ from (2). Conditional distributions relating independent Gaussian distributions are also Gaussian [2]. As a result, the conditional densities in the components of (2) take a particularly simple form [2]. We extract a set of spatio-temporal interest points [7], and associate a spatio-temporal location with every extracted subregion. We adopt the learning process described by [5] to obtain an initial spatio-temporal model and the optimal number of parts. An Expectation-Maximization (EM)-based procedure is used to simultaneously refine the initial estimates of the appearances and the spatial parameters.

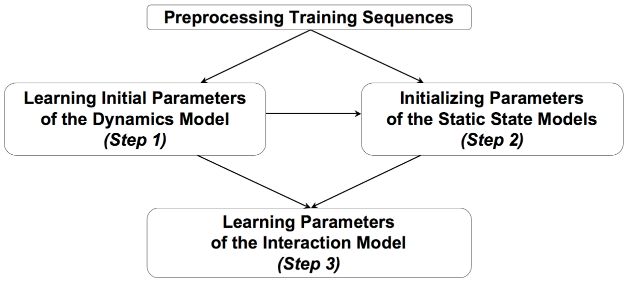

Preprocessing Training Sequences
Learning Initial Parameters of the Dynamics Model (Step 1)
Initializing Parameters of the Static State Models (Step 2)
Learning Parameters of the Interaction Model (Step 3)

Figure 3. Interaction learning process.

Learning Step 2 - Creating Initial Static-State Models. The input to this step are interest subregions extracted from all frames of the training sequences. Subregion locations are given by the x- and y-coordinates of the subregion in the frame image, and the additional temporal t-coordinate (*i.e.*, frame-position). However, some subregions may have similar appearance across various frames (*e.g.*, the appearance of the toy car remains unchanged during the "push a toy car" interaction). Hence, simply clustering appearances would make indistinguishable some similar parts belonging to different static states. Secondly, parts of a specific static-state model must have zero temporal variance as they naturally belong to the same frame. We address this initialization problem by using the model of interaction dynamics to restrict the space of input subregions. For every training sequence, we obtain MAP locations of the dynamics model:

$$\hat{\mathbf{x}}^{(\mathcal{M})} = \arg\max_{\mathbf{x}} p(\mathcal{M}|\mathbf{x}^{(\mathcal{M})})p(\mathbf{x}^{(\mathcal{M})}) \qquad (4)$$

For every sequence, the maximization in (4) results in the location $(\hat{x}, \hat{y}, \hat{t})$ of the model's landmark part. A set of initial samples $\mathbf{t_0} = \{t_1, t_2, ..., t_K\}$ of temporal displacements is created, where t_i is the static-state $\mathcal{S}_i$ temporal displacement with respect to the dynamics model's landmark node. For a given static-state $\mathcal{S}_i$, we select subregions for which temporal displacements are between $(t_i - \Delta t, t_i + \Delta t)$, where the constant Δt defines the frame range containing the corresponding static state. These subregions are subsequently used to obtain the candidate parts and initial parameters of the static-state $\mathcal{S}_i$ models. Candidate parts selection is performed as in Step 1. We cluster pose subregions to form initial parts of the underlying pose, and discard parts with low descriptiveness power with respect to their appearance. We use the descriptiveness evaluation procedure described in [5]. Learned pose parts are organized into a star-graph structure with the most descriptive part as the landmark node. Initial spatio-temporal parameters are estimated from the parts' maximum likelihood locations as follows:

$$\hat{\mathbf{x}}^{(i)} = \arg\max_{\mathbf{x}} p(\mathcal{S}_i|\mathbf{x}^{(i)}) \tag{5}$$

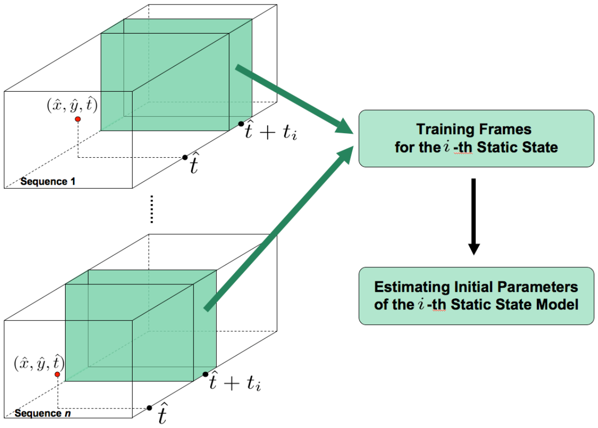

$(\hat{x}, \hat{y}, \hat{t})$
$\hat{t} + t_i$
$\hat{t}$
Sequence 1
$(\hat{x}, \hat{y}, \hat{t})$
$\hat{t} + t_i$
$\hat{t}$
Sequence n
Training Frames for the i -th Static State
Estimating Initial Parameters of the i -th Static State Model

Figure 4. Initial static-state model. Dynamics model temporal location $\hat{t}$ is estimated for each sequence. 2D interest subregions are extracted from frames in the temporal neighborhood $\hat{t}+t_i$. Subregions are clustered to form an initial static-state model for the first state. Subsequent static-state models are similarly determined.

Learning Step 3 - Creating the Global Model. In this step, the initial static-state models are combined with the model of interaction dynamics into the global model of interaction as in Figure 5(a). The initial parameters of the conditional distributions $p(\mathbf{x}_r^{(i)}|\mathbf{x}_r^{(\mathcal{M})})$ that contribute to $p(\mathbf{x}^{(i)}|\mathbf{x}^{(\mathcal{M})})$ in Equation 2 are estimated from the MAP localizations of the static states in the training sequences. However, not only the initial static-state models contain noisy parts, but the parameters of the conditional distributions $p(\mathbf{x}_r^{(i)}|\mathbf{x}_r^{(\mathcal{M})})$ are very inaccurate. We revise the parameters of the global model with the EM algorithm. The EM algorithm reestimates all model parameters including those of the dynamics model. Our EM algorithm's Expectation-step MAP estimation considers only those model configurations where static-state parts belong to the same frame. (See Section 4 for details). However, while the revised conditional distributions become better defined, the updated model still contains overlapping parts. Overlapping parts are removed and parameters revised with the EM algorithm once again. Figure 5(b) shows an example model with overlapping parts removed.

Finally, from the set of all learned static-state models we retain only those that are temporally well-defined. This is done by pruning the static-state model subgraphs whose landmark nodes have the largest temporal variance when conditioned on the dynamics model. Also, when two intervals $(t_i - \Delta t, t_i + \Delta t)$ and $(t_j - \Delta t, t_j + \Delta t)$ overlap, several instances of same static state may be learned. Therefore, we greedily select a subset of static-state models such that when conditioned on the landmark node of the dynamics model $\mathbf{x}_r^{(\mathcal{M})}$, the mean locations of their landmark nodes $\mathbf{x}_r^{(i)}$ are separated by a predefined distance (*e.g.*, one frame).

4. Classification and Inference

Interaction recognition can then be posed as a detection problem. We seek for the spatio-temporal location in the video sequence that maximizes the posterior probability of the interaction's location:

$$\hat{\boldsymbol{\mathcal{X}}} = \arg\max_{\boldsymbol{\mathcal{X}}} p(\boldsymbol{\mathcal{X}}|\boldsymbol{\mathcal{V}}) \tag{6}$$

We expect that parts representing the same static-state belong to the same video frame. As a result, the global model MAP search space can be significantly reduced. Our exact inference algorithm is as follows: (i) Consider all states of the variable $\mathbf{x}_r^{(\mathcal{M})}$; (ii) Consider all states of $\mathbf{x}_j^{(\mathcal{M})}$; (iii) Consider all states of $\mathbf{x}_r^{(i)}$. For every state $x_r^{(i)}$ of $\mathbf{x}_r^{(i)}$, consider only those states of $\mathbf{x}_j^{(i)}$ that have the same temporal coordinate as $x_r^{(i)}$. Obtain the maximizing configuration.

5. Experimental Results

Interactions Dataset. We acquired our own dataset of videos with primitive interactions. Vision-based human-object interaction recognition is a novel problem with no widely available datasets. Complex scenarios were deliberately chosen to motivate future improvements of human-object interaction methods. Example frames from our interaction dataset are shown in Figure 6. The dataset consists of videos of eight different actor-object interaction types performed by ten individuals in two different scenarios. The interactions are "grasp a cup", "grasp a fork", "touch a fork", "grasp a spoon", "touch a spoon", "grasp a toy car", "touch a toy car" and "push a toy car". Every individual performed interactions with a unique set of objects (*e.g.*, different cups were used by different individuals in a "grasp a cup" interaction). Sequences had clean and cluttered background, respectively. In the "cluttered background" scenario, the background was changed for every individual and every interaction type. Any two interaction types from our dataset differ in one of the following three aspects: (1) different objects and different motions (*i.e.*, "grasp a cup" vs. "touch a fork"); (2) similar objects and different motions (*i.e.*, "grasp a fork" vs. "touch a fork"); and (3) different objects and similar motions (*i.e.*, "grasp a fork" vs. "grasp a spoon").

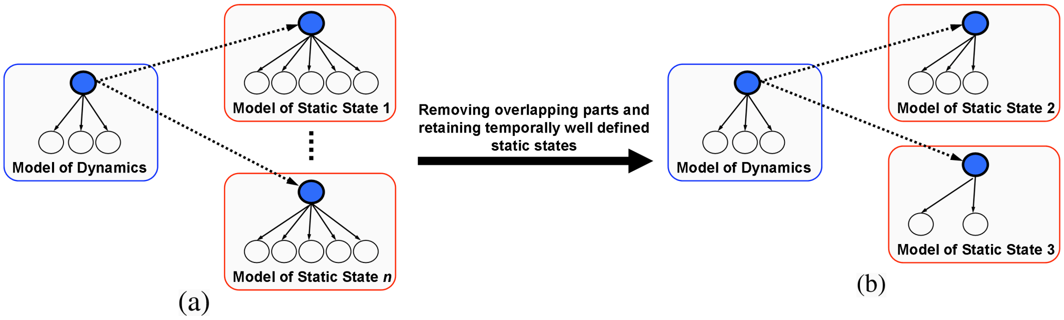
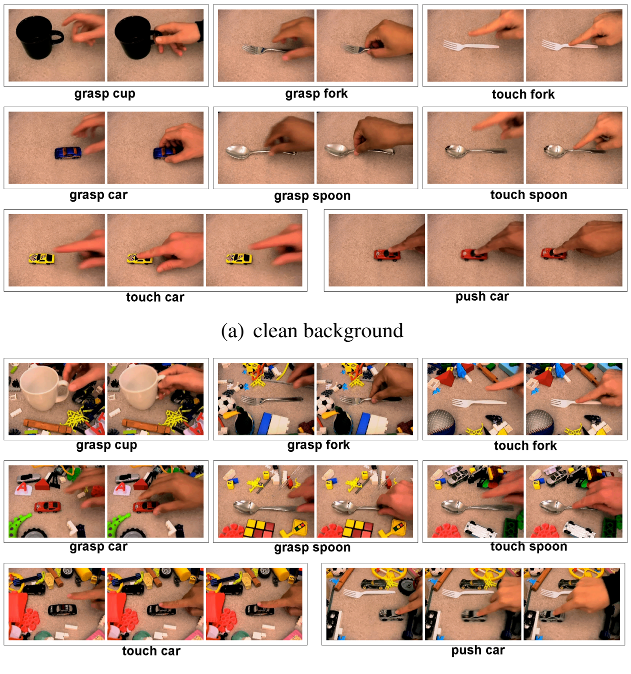


Figure 5. (a) Initial global model; (b) Nodes corresponding to overlapping parts are pruned and only temporally well-defined static states are retained.

Therefore, we believe that this dataset is suitable for evaluating our method's validity. The above choice of interactions was inspired by experiments using functional neuroimaging in humans [6]. These experiments revealed regions of the parietal lobes that are specialized for particular visuomotor actions such as reaching and grasping. Videos were acquired with a CCD camera at thirty frames per second. Frames were downsized to 144×180 pixels.

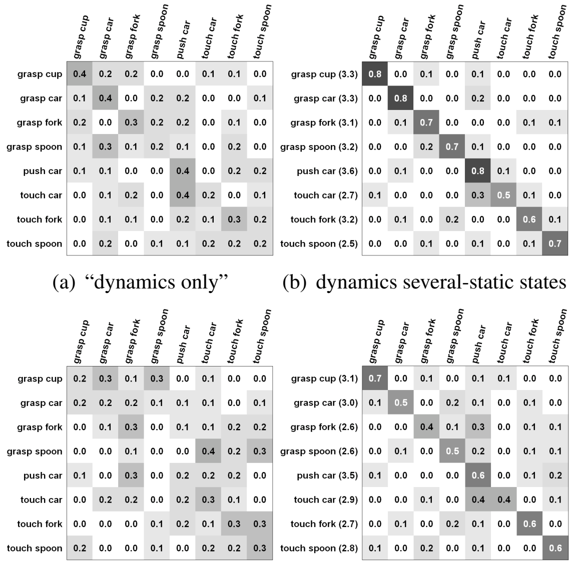


(a) clean background

(b) cluttered background

Figure 6. Sample frames from our interaction dataset.

Video Data Preparation. In our implementation, we began by obtaining a set of Gaussian smoothed edge-maps of square patches centered at the previously detected interest points. Interest point locations were detected using a Harris operator. The edge maps were used to create the static-state models. Features required to create the dynamics model were obtained using the spatio-temporal interest point detector described in [7]. In all cases, the data dimensionality was reduced using principal component analysis (PCA).



	grasp cup	grasp car	grasp fork	grasp spoon	push car	touch car	touch fork	touch spoon
grasp cup	0.4	0.2	0.2	0.0	0.0	0.1	0.1	0.0
grasp car	0.1	0.4	0.0	0.2	0.2	0.0	0.0	0.1
grasp fork	0.2	0.0	0.3	0.2	0.2	0.0	0.1	0.0
grasp spoon	0.1	0.3	0.1	0.2	0.1	0.0	0.2	0.0
push car	0.1	0.1	0.0	0.0	0.4	0.0	0.2	0.2
touch car	0.0	0.1	0.2	0.0	0.4	0.2	0.0	0.1
touch fork	0.0	0.1	0.1	0.0	0.2	0.1	0.3	0.2
touch spoon	0.0	0.2	0.0	0.1	0.1	0.2	0.2	0.2

(a) "dynamics only"

	grasp cup	grasp car	grasp fork	grasp spoon	push car	touch car	touch fork	touch spoon
grasp cup (3.3)	0.8	0.0	0.1	0.0	0.1	0.0	0.0	0.0
grasp car (3.3)	0.0	0.8	0.0	0.0	0.2	0.0	0.0	0.0
grasp fork (3.1)	0.0	0.1	0.7	0.0	0.0	0.0	0.1	0.1
grasp spoon (3.2)	0.0	0.0	0.2	0.7	0.1	0.0	0.0	0.0
push car (3.6)	0.0	0.1	0.0	0.0	0.8	0.1	0.0	0.0
touch car (2.7)	0.1	0.0	0.0	0.0	0.3	0.5	0.1	0.0
touch fork (3.2)	0.0	0.1	0.0	0.2	0.0	0.0	0.6	0.1
touch spoon (2.5)	0.0	0.0	0.1	0.0	0.1	0.0	0.1	0.7

(b) dynamics several-static states

	grasp cup	grasp car	grasp fork	grasp spoon	push car	touch car	touch fork	touch spoon
grasp cup	0.2	0.3	0.1	0.3	0.0	0.1	0.0	0.0
grasp car	0.2	0.2	0.2	0.1	0.1	0.1	0.0	0.1
grasp fork	0.0	0.1	0.3	0.0	0.1	0.1	0.2	0.2
grasp spoon	0.0	0.0	0.1	0.0	0.0	0.4	0.2	0.3
push car	0.1	0.0	0.3	0.0	0.2	0.2	0.2	0.0
touch car	0.0	0.2	0.2	0.0	0.2	0.3	0.1	0.0
touch fork	0.0	0.0	0.0	0.1	0.2	0.1	0.3	0.3
touch spoon	0.2	0.0	0.0	0.1	0.0	0.2	0.2	0.3

(c) "dynamics only"

	grasp cup	grasp car	grasp fork	grasp spoon	push car	touch car	touch fork	touch spoon
grasp cup (3.1)	0.7	0.0	0.1	0.0	0.1	0.1	0.0	0.0
grasp car (3.0)	0.1	0.5	0.0	0.2	0.1	0.0	0.1	0.0
grasp fork (2.6)	0.0	0.0	0.4	0.1	0.3	0.0	0.1	0.1
grasp spoon (2.6)	0.0	0.1	0.0	0.5	0.2	0.0	0.1	0.1
push car (3.5)	0.1	0.0	0.0	0.0	0.6	0.0	0.1	0.2
touch car (2.9)	0.0	0.0	0.1	0.0	0.4	0.4	0.0	0.1
touch fork (2.7)	0.0	0.1	0.0	0.2	0.1	0.0	0.6	0.0
touch spoon (2.8)	0.1	0.0	0.2	0.0	0.1	0.0	0.0	0.6

(d) dynamics several-static states

Figure 7. Confusion matrices. (a) clean scenario (30.0% correct classification); (b) clean scenario (70.0% correct classification); (c) cluttered scenario (23.0% correct classification); (d) cluttered scenario (54.0% correct classification).

Classification. We performed four experiments. For each scenario, we first learned a dynamics model of an interaction. We show classification results obtained for each scenario using dynamics information only as well as the combination of dynamics information and static-state information. A leave-one-out evaluation scheme was used for classification. Labeling decisions were made based only on best

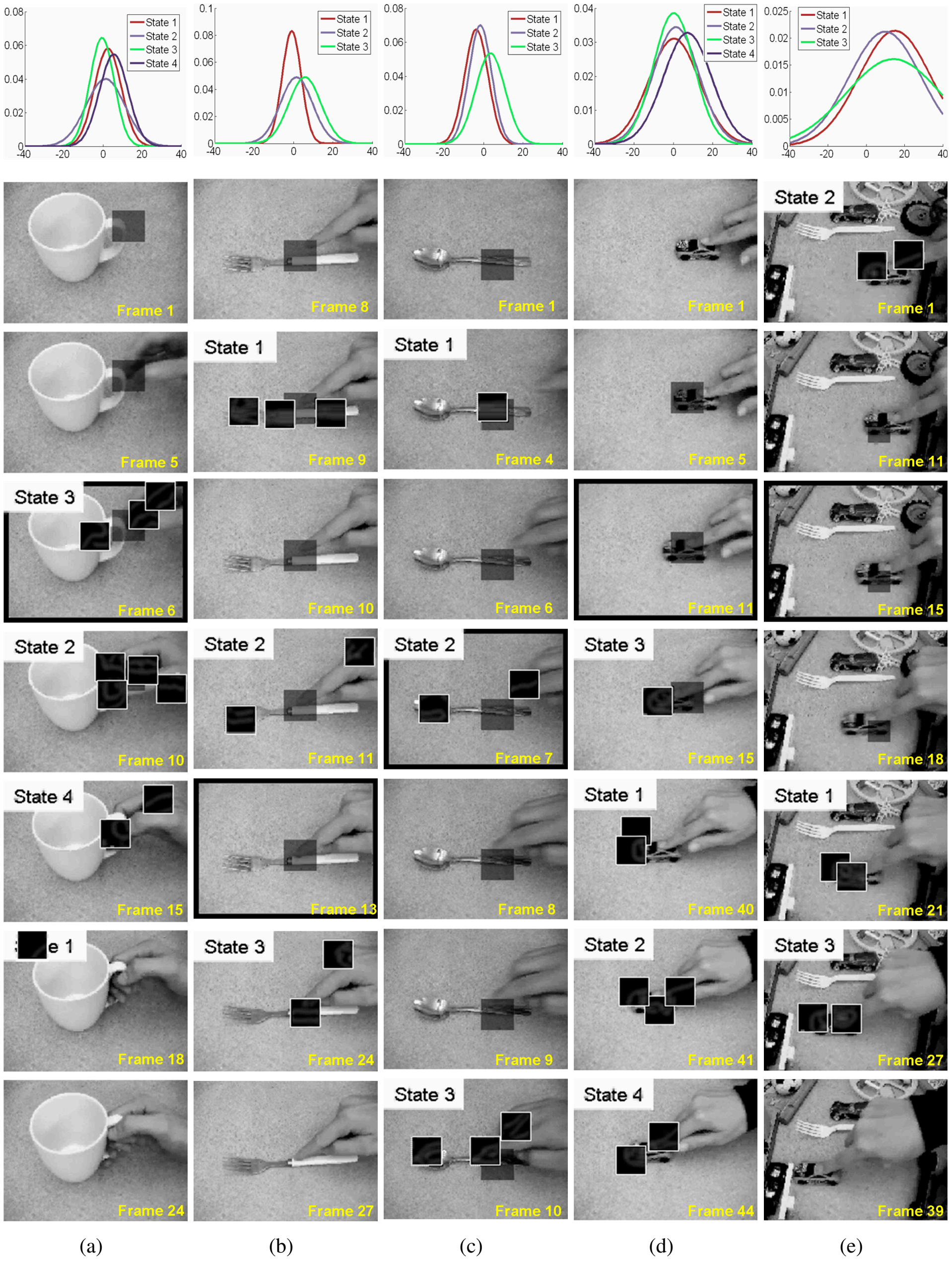


Figure 8. Learned models of interactions superimposed at the detected locations. The plots represent time-axis cross-sections of the "dynamics-static state" conditional distributions. White borders indicate static-state parts. Grayed rectangles indicate slices of the landmark spatio-temporal subregion in the corresponding frames. Dark border highlights the temporal coordinate of the spatio-temporal feature corresponding to the landmark node of dynamics.

model match. The "dynamics only" model (*i.e.*, no static-state information included) achieved only 30% and 23% correct recognition for the clean and cluttered scenarios, respectively. Confusion matrices for these results are shown in Figure 7(a,c). Results suggest that motion alone was not sufficient to perform accurate classification, and would have to be reinforced with the static-state models.

Next, static-state information was included into the framework. The initial static-state models were obtained using $\mathbf{t_0} = \{t_1, t_2, ..., t_K\}$ as initial temporal displacements (Learning Step 2). The number of initial static-state models was set to five, and we selected $\mathbf{t_0} = \{-14, -7, 0, 7, 14\}$ and $\Delta t = 4$ frames. In our approach, the parameter that indirectly governs the number of static-state models is the threshold on the temporal distance between any two static states. This threshold was set to one frame. Consequently, starting from the static-state with the lowest temporal variance, we greedily retained static-state models while meeting the temporal threshold. Confusion matrices generated by these classification results are shown in Figure 7(b,d). The figure also displays (in parentheses along side the interaction types) the average number of static-state models retained in the global model for a given interaction. Overall recognition performance in these experiments was 70.0% for the clean background scenario and 54.0% for the cluttered background scenario. This was significantly higher than the results obtained by the dynamics-only model. Improvements are still needed for cluttered scenarios. Weak recognition results for highly noisy and ambiguous interactions were expected. Figure 8 shows qualitative results for some interaction models from the latter experiments. In the figure, models of several interactions are superimposed on test sequences at the detected locations. The plots represent the time-axis cross-sections of "dynamics-static state" conditional distributions. Static-state parts are represented with the white border. Grayed rectangles represent slices of corresponding landmark spatio-temporal subregion. The model parts' appearances shown in the Figure were obtained using the closest vectors indices in the PCA-reduced feature space. Dark border highlights the temporal coordinate of the spatio-temporal feature corresponding to the landmark node of dynamics.

6. Conclusions

We presented a solution to a novel problem of recognizing primitive actor-object interactions. The concept of actor-object states was introduced using a probabilistic framework. The proposed recognition method combines static-states information with the video's motion dynamics to form a global actor-object interaction model. Additionally, we introduced a dataset of primitive actor-object interactions and showed that our approach is effective for human-object interaction classification. Our current method is view-dependent. However, single-view camera setup is a common scenario for many surveillance applications. In these scenarios, direction of motion can be quite similar (e.g., motion direction when opening a fridge is similar across different agents). Also, our approach is independent of the type of interest features. Candidate parts could be extracted by sampling the video's spatio-temporal subregions, and any existing interest feature extraction method would work. Finally, our approach is not limited to gesture-specific interactions, and should work well with full-body interactions. Future directions of investigation include a study and use of alternative appearance models.

Acknowledgments. Research supported by U.S. Office of Naval Research under contract: N00014-05-1-0764.

References

[1] J. K. Aggarwal and S. Park. Human motion: Modeling and recognition of actions and interactions. In *3DPVT*, pages 640–647, Washington, DC, USA, 2004.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[3] O. Boiman and M. Irani. Detecting irregularities in images and in video. In *CVPR*, pages I: 462–469, 2005.

[4] R. Chelappa, A. K. Roy-Chowdhury, and S. K. Zhou. *Recognition of Humans and Their Activities Using Video*. Morgan & Claypool Publishers, 2005.

[5] D. J. Crandall and D. P. Huttenlocher. Weakly supervised learning of part-based spatial models for visual object recognition. In *ECCV (1)*, volume 3951 of *LNCS*, pages 16–29. Springer, 2006.

[6] J. C. C. Culham and K. F. F. Valyear. Human parietal cortex in action. *Current Opinion of Neurobiology*, 2, March 2006.

[7] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, October 2005.

[8] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *CVPR (2)*, pages 380–387, 2005.

[9] R. Filipovych and E. Ribeiro. Combining models of pose and dynamics for human motion recognition. In *ISVC 2007*, Lake Tahoe, Nevada, USA, November 2007.

[10] A. Gupta and L. S. Davis. Objects in action: An approach for combining action understanding and object perception. In *CVPR*, pages 1–8, 2007.

[11] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, page 432, Nice, France, October 2003.

[12] B. Laxton, J. Lim, and D. Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In *CVPR*, pages 1–8, 2007.

[13] N. Nguyen, S. Venkatesh, and H. Bui. Recognising behaviours of multiple people with hierarchical probabilistic model and statistical data association. In *BMVC*, page III:1239, 2006.

[14] J. Niebles, H. Wang, H. Wang, and L. Fei Fei. Unsupervised learning of human action categories using spatial-temporal words. In *BMVC, Edinburgh*, page III:1249, 2006.

[15] J. C. Niebles and L. Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *CVPR*, Minneapolis, USA, June 2007.

[16] S. Park and J. K. Aggarwal. Semantic-level understanding of human actions and interactions using event hierarchy. In *CVPRW*, volume 1, page 12, Washington, DC, USA, 2004.

[17] P. Peursum, G. West, and S. Venkatesh. Combining image regions and human activity for indirect object recognition in indoor wide-angle views. In *ICCV*, volume 1, pages 82–89, 2005.

[18] M. S. Ryoo and J. K. Aggarwal. Recognition of composite human activities through context-free grammar based representation. In *CVPR*, pages 1709–1718, 2006.

[19] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. Rehg. A scalable approach to activity recognition based on object use. In *ICCV*, 2007.