DELIVERABLE 4.5

Evaluation of the activity recognition system

Author(s): Ninghang Hu, Ben Kröse

Project no: 287624

Project acronym: ACCOMPANY

Project title: Acceptable robotiCs COMPanions for AgeiNg Years

Doc. Status: Draft
Doc. Nature: Template
Version: 0.1
Actual date of delivery: 30 March 2014
Contractual date of delivery: Month 30
Project start date: 01/10/2011
Project duration: 36 months
Peer Reviewer: IPA

Project Acronym: ACCOMPANY
Project Title: **Acceptable robotiCs COMPanions for AgeiNg Years**

EUROPEAN COMMISSION, FP7-ICT-2011-07, 7th FRAMEWORK PROGRAMME
ICT Call 7 - Objective 5.4 for Ageing & Wellbeing

Grant Agreement Number: 287624

DOCUMENT HISTORY

Version	Date	Status	Changes	Author(s)
0.0	**2013-10-8**	**Draft**	**Initial Draft**	**Ben Kröse**
0.1	**2013-10-8**	**Draft**		**Ninghang Hu**

AUTHORS & CONTRIBUTERS

Partner Acronym	Partner Full Name	Person
UvA	**University of Amsterdam**	**Ben Kröse**
UvA	**University of Amsterdam**	**Ninghang Hu**

Short description

This deliverable reports on the evaluation of the activity recognition system in household chores in WP4 of the ACCOMPANY project.

We have already built a system to recognize low-level sub-activity sequence (accepted at ICRA 14') as well as a hierarchical approach for recognizing high-level activities (submitted to ROMAN 14'). Our experiments consist of multiple activities of users.

To evaluate the system, we use the benchmark dataset CAD-120 [1]. We choose the CAD-120 dataset for evaluation because of the following reasons: 1) CAD-120 is a very challenging dataset that presents significant variations of activities, cluttered background, viewpoint changes, and partial occlusions. 2) The dataset has been used in many recent works in the robotics research [1]–[3]. Therefore we can easily compare the performance to the state-of-the-art approaches. 3) The dataset is captured by a RGB-D camera mounted on the robot, which is closely related to the applications in robotics.

In order to incorporate confidence of annotation into our activity recognition framework, we proposed the method of soft labeling, which allows annotators to assign multiple, weighted, labels to data segments.

We are working on creating a new benchmark dataset in Troyes. The dataset will incorporate data from ambient sensors, robot sensors, overhead cameras, therefore it can be used for multi-dimensional research. The dataset will be recorded with real elderly people and will be annotated by the soft labeling method that we have proposed.

Table of Contents

1 Introduction

This deliverable reports on the evaluation of the activity recognition system in household chores in WP4 of the ACCOMPANY project.

We developed a novel discriminative model for the recognition of human activities. The novel model was tested on the (CAD-120 benchmark data set. Experimental results on this data set indicate that our model outperforms the current state-of-the-art approach by over 5% in both precision and recall, while our model is more efficient in terms of computation.

Based on the recognized sub-level activities, we proposed a two-layered approach that can recognize sub-level activities and high-level activities successively. In the first layer, the low-level activities are recognized based on the RGB-D video. In the second layer, we use the recognized low-level activities as input features for estimating high-level activities. Our model is embedded with a latent node, so that it can capture a richer class of sub-level semantics compared with the traditional approach. Our model is evaluated on a challenging benchmark dataset. We show that the proposed approach outperforms the single-layered approach, suggesting that the hierarchical nature of the model is able to better explain the observed data. The results also show that our model outperforms the state-of-the-art approach in accuracy, precision and recall.

In order to incorporate confidence of annotation into our activity recognition framework, we proposed the method of soft labeling, which allows annotators to assign multiple, weighted, labels to data segments. This is useful in many situations, e.g. when the labels are uncertain, when a part of the labels are missing, or when multiple annotators assign inconsistent labels. We treat the activity recognition task as a sequential labeling problem. Latent variables are embedded to exploit sub-level semantics for better estimation. We propose a novel method for learning model parameters from soft-labeled data in a max-margin framework. The model is evaluated on a challenging dataset (CAD-120), which is captured by a RGB-D sensor mounted on the robot. To simulate the uncertainty in data annotation, we randomly change the labels for transition segments. The results show significant improvement over the state-of-the-art approach.

The systems are evaluated on the benchmark dataset in order to compare with the state-of-the-art approaches. We are working on creating a new benchmark dataset in Troyes. The dataset will incorporate data from ambient sensors, robot sensors, and overhead cameras, therefore it can be used for multi-dimensional research. The dataset will be recorded with real elderly people and will be annotated by the soft labeling method that we have proposed.

The report is structured as follows: Section 2 describes our new approach for activity recognition. This work has been accepted for publication at ICRA14. Section 3 describes the method of recognizing high-level activities. The full papers and submissions are attached as Appendices A, B. The work of soft annotation is still under review. It will be provided once the paper gets accepted. In this paper, we present a method to train discriminative graphical models, which allows annotation uncertainty to be explicitly incorporated, in the form of soft

labeling. The advantage of soft labeling is that it incorporates the uncertainty of labels during annotation and can deal with missing labels or annotator disagreement.

2 Learning Latent Structure for Activity Recognition

Robotic companions which help people in their daily life are currently a widely studied topic. In Human-Robot Interaction (HRI), it is very important that the human activities are recognized accurately and efficiently.

In this section, we present a novel graphical model for human activity recognition. The task of activity recognition is to find the most likely underlying activity sequence based on the observations generated from the sensors. Typical sensors include ambient cameras, contact switches, thermometers, pressure sensors, and the sensors on the robot, e.g. RGB-D sensor and Laser Range Finder.

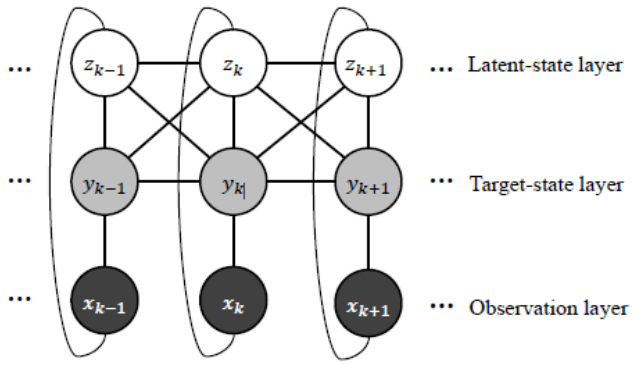


Figure 1: the proposed graphical model

Probabilistic Graphical Models have been widely used for recognizing human activities in both robotics and smart home scenarios. The graphical models can be divided into two categories: generative models [4], [5] and discriminative models [1], [6], [7]. The generative models require making assumptions on both the correlation of data and on how the data is distributed given the activity state. The risk is that the assumptions may not reflect the true attributes of the data. The discriminative models, in contrast, only focus on modeling the posterior probability regardless of how the data are distributed. The robotic and smart environment scenarios are usually equipped with a combination of multiple sensors. Some of these sensors may be highly correlated, both in the temporal and spatial domain, e.g. a pressure sensor on the mattress and a motion sensor above the bed. In these scenarios, the discriminative models provide us a natural way of data fusion for human activity recognition.

The linear-chain Conditional Random Field (CRF) is one of the most popular discriminative models and has been used for many applications. Linear-chain CRFs are efficient models because the exact inference is tractable. However, they are limited in the way that they cannot capture the intermediate structures within the target states [8]. By adding an extra layer of latent variables, the model allows for more flexibility and therefore it can be used for

modeling more complex data. The names of these models are interchangeable in the literature, such as Hidden-Unit CRF [9], Hidden-state CRF [8] or Hidden CRF [10].

In this section, we present a latent CRF model for human activity recognition. For simplicity, we use “latent variables” to refer to the augmented hidden layer, as they are unknown either in training or testing. The “target variables”, which are observed during training but not testing, represent the target states that we would like to predict, e.g. the activity labels. See Figure 1 for the graphical model and the difference between latent variables and target variables. We evaluate the model using the RGB-D data from the benchmark dataset [1]. The results show that our model performs better than the state-of- the-art approach [1], while the model is more efficient in inference.

Our contributions can be summarized as follows:

1) We propose a novel Hidden CRF model for predicting underlying labels based on the sequential data. For each temporal segment, we exploit the full connectivity among observations, latent variables, and the target variables, from which we can avoid making inappropriate conditional independence assumptions.
2) We show an efficient way of applying exact inference in our graph. By collapsing the latent states and the target states, our graphical model can be considered as a linear-chain structure. Applying exact inference under such a structure is very efficient.
3) Our software is open source and will be fully available for comparison.

Details of this work can be found in Appendix A.

3 Recognition of High-level Activities

Recently, there has been a considerable amount of work focusing on graphical models for human activity recognition. Notably, Hu et al. [3] use latent variables to exploit sub-level semantics over the activities, and their approach shows state-of-the-art results on a benchmark dataset. However, their work only allows activities to have very short duration. For real tasks in HRI, it is desirable to recognize high-level activities that have a longer duration.

We distinguish between sub-level activities and high-level activities as follows. The sub-level activities are defined as the atomic actions that relate to a single object in the environment, e.g. reaching, placing, opening, closing, etc. Most of these sub-level activities are completed in a relatively short time. In contrast, high-level activities usually refer to a whole sequence that is composed of different sub-level activities. For example, “microwaving food” is a high-level activity and it can be decomposed into a number of sub-level activities such as opening the microwave, reaching for the food, moving food, placing food, and closing the microwave.

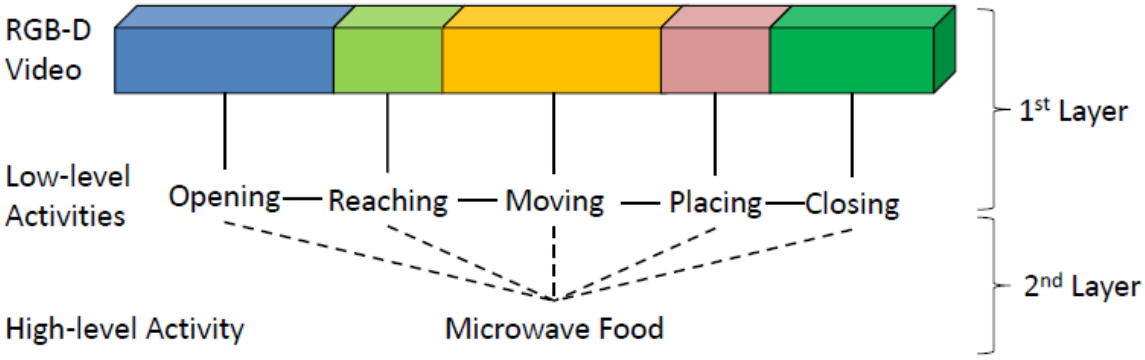


Figure 2: An illustration of our approach

The task of recognizing sub-level activities is usually formulated as a sequential prediction problem, see Figure 2. The RGB-D video is firstly divided into smaller video segments, so that each segment contains more or less one low-level activity. This can be done either by manual annotation or by automated temporal segmentation based on appearance. Spatial-temporal features are extracted for each temporal segment. Based on the input features, we need to predict the most likely underlying sequence of low-level activities. The predicted sub-level activities can be viewed as the input for inferring high-level activities. In this paper, we propose an approach for learning high-level human activities. Our approach can be decomposed into two layers, i.e. recognition of sub-level activities and inferring high-level activities based on the sub-level activities. For the first layer, we model the correlation of sub-level activities between two consecutive video segments. Similar to [3], we use latent variables to exploit the underlying semantics among sub-level activities. For example, the sublevel activity closing may refer to closing a bottle or closing the microwave. Although the two activities share the same label closing, they belong to different sub-types of closing. The latent variables are able to capture such a difference and are able to model the rich

variations of the sub-level activities. For recognizing high-level activities, we treat the output sub-level activities from the first layer as the input in the second layer, and the high-level activities are predicted based on the sequence of sub-level activities. We use a max-margin approach for learning the parameters of the model. Benefiting from the discriminative framework, our method does not need to model the correlation between the input data, thus providing us with a natural way for data fusion.

Details of this work can be found in Appendix B.

4 Conclusion and Future Work

The novel model for activity recognition was tested on a standard benchmark data set (CAD-120 benchmark). Experimental results on this data set show that our model outperforms the state-of-the-art approach by over 5% in both precision and recall, while our model is more efficient in computation.

We present a two-layered approach that can recognize low-level and high-level human activities simultaneously. We investigate the effect of using latent variables, segmentation methods, as well as different feature representations. Our results show that the two-layered approach performs better than the approach with only a single layer. Our model is also shown to outperform the state-of-the-art on the same dataset. Currently, our approach only uses the RGB-D videos for activity recognition. In our future work, we would like to fuse different cues, e.g. human locations [11], human identities [12] and ambient sensors [13], for robust estimation of human activities.

The systems are evaluated on the benchmark dataset in order to compare with the state-of-the-art approaches. We are working on creating a new benchmark dataset in Troyes. The dataset will incorporate data from ambient sensors, robot sensors, and overhead cameras, therefore it can be used for multi-dimensional research. The dataset will be recorded with real elderly people and will be annotated by the soft labeling method that we have proposed.

5 References

[1] H. S. Koppula, R. Gupta, and A. Saxena, “Learning Human Activities and Object Affordances from RGB-D Videos,” *Int. J. Robot. Res.*, vol. 32, no. 8, pp. 951–970, 2013.

[2] H. Koppula and A. Saxena, “Anticipating human activities using object affordances for reactive robotic response,” in *Proc. Robotics Science and Systems (RSS)*, 2013.

[3] N. Hu, G. Englebienne, Z. Lou, and B. Kröse, “Learning Latent Structure for Activity Recognition,” in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[4] C. Zhu and W. Sheng, “Human Daily Activity Recognition in Robot-assisted Living using Multi-sensor Fusion,” in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 2154–2159.

[5] J. Sung, C. Ponce, B. Selman, and A. Saxena, “Unstructured human activity detection from rgbd images,” in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 842–849.

[6] T. L. M. van Kasteren, G. Englebienne, and B. Kröse, “Activity recognition using semi-markov models on real world smart home datasets,” *J. Ambient Intell. Smart Environ.*, vol. 2, no. 3, pp. 311–325, 2010.

[7] N. Hu, G. Englebienne, and B. Kröse, “Posture Recognition with a Top-view Camera,” in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 2152–2157.

[8] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, “Hidden Conditional Random Fields,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1848–1852, 2007.

[9] L. Maaten, M. Welling, and L. K. Saul, “Hidden-Unit Conditional Random Fields,” in *Proc. International Conference on Artificial Intelligence and Statistics*, 2011, pp. 479–488.

[10] Y. Wang and G. Mori, “Max-margin hidden conditional random fields for human action recognition,” in *Proc. IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 872–879.

[11] N. Hu, G. Englebienne, and B. Kröse, “Bayesian Fusion of Ceiling Mounted Camera and Laser Range Finder on a Mobile Robot for People Detection and Localization,” in *IROS workshop on Human Behavior Understanding*, 2012, vol. 7559, pp. 41–51.

[12] N. Hu, R. Bormann, T. Zwölfer, and B. Kröse, “Multi-User Identification and Efficient User Approaching by Fusing Robot and Ambient Sensors,” in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[13] T. Van Kasteren, A. Noulas, G. Englebienne, and B. Kröse, “Accurate activity recognition in a home setting,” in *Proc. International Conference on Ubiquitous Computing*, 2008, pp. 1–9.

Appendix A

Learning Latent Structure for Activity Recognition*

Ninghang Hu[1], Gwenn Englebienne[1], Zhongyu Lou[1] and Ben Kröse[1,2]

***Abstract*— We present a novel latent discriminative model for human activity recognition. Unlike the approaches that require conditional independence assumptions, our model is very flexible in encoding the full connectivity among observations, latent states, and activity states. The model is able to capture richer class of contextual information in both state-state and observation-state pairs. Although loops are present in the model, we can consider the graphical model as a linear-chain structure, where the exact inference is tractable. Thereby the model is very efficient in both inference and learning. The parameters of the graphical model are learned with the Structured-Support Vector Machine (Structured-SVM). A data-driven approach is used to initialize the latent variables, thereby no hand labeling for the latent states is required. Experimental results on the CAD-120 benchmark dataset show that our model outperforms the state-of-the-art approach by over 5% in both precision and recall, while our model is more efficient in computation.**

I. Introduction

Robotic companions to help people in their daily life are currently a widely studied topic. In Human-Robot Interaction (HRI) it is very important that the human activities are recognized accurately and efficiently. In this paper, we present a novel graphical model for human activity recognition.

The task of activity recognition is to find the most likely underlying activity sequence based on the observations generated from the sensors. Typical sensors include ambient cameras, contact switches, thermometers, pressure sensors, and the sensors on the robot, *e.g.* RGB-D sensor and Laser Range Finder.

Probabilistic Graphical Models have been widely used for recognizing human activities in both robotics and smart home scenarios. The graphical models can be divided into two categories: generative models [1], [2] and discriminative models [3], [4], [5]. The generative models require making assumptions on both the correlation of data and on how the data is distributed given the activity state. The risk is that the assumptions may not reflect the true attributes of the data. The discriminative models, in contrast, only focus on modeling the posterior probability regardless of how the data are distributed. The robotic and smart environment scenarios are usually equipped with a combination of multiple sensors. Some of these sensors may be highly correlated, both in the temporal and spatial domain, *e.g.* a pressure sensor on the mattress and a motion sensor above the bed. In these scenarios, the discriminative models provide us a natural way of data fusion for human activity recognition.

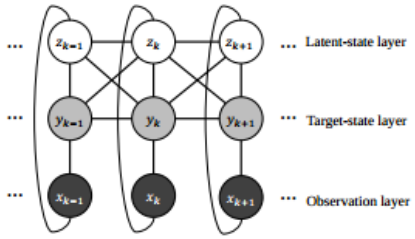


Fig. 1. The proposed graphical model. Nodes that represent the observations x are rendered in black, and they are observed both in training and testing. Grey nodes y are only observed during training but not testing, and they represent the target labels to be predicted, *e.g.* activity labels. White nodes z refer to the latent variables, which are unknown either in training or testing. Note that x_k, y_k, z_k are fully connected in our model, and also for nodes of transition.

The linear-chain Conditional Random Field (CRF) is one of the most popular discriminative models and has been used for many applications. Linear-chain CRFs are efficient models because the exact inference is tractable. However, they are limited in the way that they cannot capture the intermediate structures within the target states [6]. By adding an extra layer of latent variables, the model allows for more flexibility and therefore it can be used for modeling more complex data. The names of these models are interchangeable in the literature, such as Hidden-Unit CRF [7], Hidden-state CRF [6] or Hidden CRF [8].

In this paper, we present a latent CRF model for human activity recognition. For simplicity, we use *latent variables* to refer to the augmented hidden layer, as they are unknown either in training or testing. Intuitively, one can imagine that the latent variables represent subtypes of the activities. *e.g.* For the activity "opening", using latent variables we are able to model the difference between "opening a bottle" and "opening a door". The *target variables*, which is observed during training but not testing, represent the target states that we would like to predict, *e.g.* the activity labels. See Fig. 1 for the graphical model and the difference between latent variables and target variables. We evaluate the model using the RGB-D data from the benchmark dataset [3]. The results show that our model performs better than the state-of-the-art approach [3], while the model is more efficient in inference.

The contributions of this paper can be summarized as follows: We propose a novel Hidden CRF model for predicting underlying labels based on the sequential data. For each temporal segment, we exploit the full connectivity among

*The research has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 287624.

[1] N. Hu, G. Englebienne, Z. Lou and B. Kröse are with Intelligent System Lab Amsterdam, University of Amsterdam, 1098XH Amsterdam, The Netherlands {n.hu,g.englebienne,z.lou,b.j.a.krose}@uva.nl

[2] B. Kröse is also with the Amsterdam University of Applied Science

observations, latent variables, and the target variables, from which we can avoid making inappropriate conditional independence assumptions. We show an efficient way of applying exact inference in our graph. By collapsing the latent states and the target states, our graphical model can be considered as a linear-chain structure. Applying exact inference under such a structure is very efficient. Our software is open source and will be fully available for comparison[1].

II. Related Work

Human activity recognition has been extensively studied in recent decades. Different types of graphical models have been applied to solve the problem, *e.g.* Hidden Markov Models (HMMs) [1], [2], Dynamic Bayesian Networks (DBNs) [9], linear-chain CRFs [10], loopy CRFs [3], Semi-Markov Models [4], and Hidden CRFs [11], [8].

As has been discussed in the introduction, the discriminative models are more suitable for data fusion tasks which are very common in HRI applications, where many different sensors are used. Here we focus on reviewing the most related work that uses discriminative models for activity recognition.

Recently Koppula et al. [3] presented a model for the temporal and spatial interactions between human and objects in loopy CRFs. More specifically, they built a model that has two types of nodes to represent sub-activity labels of the human and the object affordance labels of the objects. Human nodes and objects nodes within the same temporal segment are fully connected. Over time, the nodes are transited to the nodes with the same type. The results show that by modeling the human-object interaction, their model outperforms the earlier work in [2] and [12]. For inference in the loopy graph, they solve it as a quadratic optimization problem using the graph-cut method [13]. Their inference method, however, is less efficient compared with the exact inference in a linear-chain structure as the graph cut method takes multiple iterations before convergence, and usually more iterations are preferred to ensure of a good solution.

Other work [14] augments an additional layer of latent variables to the linear-chain CRFs. They explicitly model the new latent layer to represent the duration of activities. In contrast with [3], Tang et al. [14] solve the inference problem by reforming the graph into a set of cliques, so that the exact inference can be solved efficiently using dynamic programming. In their model, the latent variables and the observation are assumed to be conditionally independent given the target states.

Our work is different from the previous approaches in both the graphical model and the efficiency of inference. Firstly, similar to [14], our model also uses an extra latent layer. But instead of explicitly modeling what the latent variables are, we learn the latent variables directly from the data. Secondly, we do not make conditional independence assumptions between the latent variables and the observations. Instead, we add one extra edge between them to make the local graph fully connected. Thirdly, although our graph also presents a lot of loops as in [3], we are able to transform the cyclic graph into a linear-chain structure where the exact inference is tractable. The exact inference in our graph only needs two passes of messages across the linear chain structure which is much more efficient than [3]. Finally, we model the interaction between the human and the objects at the feature level, instead of modeling the object affordance as target states. In such a way, the parameters are learned to be directly optimized for activity recognition rather than making the joint estimation of both object affordance and the human activity. As we apply a data-driven approach to initialize the latent variables, hand labeling of the object affordance is not necessary in our model. Our results show that the model outperforms the state-of-the-art approaches on the CAD120 dataset [3].

[1]The source code will be available at https://github.com/ninghang/activity_recognition.git

III. Model

The graphical model of our proposed system is illustrated in Fig. 1. Let $\boldsymbol{x} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\}$ be the sequence of observations, where K is the total number of temporal segments in the video. Our goal is to predict the most likely underlying activity sequence $\boldsymbol{y} = \{y_1, y_2, \ldots, y_K\}$ based on the observations. We define $\boldsymbol{z} = \{z_1, z_2, \ldots, z_K\}$ to be the latent variables in the model. We assume there are N_y activities to be recognized and N_z latent states.

Each observation $\boldsymbol{x}_k$ itself is a feature vector within the segment k. The form of $\boldsymbol{x}_k$ is quite flexible. It can be collections of data from different sources, *e.g.* simple sensor readings, human locations, human pose, object locations. Some of these observations may be highly correlated with each other, *e.g.* the wearable accelerate meters and the motion sensors. Thanks to the discriminative nature of our model, we do not need to model such correlation among the observations.

A. Objective Function

Our model contains three types of potentials that in together form the objective function.

The first potential measures the score of seeing an observation $\boldsymbol{x}_k$ with a joint-state assignment (z_k, y_k). We define $\Phi(\boldsymbol{x}_k)$ to be the function that maps the input data into the feature space. $\boldsymbol{w}$ is the vector of parameters in our model.

$$\psi_1(y_k, z_k, \boldsymbol{x}_k; \boldsymbol{w}_1) = \boldsymbol{w}_1(y_k, z_k) \cdot \Phi(\boldsymbol{x}_k) \quad (1)$$

This potential models the full connectivity among y_k, z_k and $\boldsymbol{x}_k$, avoiding making any conditional independence assumptions. It is more accurate to have such a structure since z_k and $\boldsymbol{x}_k$ may not be conditionally independent over a given y_k in many cases. To make it more intuitive, one could imagine that y_k refers to the activity drinking coffee and z_k defines the progress level of drinking. The activity drinking coffee starts with human grasping the coffee cup $(z_k = 1)$, then drinking $(z_k = 2)$, and then putting the cup back $(z_k = 3)$. Knowing it is a drinking activity, the

observation $\boldsymbol{x}_k$ varies largely over different progress level z_k.

The second potential measures the score of coupling y_k with z_k. It can be considered as either the bias entry of (1) or the prior of seeing the joint state (y_k, z_k).

$$\psi_2(y_k, z_k; \boldsymbol{w}_2) = \boldsymbol{w}_2(y_k, z_k) \tag{2}$$

The third potential characterizes the transition score from the joint state (y_{k-1}, z_{k-1}) to (y_k, z_k). Comparing with the normal transition potentials [8], our model leverages the latent variable z_k for modeling richer contextual information over consecutive temporal segments. Not only does our model contain the transition between states y_k, but it also captures the sub-level context using the latent variables. Intuitively, our model is able to capture the fact that the start of reading a newspaper is more likely to be preceded by the end of the drinking activity rather than the middle part of the drinking activity.

$$\psi_3(y_{k-1}, z_{k-1}, y_k, z_k; \boldsymbol{w}_3) = \boldsymbol{w}_3(y_{k-1}, z_{k-1}, y_k, z_k) \tag{3}$$

Summing all potentials over the whole sequence, we can write the objective function of our model as follows

$$\begin{aligned} F(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{x}; \boldsymbol{w}) = &\sum_{k=1}^{K} \{\boldsymbol{w}_1(y_k, z_k) \cdot \Phi(\boldsymbol{x}_k) + \boldsymbol{w}_2(y_k, z_k)\} \\ &+ \sum_{k=2}^{K} \boldsymbol{w}_3(y_{k-1}, z_{k-1}, y_k, z_k) \end{aligned} \tag{4}$$

The objective function evaluates the matching score between the joint states $(\boldsymbol{y}, \boldsymbol{z})$ and the input $\boldsymbol{x}$. The score equals to the un-normalized joint probability in the log space. The objective function can be rewritten into a more general linear form $F(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{x}; \boldsymbol{w}) = \boldsymbol{w} \cdot \Psi(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{x})$. Therefore the model is in the class of the log-linear model.

Note that it is not necessary to model the latent variables explicitly, but rather the latent variables can be learned automatically from the training data. Theoretically, the latent variables can represent any form of data, *e.g.* time duration, action primitives, as long as it can help with solving the task. Optimization of the latent model, however, may converge to a local minimum. The initialisation of the random variables is therefore of great importance. We compare three initialization strategies in this paper. Details of the latent variable initialization will be discussed in Section VI-D.

One may notice that our graphical model has many loops, which in general makes the exact inference intractable. Since our graph complies with the semi-Markov property, next, we will show that how we benefit from such a structure for efficient inference and learning.

IV. Inference

Given the graph and the parameters, the inference is to find the most likely joint states $\boldsymbol{y}$ and $\boldsymbol{z}$ that maximizes the objective function.

$$(\boldsymbol{y}^\star, \boldsymbol{z}^\star) = \underset{(\boldsymbol{y}, \boldsymbol{z}) \in \mathcal{Y} \times \mathcal{Z}}{\operatorname{argmax}} \; F(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{x}; \boldsymbol{w}) \tag{5}$$

Generally, solving (5) is an NP-hard problem that requires evaluating the objective function over an exponential number of state sequences. Exact inference is usually preferable as it is guaranteed to find the global optimum. However, the exact inference usually can only be applied efficiently when the graph is acyclic. In contrast, approximate inference is more suitable for loopy graphs, but may take longer to converge and is likely to find a local optimum. Although our graph contains loops, we show that we can transform the graph into a linear-chain structure, in which the exact inference becomes tractable. If we collapse the latent variable z_k with the activity state y_k into a single node, the edges between z_k and y_k become the internal factor of the new node and the transition edges collapse into a single transition edge. This results in a typical linear-chain CRF, where the cardinality of the new nodes is $N_y \times N_z$. In the linear-chain CRF, the exact inference can be performed efficiently using dynamic programming [15].

Using the chain property, we can write the following recursion for computing the maximal score over all possible assignments of $\boldsymbol{y}$ and $\boldsymbol{z}$.

$$\begin{aligned} V_k(y_k, z_k) = &\boldsymbol{w}_1(y_k, z_k) \cdot \phi(\boldsymbol{x}_k) + \boldsymbol{w}_2(y_k, z_k) \\ &+ \max_{(y_{k-1}, z_{k-1}) \in \mathcal{Y} \times \mathcal{Z}} \{\boldsymbol{w}_3(y_{k-1}, z_{k-1}, y_k, z_k) \\ &+ V_{k-1}(y_{k-1}, z_{k-1})\} \end{aligned} \tag{6}$$

Knowing the optimal assignment at K, we can track back the best assignment in the previous time step $K-1$. The process keeps going until all $\boldsymbol{y}^\star$ and $\boldsymbol{z}^\star$ have been assigned, *i.e.* the inference problem in (5) is solved.

Computing (6) once involves $O(N_y N_z)$ computations. In total, (6) needs to be evaluated for all possible assignments of (y_k, z_k), so that it is computed $N_y N_z$ times. The total computational cost is, therefore, $O(N_y^2 N_z^2 K)$. Such computation is manageable when $N_y N_z$ is not very large, which is usually the case for the tasks of activity recognition.

Next, we show how we can learn the parameters using the max-margin approach.

V. Learning

We use the max-margin approach for learning the parameters in our graphical model. The observation sequences and ground-truth activity labels are given during training $(\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \ldots, (\boldsymbol{x}_N, \boldsymbol{y}_N)$. The latent variables $\boldsymbol{z}$ are unknown from the training data. The goal of learning is to find the parameters $\boldsymbol{w}$ that minimize the loss between the predicted activities and the ground-truth labels. A regularization term is used to avoid over-fitting.

$$\min_{\boldsymbol{w}} \left\{ \frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{i=1}^{N} \Delta(\boldsymbol{y}_i, \hat{\boldsymbol{y}}) \right\} \tag{7}$$

where C is a normalization constant and $\Delta(\boldsymbol{y}_i, \hat{\boldsymbol{y}})$ measures the loss between the ground-truth and the prediction. The loss function returns zero when the prediction is the same as the ground-truth, and counts the number of disagreed elements otherwise. $\hat{\boldsymbol{y}}$ is the most likely activity sequence computed from (5) based on $\boldsymbol{x}_i$.

Optimizing (7) directly is not possible as the loss function involves computing the argmax in (5). Following [16] and [17], we substitute the loss function in (7) by the margin rescaling surrogate which serves as an upper-bound of the loss function.

$$\min_{\boldsymbol{w}}\{\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\max_{(\boldsymbol{y},\boldsymbol{z})\in\mathcal{Y}\times\mathcal{Z}}[\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + F(\boldsymbol{x}_i,\boldsymbol{y},\boldsymbol{z};\boldsymbol{w})] - C\sum_{i=1}^{n}\max_{\boldsymbol{z}\in\mathcal{Z}}F(\boldsymbol{x}_i,\boldsymbol{y}_i,\boldsymbol{z};\boldsymbol{w})\} \quad (8)$$

The second term in (8) can be solved using the augmented inference, *i.e.* by plugging in the loss function as an extra factor in the graph, the term can be solved in the same way as the inference problem using (5). Similarly, the third term of (8) can be solved by adding $\boldsymbol{y}_i$ as the evidence into the graph and then applying inference using (5). As the exact inference is tractable in our graphical model, both of the terms can be computed very efficiently.

Note that (8) is the summation of a convex and a concave function. This can be solved with the Concave-Convex Procedure (CCCP) [18]. By substituting the concave function with its tangent hyperplane function, which serves as an upper-bound of the concave function, the concave term is changed into a linear function. Thereby (8) becomes convex again.

We can rewrite (8) in the form of minimizing a function subject to a set of constraints by adding slack variables

$$\min_{\boldsymbol{w},\boldsymbol{\xi}}\left\{\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\xi_i\right\} \quad (9)$$
$$s.t.\ \forall i\in\{1,2,\ldots,n\},\forall\boldsymbol{y}\in\mathcal{Y}:$$
$$F(\boldsymbol{x}_i,\boldsymbol{y}_i;\boldsymbol{w}) - F(\boldsymbol{x}_i,\boldsymbol{y};\boldsymbol{w}) \geq \Delta(\boldsymbol{y}_i,\boldsymbol{y}) - \xi_i$$

Note that there are exponential number of constraints in (9). This can be solved by the cutting-plane method [19].

Another intuitive way to understand the CCCP algorithm is to consider it as solving the learning problem with incomplete data using Expectation-Maximization (EM) [20]. In our training data, the latent variables are not given. We can start by initializing the latent variables. Once we have the latent variables, the data become complete. Then we can use the standard Structured-SVM to learn the model parameters (M-step). After that, we can update the latent states again using the parameters that are learned (E-step). The iteration continues until convergence.

The CCCP algorithm decreases the objective function in every iteration. However, it cannot guarantee of finding the global optimum. To avoid of being trapped in the local minimum, the latent variables need to be carefully initialized. In this paper, we present three different initialization strategies, and details will be presented in Section VI-D.

Note that the inference algorithm is extensively used in learning. As we are able to compute the exact inference by transforming the loopy graph into a linear-chain graph, our learning algorithm is much faster and more accurate compared with the other approaches with approximate inference.

VI. Experiments

Our system is built upon three parts, the graphical model, the inference part and the learning part. We construct the graphical model and build the CCCP algorithm in Matlab. For exact inference, we adopt the inference engine from libDAI [21]. For learning, we take the Structured SVM framework provided by [16]. We compare the results with the state-of-the-art approach in [3].

A. Data

We evaluate our model on the CAD-120 dataset [3]. The dataset has 120 RGB-D videos with 4 subjects performing daily life activities. Each video is annotated with one high-level activity label and a sequence of sub-activity labels. The ground-truth of the segments and object affordance labels are also provided. In this paper, we use the sub-activity labels for evaluation. But our model can be easily extended into a hierarchical approach that can recognize higher-level activities, which will be reported in our next paper. As in [3], we use the ground-truth segments that are provided by the dataset.

For comparison, the same input features[2] are used as in [3]. The features are human skeleton features $\phi_a(\boldsymbol{x}_k) \in \mathbb{R}^{630}$, object features $\phi_o(\boldsymbol{x}_k) \in \mathbb{R}^{180}$, object-object interaction features $\phi_{oo}(\boldsymbol{x}_k) \in \mathbb{R}^{200}$, object-subject relation features $\phi_{oa}(\boldsymbol{x}_k) \in \mathbb{R}^{400}$, and the temporal objection and subject features $\phi_t(\boldsymbol{x}_k) \in \mathbb{R}^{200}$. These features are concatenated into a single feature vector, which is considered as the observation of one sub-activity segment, *i.e.* $\Phi(\boldsymbol{x}_k)$.

B. Evaluation Criteria

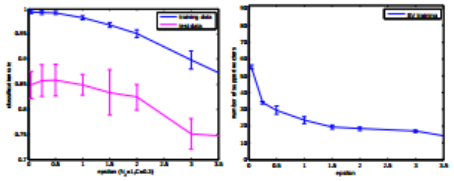
Our model is evaluated with 4-fold cross-validation. The folds are split based on the 4 subjects, *i.e.* the model is trained on videos of 3 persons and test on a *new person*. Each cross-validation is run for 3 times. To check the generalization of our model across different data, the results are averaged across the folds. In this paper, accuracy (classification rate), precision and recall are reported for comparing the results. In the CAD-120 dataset, more than half the instances are "reaching" and "moving". Therefore we consider precision and recall to be relatively better evaluation criteria than accuracy, as they remain meaningful despite class imbalance.

C. Baseline

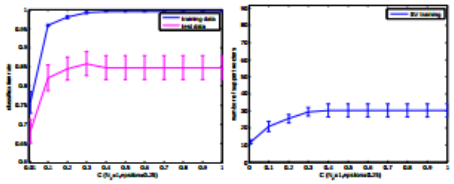
Our baseline approach uses only one latent state in our model ($N_z = 1$), which is equivalent to a linear-chain CRF. The parameters of the baseline model are learned with the standard Structured-SVM. We use the margin rescaling surrogate as the loss and L1-norm for the slacks. For optimization we use the 1-slack algorithm (primal) as being described in [22].

We apply a grid search for the best SVM parameters of C and ϵ. C is the normalization constant that is the trade-off between model complexity and classification loss. ϵ defines the stop threshold of optimization. When ϵ is small, the

[2]Input features can be downloaded from http://pr.cs.cornell.edu/humanactivities/data/features.tar



(a) $C = 0.30$, change ϵ



(b) $\epsilon = 0.25$, change C

Fig. 3. Another view of the grid search for the best C and ϵ. (a) shows the change of classification rate over ϵ when C is fixed to 0.3. When ϵ is small, a large number of support vectors is added and the model overfits. When ϵ is too large, the model is underfitting and the iterations stop too early, with too few support vectors. (b) shows the change of classification rate over C when ϵ is fixed to 0.25. When C is small, the learning algorithm tries to find a model as simple as possible, so that the performance is very low. When C is very large, the model overfits and the performance drops.

learning process takes longer time to converge and the trained model contains more support vectors. We show results of the grid search in Fig. 2. In Fig. 3 we show the curve of accuracy when keeping one of the parameters fixed.

Based on these results, we choose $C = 0.3$ and $\epsilon = 0.25$ for our experiments.

D. Initialize Latent Variables

In the our latent model, we choose the same C and ϵ as in the linear-chain CRF. Parameters of the model are initialized as zeros. To initialize the latent states, we adopt three different initialization strategies. a) Random initialization. b) A data-driven approach. We apply clustering on the input data $\boldsymbol{x}$. The number of clusters is set to be the same as the number of latent states. We run K-means for 10 times. Then we choose the best clustering results that with the minimal within-cluster distances. The labels of the clusters are assigned as the initial latent states. c) Object affordance. The object affordance labels are provided by the CAD120 dataset, which are used for training in [3]. We apply the K-means clustering upon the affordance labels. As the affordance labels are categorical, we use 1-of-N encoding to transform the affordance labels into binary values for clustering.

E. Results

Table I compares the activity recognition performance between our model and the state-of-the-art approach in [3]. We evaluate the model with different number of latent states, *i.e.* latent-2, latent-3 and latent-4, as well as the different initialization strategies, *i.e.* random, data-driven and affordance.

TABLE I
RESULTS OF ACTIVITY RECOGNITION

	Accuracy	Precision	Recall	F-score
Koppula, et al. [3]	86.0 ± 0.9	84.2 ± 1.3	76.9 ± 2.6	80.4 ± 1.5
latent-1 linear CRF	85.7 ± 2.9	86.4 ± 6.1	82.4 ± 4.0	82.6 ± 6.2
latent-2 random	84.0 ± 2.8	85.6 ± 4.6	79.5 ± 5.4	80.1 ± 6.5
latent-2 data-driven	$\mathbf{87.0 \pm 1.9}$	$\mathbf{89.2 \pm 4.6}$	$\mathbf{83.1 \pm 2.4}$	$\mathbf{84.3 \pm 4.7}$
latent-2 affordance	$\mathbf{87.0 \pm 2.1}$	$\mathbf{88.3 \pm 4.3}$	$\mathbf{84.0 \pm 3.2}$	$\mathbf{84.3 \pm 5.1}$
latent-3 random	83.1 ± 2.2	86.1 ± 4.5	76.3 ± 4.8	78.1 ± 6.1
latent-3 data-driven	86.0 ± 1.9	87.2 ± 2.9	82.3 ± 2.4	82.9 ± 4.2
latent-3 affordance	86.0 ± 2.0	88.0 ± 4.6	81.5 ± 3.4	82.1 ± 4.8
latent-4 random	82.8 ± 3.2	85.9 ± 5.0	76.3 ± 5.6	77.5 ± 6.9
latent-4 data-driven	85.9 ± 1.7	86.8 ± 2.7	82.4 ± 2.0	82.8 ± 3.7
latent-4 affordance	85.7 ± 1.6	86.4 ± 2.8	81.7 ± 2.9	82.0 ± 3.6

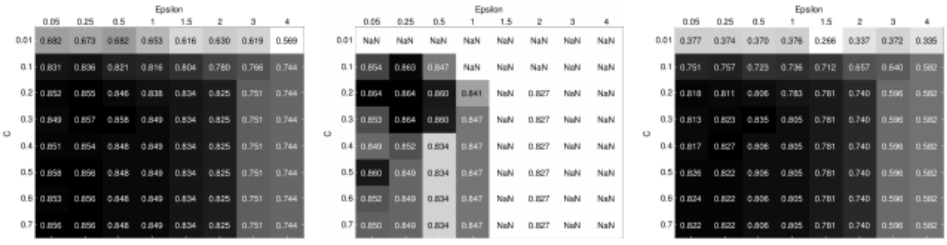
We show that with the optimal SVM parameters, the baseline performs better on the precision and recall compared with [3], but worse on the accuracy. This is because the baseline does not model the object affordance as target variables, and the parameters are optimized directly for minimizing the loss in activity recognition. The other reason is that the baseline model follows a linear-chain structure, and it is guaranteed to find the global optimal solution.

By adding the latent variables, our model can achieve better results than the baseline, but only when the latent variables are properly initialized. When the latent variables are randomly initialized, the average performance is much worse in most of the cases and shows a large variance as it most likely to have converged to a local minimum. We note that the data-driven initialization (clustering on $\boldsymbol{x}$) performs as good as the initialization with the hand-labeled object affordances.

We also compare the model when different numbers of latent states are used. We obtain better performance when we use only 2 latent states instead 3 or 4. This is partly because there are more parameters to be tuned when the model contains more latent states. The other reason is that the model may be too complex and overfits the data. Therefore choosing the number of latent states is also data related. If we use a more complex dataset, more latent states need to be used.

Fig. 4 shows the confusion matrix of activity classification. We can see that higher values present on the diagonal of the confusion matrix, and they represent the activities that are correctly classified. The most difficult classes are eating and scrubbing. Eating is sometimes confused with the drinking, and scrubbing is likely to be confused with reaching, drinking and placing.

Our best performance is obtained when we use 2 latent states and the model is initialized by clustering on the input data. We get 89.2% on the average precision and 83.1% on the average recall, which outperforms the state-of-the-art by over 5% on both precision and recall. We believe the performance can be further improved if we apply grid search for the optimal learning parameters of the latent-state model.



(a) Average Accuracy (b) Average Precision (c) Average Recall

Fig. 2. Performance of the baseline approach ($N_z = 1$). We apply a grid search to choose the best C and ϵ. The results are averaged on multiple runs of 4-fold cross-validation. The nan entry in (b) means that at least one of the classes gets no positive detection. Based on the grid search, we choose $C = 0.3$ and $\epsilon = 0.25$.

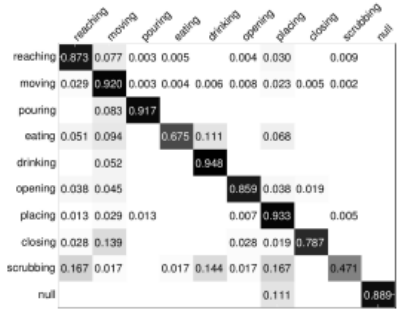


Fig. 4. Confusion matrix over different activity classes. Rows are ground-truth labels and columns are the detections. Each row is normalized to sum up to one, as one data object can only be associated with a single class label.

VII. Conclusion and Future Work

In this paper, we present a novel Hidden-state CRF model for human activity recognition. We use the latent variables to exploit the underlying structures of the target states. By making the observation and state nodes fully connected, the model do not require any conditional independence assumption between latent variables and the observations. The model is very efficient in that the inference algorithm is applied to a linear-chain structure. The results show that the proposed model outperforms the state-of-the-art approach. The model is very general that it can be easily extended for other prediction tasks on sequential data.

References

[1] C. Zhu and W. Sheng, "Human daily activity recognition in robot-assisted living using multi-sensor fusion," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2009, pp. 2154–2159.

[2] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 842–849.

[3] H. Koppula and A. Saxena, "Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation," *International Journal of Robotics Research (IJRR)*, 2013.

[4] T. van Kasteren, G. Englebienne, and B. J. Kröse, "Activity recognition using semi-markov models on real world smart home datasets," *Journal of Ambient Intelligence and Smart Environments*, vol. 2, no. 3, pp. 311–325, 2010.

[5] N. Hu, G. Englebienne, and B. Kröse, "Posture recognition with a top-view camera," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[6] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 29, no. 10, pp. 1848–1852, 2007.

[7] L. Maaten, M. Welling, and L. K. Saul, "Hidden-unit conditional random fields," in *International Conference on Artificial Intelligence and Statistics*, 2011, pp. 479–488.

[8] Y. Wang and G. Mori, "Max-margin hidden conditional random fields for human action recognition," in *Proc. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 872–879.

[9] Y.-c. Ho, C.-h. Lu, I.-h. Chen, S.-s. Huang, C.-y. Wang, L.-c. Fu, *et al.*, "Active-learning assisted self-reconfigurable activity recognition in a dynamic environment," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2009, pp. 1567–1572.

[10] D. L. Vail, M. M. Veloso, and J. D. Lafferty, "Conditional random fields for activity recognition," in *Proc. International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM, 2007, p. 235.

[11] S. B. Wang, A. Quattoni, L.-P. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," in *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 2006, pp. 1521–1527.

[12] B. Ni, P. Moulin, and S. Yan, "Order-preserving sparse coding for sequence classification," in *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 173–187.

[13] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing binary mrfs via extended roof duality," in *Proc. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2007, pp. 1–8.

[14] K. Tang, L. Fei-Fei, and D. Koller, "Learning latent temporal structure for complex event detection," in *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 1250–1257.

[15] R. Bellman, "Dynamic programming and lagrange multipliers," *The Bellman Continuum: A Collection of the Works of Richard E. Bellman*, p. 49, 1986.

[16] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," pp. 1453–1484, 2005.

[17] C.-N. Yu and T. Joachims, "Learning structural svms with latent variables," in *Proc. of International Conference on Machine Learning (ICML)*. ACM, 2009, pp. 1169–1176.

[18] A. L. Yuille and A. Rangarajan, "The concave-convex procedure (cccp)," *Advances in Neural Information Processing Systems (NIPS)*, vol. 2, pp. 1033–1040, 2002.

[19] J. E. Kelley, Jr, "The cutting-plane method for solving convex programs," *Journal of the Society for Industrial & Applied Mathematics*, vol. 8, no. 4, pp. 703–712, 1960.

[20] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. John Wiley & Sons, 2007, vol. 382.

[21] J. M. Mooij, "libDAI: A free and open source C++ library for discrete approximate inference in graphical models," *Journal of Machine Learning Research*, vol. 11, pp. 2169–2173, Aug. 2010.

[22] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural svms," *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009.

Appendix B

A Two-layered Approach to Recognize High-level Human Activities

Ninghang Hu[1], Gwenn Englebienne[1] and Ben Kröse[1,2]

***Abstract*— Automated human activity recognition is an essential task for Human Robot Interaction (HRI). A successful activity recognition system enables an assistant robot to provide precise services. In this paper, we present a two-layered approach that can recognize sub-level activities and high-level activities successively. In the first layer, the low-level activities are recognized based on the RGB-D video. In the second layer, we use the recognized low-level activities as input features for estimating high-level activities. Our model is embedded with a latent node, so that it can capture a richer class of sub-level semantics compared with the traditional approach. Our model is evaluated on a challenging benchmark dataset. We show that the proposed approach outperforms the single-layered approach, suggesting that the hierarchical nature of the model is able to better explain the observed data. The results also show that our model outperforms the state-of-the-art approach in accuracy, precision and recall.**

I. INTRODUCTION

Recently, there has been a considerable amount of work focusing on graphical models for human activity recognition [1], [2], [3], [4], [5], [6], [7]. Notably, Hu et. al. [7] use latent variables to exploit sub-level semantics over the activities, and their approach shows state-of-the-art results on a benchmark dataset. However, their work only allows activities to have very short duration. For real tasks in HRI, it is desirable to recognize high-level activities that have a longer duration.

We distinguish between sub-level activities and high-level activities as follows. The sub-level activities are defined as the atomic actions that relate to a single object in the environment, *e.g. reaching, placing, opening, closing*, etc. Most of these sub-level activities are completed in a relatively short time. In contrast, high-level activities usually refer to a whole sequence that is composed of different sub-level activities. For example, "microwaving food" is a high-level activity and it can be decomposed into a number of sub-level activities such as *opening* the microwave, *reaching* for the food, *moving* food, *placing* food, and *closing* the microwave.

The task of recognizing sub-level activities is usually formulated as a sequential prediction problem, see Fig. 1. The RGB-D video is firstly divided into smaller video segments, so that each segment contains more or less one low-level activity. This can be done either by manual annotation or by automated temporal segmentation based on appearance

The research has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 287624, and also partly from the EU-FP7 project MONARCH.

[1] N. Hu, G. Englebienne, and B. Kröse are with Intelligent System Lab Amsterdam, University of Amsterdam, 1098XH Amsterdam, The Netherlands {n.hu,g.englebienne,b.j.a.krose} @ uva.nl

[2] B. Kröse is also with the Amsterdam University of Applied Science.

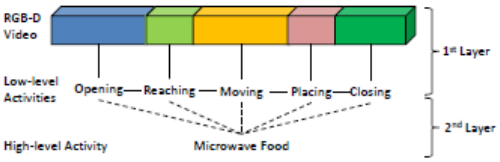


Fig. 1. An illustration of our approach. The input video is represented as a spatial-temporal volume. The video is discretized into multiple temporal segments for modeling, and spatial-temporal features are extracted at each temporal segment. In the first layer, sub-level activities are directly recognized from the input features with one atomic activity per segment. In the second layer, the high-level activities is described in terms of the sub-level activity sequence (dotted lines). Note that the video segments may not have the same length, thus a segmentation method needs to be applied.

features. Spatial-temporal features are extracted for each temporal segment. Based on the input features, we need to predict the most likely underlying sequence of low-level activities. The predicted sub-level activities can be viewed as the input for inferring high-level activities.

In this paper, we propose an approach for learning high-level human activities. Our approach can be decomposed into two layers, *i.e.* recognition of sub-level activities and inferring high-level activities based on the sub-level activities. For the first layer, we model the correlation of sub-level activities between two consecutive video segments. Similar to [7], we use latent variables to exploit the underlying semantics among sub-level activities. For example, the sub-level activity *closing* may refer to closing a bottle or closing the microwave. Although the two activities share the same label *closing*, they belong to different sub-types of closing. The latent variables are able to capture such a difference and are able to model the rich variations of the sub-level activities. For recognizing high-level activities, we treat the output sub-level activities from the first layer as the input in the second layer, and the high-level activities are predicted based on the sequence of sub-level activities. We use a max-margin approach for learning the parameters of the model[1]. Benefiting from the discriminative framework, our method does not need to model the correlation between the input data, thus providing us with a natural way for data fusion.

The rest of the paper is organized as follows. After reviewing the related work in Section II, we introduce the two layered approach in Section III. We present details of the experiments and we compare our model with the single layered approach in Section IV.

[1]Our source code will be updated at https://github.com/ninghang/activity_recognition

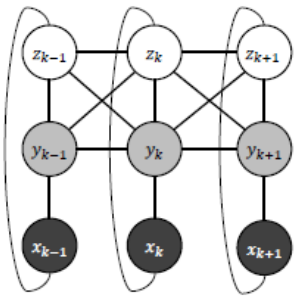
II. Related Work

Depending on the complexity and duration of activities, approaches of activity recognition can be separated into two categories [8], single-layered approaches and hierarchical approaches. The single-layer approaches [9], [10], [11], [12], [13], [14], [15], [16], [17] refer to the methods that are able to recognize human activities directly from the data, without defining any activity hierarchy. Usually these activities are both simple and short, so no higher-level layers are required. Typical activities of this category include walking, waiting, falling, jumping and waving. Nevertheless, in the real world, activities are not always as simple as these basic actions. For example, the activity of preparing some breakfast may consists of multiple sub-activities, such as opening a fridge, getting salad and making coffee. Before correctly estimating the high-level activity, the hierarchical models [4], [7], [18], [19], [5] need to recognize the sub-activities. Next, we review the relevant work that uses hierarchical models.

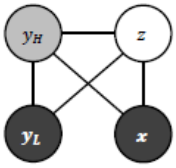
Sung et al. [1] proposed a hierarchical maximum entropy Markov model that detect high-level activities from RGB-D videos. They consider the sub-activities as hidden nodes which are learned implicitly. Recently, Koppula et al. [4] present an interesting approach that models both activities and objects affordance as random variables. These nodes are inter-connected to model object-object and object-human interactions. Nodes are connected across the segments to enable temporal interaction. Given a test video, the model jointly estimates both human activities and object affordance labels using a graph-cut algorithm. After the low-level activities are recognized, the high-level activities are estimated using a multi-class SVM. Hu et al. [7] encode the interactions between objects and humans at the feature level for recognizing low-level activities. They propose to add a latent layer to exploit underlying semantics between temporal segments. The inference algorithm of their model is very efficient as the graph can be viewed as a linear-chain structure. They were able to recognize low-level activities but did not consider the high-level activities. Our work is an extension of [7] for the task of recognizing high-level activities. Different from the multi-class SVM that is used in [4], we add a latent node to enrich the expressiveness of our model. Further, we evaluate the sub-level and high-level activities in a row and we experiment the effect of using different segmentation methods and feature representation in the context of recognizing high-level activities.

III. Approach

Our main goal is to predict the high-level activities based on RGB-D videos. The proposed two-layered approach is an extension of our previous work [7]. In the first layer, we divide the video into temporal segments and we adopt the approach in [7] to predict one sub-level activity per segment. In the second layer, we use the predicted sub-level activity labels as the input for estimating high-level activities.



(a) First layer



(b) Second layer

Fig. 2. The graphical models that are used in our two layered approach. (a) shows the graphical model for recognizing low-level activities $\boldsymbol{y}_k$ based on the observed video $\boldsymbol{x}_k$. (b) shows the graphical model for recognizing high-level activities y_H based on the recognized sub-level activity sequence $\boldsymbol{y}_L$ and occlusion features $\boldsymbol{x}$.

A. Recognizing Low-level Activities

In the first layer of our approach, we predict the low-level activity sequence based on the observed RGB-D video. For that we adopt a similar approach as in [7]. The video is firstly discretized into small segments based on the work of [4]. We use a sequence of random variables to model the sub-level activities in the video, with one sub-level activity node per segment. Adjacent sub-activity nodes are inter-connected to model the temporal interaction. Latent variables are appended for each temporal segment. The model is very flexible in encoding the full connectivity among observations, latent states, and activity states, therefore it is able to capture a richer class of contextual information in both state-state and observation-state interactions.

As loops are present in the model, parameter learning becomes difficult as exact inference is not tractable. We apply the same tricks as in [7] that we consider the activity node and latent node from the same segment as a single node, thereby the model is transformed into a linear-chain structure where many efficient inference solves can be applied. The graphical model is illustrated in Fig. 2(a). For more details, readers can refer to [7].

B. Recognize High-level Activities

The graphical model that recognizes high-level activities is illustrated in Fig. 2(b). Let $\boldsymbol{y}_L$ be a vector of low-level activities that are estimated from the first layer. Let $\boldsymbol{x}$ denote the global features extracted from the RGB-D videos. In this

paper, we consider it as the occlusion features that are used in [4]. Similar to [7], we append a latent node z to enrich the expressiveness of the model. Intuitively, we can think of that z models the sub-type of the high-level activity. Based on the observed low-level activities and the global features, our goal is to predict the most likely underlying high-level activity label y_H. Note that $\boldsymbol{x}$ and $\boldsymbol{y}_L$ are both observations in the second layer, and they are observed both in training and testing. In contrast, the high-level activity label $\boldsymbol{y}_H$ is observed only in training, and it is the target to be predicted during testing.

1) Feature Representation: We extract the n-gram features [20], *i.e.* $\phi(\boldsymbol{y}_L)$, from the low-level activity vector $\boldsymbol{y}_L$. Specifically, unigram (1-gram) is identical to the bag-of-words representation where values in the feature vector represent the occurrence of different words (sub-level activity labels in our case). As for bigram (2-gram) features, we compute the occurrence of pairwise activities that are contiguous in the sequence. Likewise, n-gram computes the frequencies of N contiguous activities. The advantage of using n-gram ($n > 1$) feature representation is that it encodes the temporal relation between two contiguous sub-level activities.

To further encoding the temporal semantics, we extract a set of occlusion features in a similar way as in [4]. We divide each video into 10 segments with equal length, and the features are computed as the fraction of objects that are occluded. These occlusion features are very helpful for disambiguating mirrored activities. For example, the "stacking" and "unstacking" are two mirrored high-level activities, and they contain exactly the same unigram features. In contrast, the occlusion features can capture the global changes of the sequence, therefore they are more capable of distinguish mirrored activities.

2) Potential Function: Our model consists of three potentials. The potentials are introduced separately, and after that we give the potential function of our model.

The first potential measures the score of seeing a sub-level activity sequence $\boldsymbol{y}_L$ with a joint-state assignment (y_H, z). $\boldsymbol{w}$ is the vector of parameters in our model. Note that we denote a sub-set of the parameter vector as $\boldsymbol{w}(y_H, z)$ where the parameters corresponds with y_H and z.

$$\psi_1(\boldsymbol{y}_L, y_H, z; \boldsymbol{w}_1) = \boldsymbol{w}_1(y_H, z) \cdot \phi(\boldsymbol{y}_L) \tag{1}$$

This potential models the interaction among y_H, z and $\boldsymbol{y}_L$. Since these nodes are fully connected, we can avoid making conditional independence assumptions among them. Traditional models assume that the latent component z and low-level activities $\boldsymbol{y}_L$ are conditionally independent once y_H is given. This is not true when y_H has a large variation, *e.g.* when performing the high-level activities "making cereal", people can either be sitting or standing. Although both of the two sub-type activities belong to the same activity "making cereal", they may differ significantly in the observed video. Using the latent variable, our model is able to capture such a difference, thus such a structure is more expressive and flexible for modeling human activities.

The second potential measures the score of coupling y_H with z. It can be considered as either the bias entry of (1) or the prior of seeing the joint state (y_H, z). Intuitively, this potential favors particular sub-types of the high-level activity rather than the other sub-types.

$$\psi_2(y_H, z; \boldsymbol{w}_2) = \boldsymbol{w}_2(y_H, z) \tag{2}$$

The third potential is similar to the first potential, and the main difference is that the observed variables in this potential are occlusion features $\boldsymbol{x}$ rather that sub-level activities $\boldsymbol{y}_L$. The third potential favors a particular assignment of the joint state (y_H, z) based on occlusion features.

$$\psi_3(\boldsymbol{y}_L, y_H, z; \boldsymbol{w}_3) = \boldsymbol{w}_3(y_H, z) \cdot \boldsymbol{x} \tag{3}$$

Based on the three potentials that we have defined, we can write the potential function of our model as

$$\begin{aligned} F(\boldsymbol{x}, \boldsymbol{y}_L, y_H, z; \boldsymbol{w}) =& \boldsymbol{w}_1(y_H, z) \cdot \phi(\boldsymbol{y}_L) + \boldsymbol{w}_2(y_H, z) \\ &+ \boldsymbol{w}_3(y_H, z) \cdot \boldsymbol{x} \end{aligned} \tag{4}$$

The above potential function measures the compatibility of certain joint states with all observations. The function returns a high value when the observations match with a particular high-level activity $\boldsymbol{y}_L$ and sub-type label z, and vice versa. The return value of the function can be considered as the un-normalized joint probability in log space. It is not hard to see that the potential function is a linear production of parameters and features. The objective function can be rewritten into a more general linear form $F(\boldsymbol{x}, \boldsymbol{y}_L, y_H, z; \boldsymbol{w}) = \boldsymbol{w} \cdot \Psi(\boldsymbol{x}, \boldsymbol{y}_L, y_H, z)$. Therefore the model is in the class of the log-linear model.

3) Inference Algorithm: The inference problem is to find the most likely high-level activities based on the observations, *i.e.* finding the joint state assignment $\boldsymbol{y}_H$ and $\boldsymbol{z}$ that maximizes the potential function. During inference, we assume all model parameters are known. The method of learning those parameters will be presented later in Section III-B.4.

Formally, our goal is to solve the following equation:

$$(y_H^*, z^*) = \underset{(y_H, z) \in \mathcal{Y}_H \times \mathcal{Z}}{\operatorname{argmax}} F(\boldsymbol{x}, \boldsymbol{y}_L, y_H, z; \boldsymbol{w}) \tag{5}$$

Since there are only a limited number of high-level activities and latent states, we can enumerate all possible joint state assignments and find the activity and latent state pair that holds the highest potential value. Such process can be paralleled and evaluating (5) only involves linear production, therefore the inference algorithm is very efficient and it can be solved in real time.

Now that we show the high-level activity can be efficiently predicted in our model, next, we present the method for learning the model parameters using a max-margin approach.

4) Learning Model Parameters: We use the max-margin approach for learning the parameters in our graphical model. The observations and ground-truth high-level activity labels are given during training $(\boldsymbol{x}^{(1)}, \boldsymbol{y}_L^{(1)}, y_H^{(1)}), (\boldsymbol{x}^{(2)}, \boldsymbol{y}_L^{(2)}, y_H^{(2)}), \ldots, (\boldsymbol{x}^{(N)}, \boldsymbol{y}_L^{(N)}, y_H^{(N)})$.

TABLE I
PERFORMANCE OF HIGH-LEVEL ACTIVITY RECOGNITION

METHOD	GROUNDTRUTH SEGMENTATION			APPEARANCE-BASED SEGMENTATION		
	ACCURACY	PRECISION	RECALL	ACCURACY	PRECISION	RECALL
SINGLE LAYER	74.2 ± 10.2	78.5 ± 9.4	73.3 ± 10.5	75.0 ± 10.7	79.0 ± 9.8	74.2 ± 11.0
KOPPULA, ET AL. [4]	84.7 ± 2.4	85.3 ± 2.0	84.2 ± 2.5	77.5 ± 4.1	80.1 ± 3.9	76.7 ± 4.2
OUR MODEL (UNIGRAM)	90.0 ± 2.9	92.8 ± 2.3	89.7 ± 3.0	79.0 ± 6.2	86.4 ± 4.9	78.8 ± 5.9
OUR MODEL (UNIGRAM+BIGRAM)	87.4 ± 5.1	92.4 ± 3.1	86.9 ± 5.2	75.0 ± 4.1	83.2 ± 5.4	74.6 ± 4.2

The superscript represents the index of different training examples. The latent variable z is unknown from the training data. The goal of learning is to find the optimal parameter set $\boldsymbol{w}$ that minimize the loss between the predicted high-level activities and the ground-truth labels. A regularization term is used to avoid over-fitting.

$$\min_{\boldsymbol{w}} \left\{ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N} \Delta(y_H^{(i)}, \hat{y}) \right\} \tag{6}$$

where C is a normalization constant and $\Delta(y_H^{(i)}, \hat{y})$ measures the zero-one loss between the ground-truth label and the prediction. $\hat{y}$ is the most likely activity sequence computed from (5). The loss function returns zero when the prediction is the same as the ground-truth, and counts the number of disagreed elements otherwise.

$$\Delta(y_H^{(i)}, \hat{y}) = \begin{cases} 1 & \text{if } \hat{y} \neq y_H^{(i)} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Optimizing (6) directly is not possible as the loss function involves computing the argmax in (5). Following [21] and [22], we substitute the loss function in (6) by the margin rescaling surrogate which serves as an upper-bound of the loss function.

$$\begin{aligned} \min_{w}\{\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \max_{y,z}[\Delta(y_H^{(i)}, y) + F(x^{(i)}, y_L^{(i)}, y, z; w)] \\ - C\sum_{i=1}^{n} \max_{z} F(x^{(i)}, y_L^{(i)}, y_H^{(i)}, z; w)\} \end{aligned} \tag{8}$$

The second term in (8) can be solved using the augmented inference, *i.e.* by plugging in the loss function as an extra factor in the graph, the term can be solved in the same way as the inference problem using (5). Similarly, the third term of (8) can be solved by adding $y_H^{(i)}$ as the evidence into the graph and then applying inference using (5). As the exact inference is tractable in our graphical model, both of the terms can be computed very efficiently.

Note that (8) is the summation of a convex and a concave function. This can be solved with the Concave-Convex Procedure (CCCP) [23]. By substituting the concave function with its tangent hyperplane function, which serves as an upper-bound of the concave function, the concave term is changed into a linear function. Thereby (8) becomes convex again.

We can rewrite (8) in the form of minimizing a function subject to a set of constraints by adding slack variables

$$\min_{w,\xi} \left\{ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \right\} \tag{9}$$
$$s.t.\ \forall i \in \{1, 2, \ldots, n\}, \forall y \in \mathcal{Y}_H :$$
$$F(x^{(i)}, y_L^{(i)}, y_H^{(i)}, z; w) - F(x^{(i)}, y_L^{(i)}, y, z; w) \geq \Delta(y_H^{(i)}, y) - \xi_i$$

Note that there are exponential number of constraints in (9). This can be solved by the cutting-plane method [24].

Another intuitive way to understand the CCCP algorithm is to consider it as solving the learning problem with incomplete data using Expectation-Maximization (EM). In our training data, the latent variables are not given. We can start by initializing the latent variables. Once we have the latent variables, the data become complete. Then we can use the standard Structured-SVM to learn the model parameters (M-step). After that, we can update the latent states again using the parameters that are learned (E-step). The iteration continues until convergence.

IV. EXPERIMENT AND RESULTS

A. Dataset

The models are evaluated on the benchmark dataset CAD-120 [4]. The dataset consists of 120 RGB-D videos, which are collected by the Microsoft Kinect sensor. Each video contains one high-level activity and a sequence of sub-level activities. There are in total 10 high-level activities in the dataset, including *making cereal*, *taking medicine*, *stacking objects*, *unstacking objects*, *microwaving food*, *picking up objects*, *cleaning objects*, *taking food*, *arranging objects*, *having a meal*. The dataset also consists of 10 sub-level activities, *i.e. reaching*, *moving*, *pouring*, *eating*, *drinking*, *opening*, *placing*, *closing*, *scrubbing*, and *null*. Both high-level and sub-level activities are manually annotated in the dataset. Using the skeleton tracker from OpenNI[2], body part locations are obtained for each frame. The objects are detected in an automatic way and locations of the objects are also provided by the dataset.

The dataset is very challenging in the following aspects. a) The activities in the dataset are performed with four different actors. They behave quite differently, *e.g.* left or right handed, front view or side view. b) There is also a large variation even for the same activity. *e.g.* the sub-level activity *opening* can refer to opening a bottle or opening the microwave. Although both of them have the same label, they appear significantly different from each other in the video. c) Partial or full occlusion is also a very challenging aspect for this dataset. *e.g.* in some of the videos, legs are completely occluded by the table, and objects are frequently occluded by the other objects. This makes it difficult to obtain accurate object locations as well as body skeleton, therefore the generated data is noisy.

We choose to evaluate on this benchmark dataset because there are existing approaches [4], [7] that we can directly compared with.

[2] http://www.openni.org/

B. Evaluation Criteria

All models are evaluated in terms of accuracy, precision and recall with 4-fold cross-validation. The folds are separated based on different human actors, *i.e.* the model is trained using videos performed by 3 persons and it is then tested on a new person.

C. Experiment Setup

We compare two different segmentation methods to the videos. In the first method, we use the ground truth segmentation which is manually annotated. For the second segmentation, we apply an appearance-based approach, *i.e.* we extract the spatial-temporal features for all the frames, and similar frames are grouped together to form segments using a graph-based approach. For the reason of comparison, we use the same video segmentation parameters as in [4].

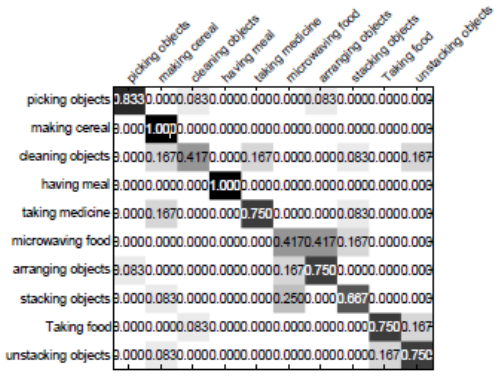
In our model, parameters of the two layers are trained in parallel sessions. In the first session, we learn a latent discriminative model for recognizing sub-level activities based on the RGB-D videos. Following [4], we extract three set of features from the video, which encode human-object interaction, object-object interaction and temporal interaction respectively. In the second session, we learn a model that infers the high-level activity from the sequence of sub-level activities. We extract unigram and bigram features based on the sub-level activity sequence as well as the occlusion features as described in Section III-B.1. Our model consists of latent variables. The cardinality of the latent variables is a hyper parameter that is estimated based on cross-validation. During testing, the low-level activities and high-level activities are recognized in succession. We first infer the sub-level activities in the first layer. After that, we use the learned parameters to map the obtained sub-level activities to high-level activities.

Our model is compared with two baseline approaches. In the first baseline approach, we use a single layer model for recognizing high-level activities, *i.e.* we learn a direct mapping from video features to the high-level activity. As the second baseline, we adopt the recent work [4] in activity recognition for comparison.

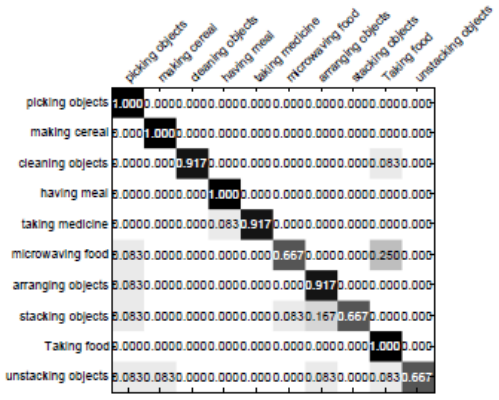
D. Results

Table I compares the performance of different methods on the recognition task of high-level activities. "Single Layer" refers to the first baseline approach where we learn a direct mapping from video level features to high-level activities. The single layer approach reaches an average performance of over 70% in both segmentation methods but with a large standard derivation of around 10%. In contrast, the two-layered approaches outperform the single-layer approach by over 10 percentage points. Notably, when using the model with unigram features, we reach the best performance on both segmentation methods. For the ground truth segmentation, our model with unigram segmentation gets 90% in accuracy and 92.8% in precision and 89.7% in recall, and the standard deviation is less than 3 percentage points. When using appearance-based segmentation, both accuracy and recall drop to below 80% while the average precision is 86.4%. After adding bigram features, the performance drops slightly. We believe that this is because of the sparsity of sub-level activities and that the performance will be better than the unigram once we have more training data for high-level activities.

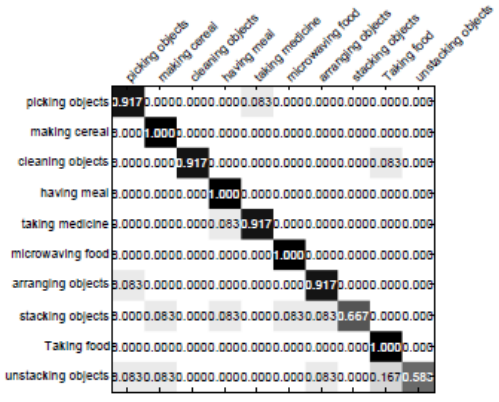
Fig. 3 illustrates the confusion matrix of both single-layered approach and the proposed methods. We can see that the activities including *cleaning objects*, *microwaving food* and *stacking objects* are heavily confused with other activities in single-layered approach. In contrast, there is a strong diagonal for the hierarchical approach with small errors, *e.g.* using the unigram+bigram, the microwaving food is confused with taking food and unstacking objects is confused with picking objects, making cereal, arranging objects and taking food. Our model performs better using only the unigram features and only a few errors occur for unstacking objects and stacking objects.



(a) Single-layer Approach



(b) Hierarchical Approach (unigram+bigram)



(c) Hierarchical Approach (unigram)

Fig. 3. Confusion matrix of high-level activity recognition

V. Conclusion and Future Work

In this paper, we present a two-layered approach that can recognize low-level and high-level human activities simultaneously. We investigate the effect of using latent variables, segmentation methods, as well as different feature representations. Our results show that the two-layered approach performs better than the approach with only a single layer. Our model is also shown to outperform the state-of-the-art on the same dataset.

Currently, our approach only uses the RGB-D videos for activity recognition. In our future work, we would like to fuse different cues, *e.g.* human locations [25], human identities [14] and ambient sensors [26], for robust estimation of human activities.

Acknowledgment

We thank Hema Koppula for very helpful suggestions and also thank her for providing us with features and segmentation details of the CAD-120 dataset.

References

[1] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 842–849.

[2] K. Tang, L. Fei-Fei, and D. Koller, "Learning Latent Temporal Structure for Complex Event Detection," in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 1250–1257.

[3] T. Lan, L. Sigal, and G. Mori, "Social roles in hierarchical models for human activity recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*. Ieee, June 2012, pp. 1354–1361.

[4] H. S. Koppula, R. Gupta, and A. Saxena, "Learning Human Activities and Object Affordances from RGB-D Videos," *International Journal of Robotics Research (IJRR)*, vol. 32, no. 8, pp. 951–970, 2013.

[5] H. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," in *Proceedings of the Robotics Science and Systems (RSS)*. RSS, 2013.

[6] N. Hu, G. Englebienne, and B. Kröse, "Posture Recognition with a Top-view Camera," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 2152–2157.

[7] N. Hu, G. Englebienne, Z. Lou, and B. Kröse, "Learning Latent Structure for Activity Recognition," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[8] J. K. Aggarwal and M. S. Ryoo, "Human Activity Analysis: A Review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 16, 2011.

[9] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[10] J. Liu and M. Shah, "Learning human actions via information maximization," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[11] S. Niyogi and E. Adelson, "Analyzing and recognizing walking figures in XYT," in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 1994.

[12] M. Ryoo and J. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 2009.

[13] M. Hoai, Z. Lan, and F. D. la Torre, "Joint segmentation and classification of human actions in video," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3265–3272.

[14] N. Hu, R. Bormann, T. Zwölfer, and B. Kröse, "Multi-User Identification and Efficient User Approaching by Fusing Robot and Ambient Sensors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[15] P. Matikainen, R. Sukthankar, and M. Hebert, "Model recommendation for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2256–2263.

[16] Q. Shi, L. Cheng, L. Wang, and A. Smola, "Human Action Segmentation and Recognition Using Discriminative Semi-Markov Models," *International Journal of Computer Vision (IJCV)*), vol. 93, pp. 22–32, 2011.

[17] R. Kelley, M. Nicolescu, A. Tavakkoli, C. King, and G. Bebis, "Understanding human intentions via hidden markov models in autonomous mobile robots," in *Proceedings of the International Conference on Human-Robot Interaction (HRI)*. IEEE, 2008, pp. 367–374.

[18] Y. Ivanov and A. Bobick, "Recognition of visual activities and interactions by stochastic parsing," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, 2000.

[19] S. Savarese, A. DelPozo, J. Niebles, and L. F.-F. L. Fei-Fei, "Spatial-Temporal correlatons for unsupervised action classification," in *2008 IEEE Workshop on Motion and video Computing*, 2008.

[20] J. Fürnkranz, "A study using n-gram features for text categorization," *Austrian Research Institute for Artifical Intelligence*, pp. 1–10, 1998.

[21] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *Journal of Machine Learning Research*, pp. 1453–1484, 2005.

[22] C. N. Yu and T. Joachims, "Learning structural SVMs with latent variables," in *Proceedings of the International Conference on Pattern Recognition (ICPR)*. ACM, 2009, pp. 1169–1176.

[23] A. Yuille and A. Rangarajan, "The concave-convex procedure (CCCP)," *Advances in neural information processing systems (NIPS)*, vol. 2, pp. 1033–1040, 2002.

[24] J. E. Kelley and Jr., "The cutting-plane method for solving convex programs," *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, no. 4, pp. 703–712, 1960.

[25] N. Hu, G. Englebienne, and B. Kröse, "Bayesian Fusion of Ceiling Mounted Camera and Laser Range Finder on a Mobile Robot for People Detection and Localization," in *IROS workshop on Human Behavior Understanding*, ser. Lecture Notes in Computer Science. Springer, 2012, vol. 7559, pp. 41–51.

[26] D. McIlwraith, J. Pansiot, and G.-Z. Y. G.-Z. Yang, "Wearable and ambient sensor fusion for the characterisation of human motion," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.