Low Cost Activity Recognition Using Depth Cameras and Context Dependent Spatial Regions

Michael Karg[1]* and Alexandra Kirsch[2]

[1] Institute for Advanced Study, Technische Universitt München, Lichtenbergstrasse 2a, D-85748 Garching, Germany, http://hcai.in.tum.de
[2] Department of Computer Science, University of Tübingen, Sand 14, D-72076 Tübingen, Germany, http://www.hci.uni-tuebingen.de

Abstract. To be useful helpers for humans in domestic environments, robots should be aware of human task execution to anticipate and adequately react to human actions. Hence the field of activity recognition has become of increasing interest in the robotics community and many approaches are based on sequences of object detections or human posture recognition, requiring the environment to be equipped with loads of sensors or extremely expensive motion tracking systems. In this paper we investigate the use of inexpensive depth cameras to perform activity recognition using context dependent spatial regions with two different approaches for activity recognition: Spatio-Temporal Plan Descriptions and Hierarchical Hidden Markov Models. We evaluate both approaches in a simulated and a real-world environment, showing that reliable activity recognition is possible using a sensor setting for less than 250 $ in a spatially limited environment.

1 Introduction

Personal robots that work together with humans in human-centered environments are seen as a promising future application area for robotic systems. They are expected to leave closed factory environments and assist us with uncomfortable tasks in our homes. Especially elderly people with minor disabilities are seen as beneficiaries of domestic robot helpers since they could help them to live independently in their own home as long as possible [19]. Mitzner et al. [12] asked 21 independent living seniors for which activities they would prefer assistance rather from a robot than a human and found that this was the case for 28 out of 48 activities, which mostly involved household duties (e.g. cleaning and washing) or manual labor (e.g. gardening, mowing the lawn). A robot that is a useful helper for humans should have certain knowledge about the current human activity to be able to take into account human behavior and react adequately to it. We think that this capability plays a key role for future robotic

* With the support of the Technische Universität München - Institute for Advanced Study, funded by the German Excellence Initiative.

helpers that are to perform complex tasks in human centered environments and is essential to enable commercial success of such robots. Imagine a household robot that is supposed to clean the kitchen after its user had diner. In this case, the robot should be aware whether the human has already had diner and not clean the table in a case where the human has not eaten yet. Also for the detection of errors, activity recognition plays an important role. If we imagine a human sleeping in the morning or late evening, this should be classified as a normal event, but a sleeping human in the middle of the day or on the ground could be a sign for the robot that something is not right. Thus, the ability for a robot to perform activity recognition and distinguish different human behaviors in different contexts from each other is essential if household assistants should become useful and comfortable assistants in human environments.

While there already exist a couple of approaches to equip robots with this capability, mostly the environment has to be equipped with loads of sensors like RFID tags and readers or extremely expensive motion tracking systems. In this paper we investigate two different approaches of activity recognition that use a Kinect for motion tracking to allow reliable activity recognition in spatially limited environments. Our application domain is a kitchen environment. We evaluate our approach in a simulated scenario and in a real-world kitchen environment using only a Kinect sensor.

2 Related Work

Many approaches in the field of activity recognition make use of Hidden Markov Models (HMMs) to distinguish between various high level tasks. Buettner et al. [2] equip a large set of everyday objects in an apartment with RFID-based sensors and record sequences of object detections while a person performs typical everyday activities to train an HMM for activity recognition. Nguyen et al. [13], in contrast, use manually assigned spatial regions and a multi-camera tracking system to train a HMM for recognizing high level activities. They use an Abstract Hidden Markov Model and extend it with a memory that allows them to model a richer class of context-free and state-dependent behaviors. Also other researchers use different extensions to the Hidden Markov Model to overcome various limitations. Nguyen et al. [14] and Bui et al. [3] introduce the general concept of hierarchies to HMMs which become Hierarchical Hidden Markov Models (HHMMs), while Duong et al. [4] propose to use different layers in the HMM to account for hierarchies and durations. But HMMs are not the only models used for activity recognition. There are also approaches using Hierarchical Conditional Random Fields (CRFs) [16], Hierarchical Maximum Entropy Markov Models (MEMM) [18] or Monte-Carlo based methods [15].

Except for Sung et al. [18], who use RGBD cameras to detect activities from human body posture, most approaches rely on the use of object detections or motion tracking systems that produce quite reliable data but are very expensive and intrusive. Perkowitz et al. [15] equip everyday objects in a human household with RFID tags to obtain sequences of object detections from human everyday

activities and Buettner et al. [2] introduce RFID-based sensors, which are less intrusive, but still the apartment they used for testing had to be equipped with four antennas, and sensors had to be attached to the 25 objects they used.

Townsend et al. [22] found that humans partition activities of daily living (ADLs) into sequences of subtasks that they carry out at the same places and usually even at similar times. Trying to analyze such ADLs, Logan et al. [11] set up a sensor-equipped apartment, in which a married couple was living for 10 weeks given the simple task to just normally continue with their life. The data has been made publicly available as the MIT PlaceLab Dataset PLCouple[1]. One of their key findings regarding activity recognition, was that for most activities, motion-based sensors yield better performance than other modalities like reed switches or RFID sensors. When humans think about spatial regions, they tend to classify them according to their functional use instead of geometry alone. Zender et al. [25] propose that a robot working in a human-centered environment would have advantages if it understood its environment in terms of human spatial concepts. Also Klenk et al. [9] see the understanding of human spatial concepts as "essential for cognitive systems performing tasks for humans in everyday environments". They use context dependent spatial regions by learning from qualitative spatial representations and semantic labels. To automatically learn the context of locations, Stulp et al. [17] represent spatial regions according to their use to equip a robot with an understanding of action-related places (ARPlaces) using probability distributions that model the chance for a robot to successfully grasp an object. Liao et al. [10] analyze patterns of GPS traces to automatically label significant places that the human visits during his daily life (like his home or working place) according to their function. In previous work [8] we learned context dependent spatial regions by analyzing motion tracking data and defining regions, that a human visits to perform pick- and place actions, relative to storage places of objects. These spatial regions can then be used to model human activities in spatio-temporal plan descriptions (STPRs) which they directly use to distinguish between different human behaviors.

In this paper, we will first use STPRs for activity recognition using string comparison methods in a spatially challenging environment and then make use of Hierarchical Hidden Markov Models (HHMMs). HHMMs are also used for activity recognition based on simulated behavior models by Bui et al. [3] and Nguyen et al. [14] using manually labeled locations in an office environment. In contrast to their work, we will use context dependent spatial regions that were learned from motion tracking data in a real-world setting and perform activity recognition in a narrow real-world kitchen environment that makes it hard to distinguish unique locations. We will compare STPRs with HHMMs for activity recognition and set up a system that performs activity recognition in real-time using motion tracking data and object detections in simulation and a real-world scenario.

[1] `http://architecture.mit.edu/house_n/data/PlaceLab/PLCouple1.htm`

3 Activity Recognition Using Context Dependent Spatial Regions

For the generation of our model, we assume that we have a semantic map of the environment that provides us with information about objects and furniture in the environment. Such semantic maps lately have become increasingly popular and can be generated widely autonomously as Blodow et al. [1] show. Tenorth et al. [20] even link such semantic maps with knowledge bases to perform context dependent reasoning about the environment of a robot like inferring likely storage locations of objects in a kitchen.

From a dataset of 12 participants performing different pick- and place tasks — which is different from the dataset we used for evaluation — we learned where humans generally are located when performing specific actions, such as grasping objects from a table, relative to reference objects, such as the table. Combining this information with the semantic map of the environment, we generate a spatial model ψ that represents locations a human visits during the execution of different activities in a specific environment. It consists of a set of Gaussians P_i that represent annotated locations in reference to furniture objects o_i:

$$\psi = \{l_1, l_2, ..., l_n\} \text{ with } l_i = (P_i, o_i)$$

So the spatial model extends a semantic map with general information about locations where the human is likely to perform actions related to objects in the semantic map. In our case the locations represent locations where humans generally picks up objects from different furniture objects. An advantage of such a representation is that spatial regions can be equipped with a meaning by linking them to instances in the semantic map. Thus, a robot can be aware that the location where the human currently standing is not only a set of coordinates, but a place from which he usually picks up things from a drawer giving us the possibility to map coordinates in the map to qualitative representations of locations like "Drawer". For this work, we will use this information only for the naming of the locations, but in general one could do more sophisticated reasoning like inferring the locations from where a human typically picks up cold drinks or similar.

We use this spatial model to map coordinates in the map to qualitative representations of locations and based on such a spatial model, we generate two different representations as a basis for activity recognition:

- Spatio-Temporal Plan Representations as introduced in out previous work [8]
- Hierarchical Hidden Markov Models as used by Bui et al. [3].

3.1 Spatio-Temporal Plan Representations

Spatio-Temporal Plan Representations (STPRs) as presented in [8] describe an action by the location the action is performed at and the time it takes to complete the action. It ignores typical action properties that are typically used, for

example in AI planning, such as preconditions or the goal of the action. We can describe an activity as a set of n tuples, each consisting of a location l_i and a duration t_i.

$$\langle (l_1, t_1), (l_2, t_2) ..., (l_n, t_n) \rangle$$

Such spatio-temporal plan representations can be used to distinguish between different activities using string comparison methods like the *Generalized Levenshtein Similarity (GLS)* [24] on a string representation of the sequences of locations as shown in [8]. GLS is a normalized edit metric for two strings, which is defined as follows:

$$GLS = \frac{\alpha * (|X| + |Y|) - GLD\,(X, Y)}{2}.$$

Here X and Y represent string-representations of the location sequences that model our activities and α describes the maximum weight of all of the three string-edit operations (insert, delete, replace). GLD is the Generalized Levenshtein Distance, which basically measures how many string operations are necessary to transform string X into string Y. The GLS computed from string-representations of the locations are used as confidence values for a match between an observed sequence and a known sequence. In this work, we will only use the locations of the STPRs for activity recognition.

3.2 Hierarchical Hidden Markov Models

Another way to represent human activities using context dependent spatial regions is to use an HHMM from the sequences of locations from our spatial model. This representation has the advantage over STPRs that as a probabilistic model it can account for uncertainties in the observations and variations in the order of subtasks in an activity. A Hierarchical HMM introduces the concept of hierarchies to HMMs by allowing each state of the HMM to be an HMM itself as indicated in Figure 1. This allows us to model multi-level stochastic processes and makes it necessary to differentiate between two types of states: The *internal states* are hidden states that are HMMs themselves and don't emit single observation symbols, but rather sequences of observations by recursive activation of one of its substates. States that actually emit output symbols and are located at the lowest hierarchical level are called the *production states*. The activation of a substate by its internal state is called a *vertical transition* while a transition between two production states is referred to as *horizontal transition*. Every sequence of production states has exactly one terminal state, which, when reached, ends the process of recursive state activation and leads to a vertical transition upwards in the hierarchy.

In a more formal description, an HHMM can be described as a three-tuple consisting of a topological structure ζ, an observation model Y and a set of parameters θ. The topology defines the number of levels $D = \{d_1, ..., d_n\}$, parent-child relationships between levels and the state space at each level, while the

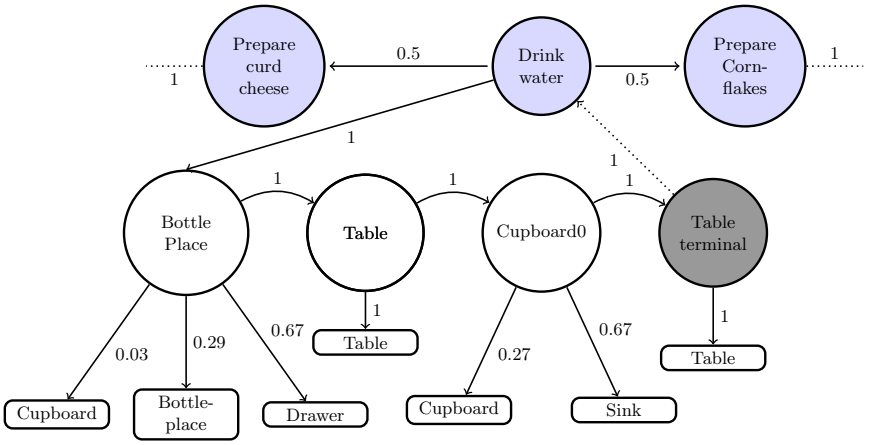

Fig. 1. A hierarchical HMM generated from a sequence of observed context dependent spatial regions. Light gray nodes represent plan-states of the HMM, which are HMMs themselves. Only the "Drink Water" HMM is shown in detail. White nodes correspond to locations where the observed human is standing and represent the production states of the HHMM. The dark gray node is a terminal state leading to a vertical transition in the HHMM when it is reached. Rectangles are observations that are expected at the specific locations.

observation model describes the set of possible observations $Y = \{y_1, ..., y_m\}$. Given ζ and Y, the set of parameters of the HHMM θ is defined by

$$\theta = \{B_{y|p}, \pi^{d,p^*}, A_{i,j}^{d,p^*}, A_{i,end}^{d,p^*} \mid \forall\,(y, p, d, p^*, i, j)\}$$

where $B_{y|p}$ describes the probability of an observation $y \in Y$ while being in production state p and π^{d,p^*} represents the initial distribution over all children of the internal state p^*. The transition probabilities between child nodes $i, j \in child\,(p^*)$ are described by $A_{i,j}^{d,p^*}$ and the probability of an internal state terminating given its child is the production state i is $A_{i,end}^{d,p^*}$. In our application of HHMMs for activity recognition, we are interested in the posterior marginals $P(p_i^* \mid y_{1:t})$ of the internal states at every point in time, which, in our case, represent the probabilities of the human performing specific activities given a sequence of location observations. One should note that any HHMM can be decomposed into a standard HMM using the "flattening" method [23] which makes algorithms that work on HMM applicable to HHMMs. We use a variant of the Forward Backward Algorithm on the flattened HHMM, thus obtaining the posterior marginals for all states. For a more detailed formal description of HHMMs, which lies beyond the scope of this paper, we refer to the work of Fine et al .[6] and Bui et al. [3].

4 A Morning Routine Dataset

To obtain a realistic experimental setting for a domestic robot helper, we decided to use a typical morning routine of a human in a kitchen since we think that this is one of the main future operation areas of domestic robot helpers. We investigated morning routines of the MIT PlaceLab PLCouple dataset [7] and found that the number of different activities that a human commonly performs during his morning routine is rather limited. During 10 weeks, the participants of the study performed only 23 different activities between 6 and 12 am of which only 11 were performed in the kitchen. Although annotations of the PLCouple dataset are publicly available, sensor data has only been recorded for one of the two participants since, due to financial restrictions, there was only one RFID reader available for the experiment. As many tasks in the morning routine were performed cooperatively and full audio and video data is not available because of privacy issues, we decided to record a new dataset that realistically captures the human morning routine of a person.

We asked one male participant, who had no knowledge about our system, to note all activities that he performed on a weekday in the time between getting up and going to work for three weeks. He furthermore was instructed to write down the approximate durations of the activities as well as the locations at which he stood still while performing those tasks. In compliance with the data from the MIT PlaceLab PLCouple dataset, in our data the number of activities that the participant performed in the kitchen between 6 and 12 am was limited to 10: prepare a drink, drink a glass of water, prepare cereals, eat cereals, prepare curd-cheese, eat curd-cheese, prepare bread, eat bread, clean the table and prepare for work.

To obtain motion tracking data and object detections, we equipped an experimental kitchen with two Kinect sensors, one for motion tracking using the Openni tracker[3] and one for object detections based on visual markers using the ROS AR Kinect toolbox[4]. The visual markers were used to limit the effort of our experiments and could be replaced with more elaborate object detection systems e.g. based on cameras. Motion tracking worked quite well, but especially when dealing with partial occlusions, the tracker sometimes lost track of the person or produced inaccurate measurements resulting in a "jumping" of several joints of the tracked person. The object detections using the visual markers unfortunately performed poorly in our setting and were not used for this work.

In previous work, we made good experiences with the testing of algorithms in simulation, so we also set up a simulated environment of the same kitchen using the MORSE simulator[5] [5] which includes a human avatar that can be controlled by a human and used to perform pick- and place tasks in the simulated environment like in 3D computer games. The experimental kitchen is shown in Figure 2 in its simulated version as well as the real kitchen environment.

[3] `http://ros.org/wiki/openni_tracker`
[4] `http://www.ros.org/wiki/ar_kinect`
[5] `http://morse.openrobots.org`

Furthermore, the picture shows a visualization of the motion tracking data and object detections that are obtained by the two Kinect sensors.

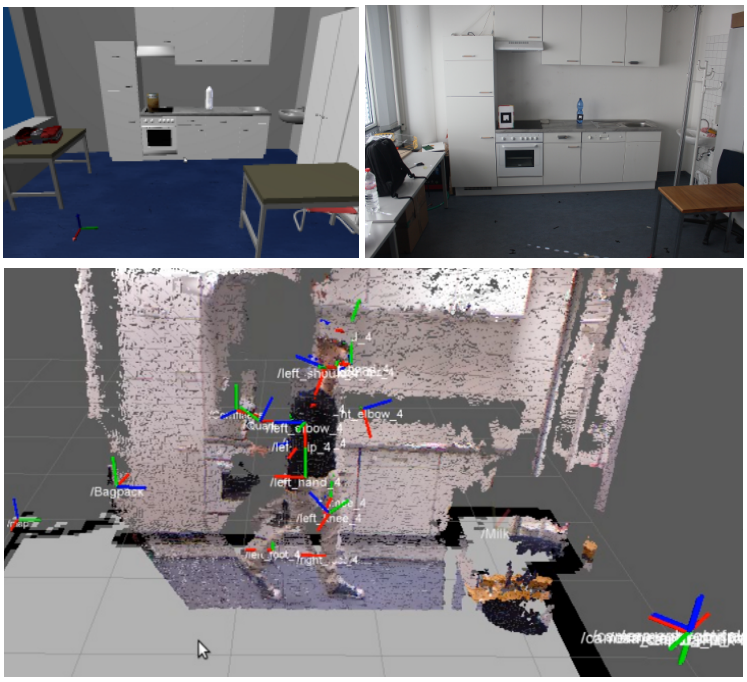

Fig. 2. Upper left: The simulated environment in the MORSE simulator. Upper right: The real experimental kitchen environment. Lower picture: Sensor input of the real scenario. The motion tracker returns coordinate transformations for each joint of the human while the visual marker detection provides us with the approximate position of the markers.

We asked the participant to reenact his common morning routine for each day of the three weeks according to his notes in the experimental kitchen in the real world as well as in simulation. Since in the simulated scenario there is no possibility to simulate a human eating or drinking, the participant waited for a realistic amount of time at e.g. the table when at this point of time he would be eating or drinking. Furthermore, the data was manually annotated with ground truth labels and made available to the public for download[6].

5 Approach and Evaluation

We create a spatial model of our experimental kitchen environment using a semantically annotated map of the environment and the learned relative locations

[6] (URL omitted due to double blind review)

as explained in section 3. To keep the effort limited, we manually generate STPRs and the state transitions of the HHMMs of the activities using sequences of context dependent spatial regions based on the spatial model. However, there are also approaches to learn such models from observations [8,**?**] or even infer actions and likely places from instructions from the web [21]. One disadvantage of the straightly linear nature of STPRs is the missing possibility of accounting for uncertainties in the order in which sub-actions of the activity were performed. For example while setting the table, the participant sometimes brought the plate first, while at another time, the first object that he picked up, were the cereals. We decided to generate one STPR for each such variation, but treating all of the variations as the same activity when calculating probabilities.

To extract locations in the data, we had to take into account that humans never stand completely still. A location is detected when the person is not moving more than 25 cm within 0.5 seconds (0.5 m/s) and a simple motion pattern of moving towards an object, staying there, and moving away again, is detected. When we detect such an event, the spatial model is queried using the current coordinates of the human, which furniture object the human is most likely pick up something from and a location observation is added to our observation sequence.

5.1 Single Activity Recognition

We calculate normalized confidence values using Generalized Levenshtein Similarity (GLS) as soon as an activity is finished. Activity recognition using the GLS approach has quite a hard time distinguishing between different plans. Table 1 shows the probabilities assigned to each known activity when the "Prepare Cereals" activity is observed. The values are averaged over the 8 days in the dataset when the participant had cereals for breakfast (for better comparability with the data of the HHMMs, probabilities were calculated form the GLS values). $P(a)_{sim}$ is the probability of activity a being observed in the simulated dataset and $P(a)_{real}$ corresponds to the same probability for the real dataset.

Even though the maximum of the average values correctly classifies "Prepare Cereals" as the most likely activity, variance of the mean values of all plans is small and in some cases, activity recognition is undecided or wrong. In the simulated data, still 7 of the 8 "Prepare Cereals" instances are classified correctly, but with the difference to the probabilities of the other activities being rather small. In one out of the 8 cases, "Prepare Cereals" has been classified wrongly as one of "Prepare curd cheese" or "Clean table after cereals" (with the same probability). For the real data, only 2 of the activities were correctly classified, in 4 cases, classification was wrong and in the remaining 2 cases, probabilities for two plans were the same (including the correct one).

In contrast to [8], activity recognition performs significantly worse, which we found to be caused by observations of spatial regions that were labeled wrongly when querying the spatial model. In [8], there were only few spatial regions which also were quite unique. Our setting contained far more spatial regions that were located close to each other, resulting in many overlappings of the corresponding gaussians. This is caused due to the fact that in a some kitchens (like ours from

a) STPRs and GLS:

Activity s	$P(a)_{sim}$ (%)	$P(a)_{real}$(%)
Drink water	11.24	11.87
Prepare cereals	23.40	19.94
Prepare curd cheese	21.40	19.49
Clean table cereals	18.34	18.11
Clean table curd cheese	16.61	17.66
Prepare work	9.00	12.93

b) HHMMs:

Activity s	$P(a)_{sim}$ (%)	$P(a)_{real}$(%)
Drink water	2.30	1.78
Prepare cereals	55.56	41.98
Prepare curd cheese	26.68	34.08
Clean table cereals	9.84	5.57
Clean table curd cheese	3.44	14.20
Prepare work	2.18	2.42

Table 1. Probabilities for activities when "Prepare cereals" is performed using STPRs and GLS (a) and HHMMs (b).

figure 2), cupboards are for example located above drawers or other cupboards, which leads to the human standing at the almost same location when picking up an object from a drawer or a cupboard that is directly above the drawer. A comparison of both spatial models can be seen in figure 3.

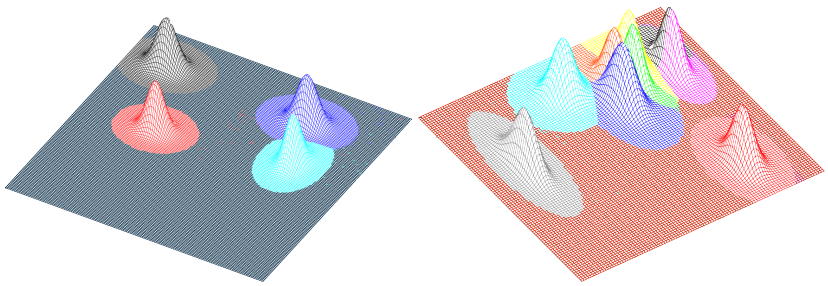

Fig. 3. Left picture: A spatial model with only a four context dependent spatial regions. In this case, it is easy to distinguish between different locations. Right picture: A spatial model of a more realistic kitchen environment where some furniture objects are close to each other. Most of the gaussians are located very close together and thus make it hard to reliably detect unique locations due to overlapping.

The overlapping of the spatial regions partially resulted in wrong observations that were added to our observation sequence in cases when two spatial areas and their corresponding gaussians were close to each other. Since STPRs do not use any observation model, in cases of heavy overlapping one could see the labeling of those spatial regions as almost random.

HHMMs should overcome these limitations. The transitions of the HHMM were manually set according to the notes of our dataset. To calculate the observation probabilities, we used the ground truth labeling of the dataset and calculated observation probabilities in an initial training phase also using Maximum Likelihood estimation. This part is the most time consuming part in the initialization, but we also conducted first experiments on estimating observation probabilities directly from the spatial model of the environment. Further investigations towards this direction are part of future work.

The HHMM we generated this way comes with the advantages of accounting for uncertainties in the order in which the locations were visited, as well as uncertainties in the observations of the locations. Furthermore, it can directly be used to estimate probabilities for all of its internal states, the activities, at every time step using the Forward-Backward Algorithm, giving the posterior marginals over all activities. Table 1 shows probabilities for the activities when the human performs the "Prepare Cereals" averaged over the 8 days when the human had cereals, this time using the HHMM for recognition. Compared to the probabilities when using GLS as in table 1, we see that HHMM-based recognition clearly outperforms the GLS approach. The probabilities indicate that the observations fit well to one of the "Prepare Food" tasks. However, the distinction between "Prepare Cereals" and "Prepare curd cheese" is rather difficult, since the difference between both plans is rather small. When preparing cereals, the human goes to the ceramic glass cooktop (location of the cereals) first, whereas when preparing curd-cheese, he first visits the fridge. The location of the ceramic glass cooktop and the refrigerator are very close to each other (as can be seen in picture 2), so especially when using the noisy Kinect tracking, we can hardly distinguish between those places. Overall, the HHMM approach classified all 8 of the prepare cereals activities and 2 of the 4 prepare curd-cheese activities correctly in the simulated data. For the real data, 7 of the 8 "prepare cereals" activities were classified correctly and 1 of the 4 "prepare curd-cheese" activities. In all false positive cases, the most likely activity was the other food-preparing activity.

5.2 Live Activity Recognition Using HHMMs

The experiments illustrated above show that the HHMM based approach is able to detect single plans although some spatial regions are close to each other and thus overlapping. The more interesting, although more difficult use case of our activity recognition is when the human performs several plans after another. We set up our HHMM based approach to perform live activity recognition in a kitchen environment and in the following experiments, we will measure how well this approach is able to detect different activities over time, including transitions between activities. We calculated precision and recall values for each activity a in the following way:

$$precision = \frac{|t_a \cap t_a^*|}{|t_a^*|}.$$

Where t_a represents the time when activity a has been executed by the participant according to the ground truth labels of the dataset and t_a^* stands for the time where the detection estimates activity a to be the most likely one. Accordingly recall was calculated in the following way:

$$recall = \frac{|t_a \ \cap \ t_a^*|}{|t_a|}.$$

Furthermore, we calculate the *accuracy* which is the proportion of true classification results (true positives and true negatives) during the whole observation period t_{obs}:

$$accuracy = \frac{|t_a \ \cap \ t_a^*| + |\bar{t_a} \ \cap \ \bar{t_a^*}|}{|t_{obs}|}.$$

$\bar{t_a}$ corresponds to the time when activity a has not been performed and $\bar{t_a^*}$ represents time periods when activity a has not been classified as the most likely activity. We did the experiment using data from simulation as well as the real data of our dataset and the resulting precision, recall and accuracy values for each activity can be found in Table 2 a) for the simulated data and Table 2 b) for the real data. Although the precision and recall rates might suggest otherwise, the "Prepare Work" activity is recognized very well in most cases as can also be seen in figure 4. The reason for having lower precision and recall rates in our experimental setting, is that we unluckily chose the position we used for the initialization of the skeleton tracker (and the starting position of the human in simulation) close to a region that is (uniquely) assigned to the "Prepare Work" activity, thus creating a strong bias towards this activity in the beginning resulting in lower precision rates of the "Prepare work" activity and lower recall rates of the "Drink water" activity.

The values indicate that our system can distinguish between different plans, although the recognition between some activities does not perform really well. Again, if it comes to the recognition of the different food preparing and cleaning activities, distinction between them is rather hard due to their similarity which seems to be a bigger problem in the noisy real-data than in simulation. To get an impression whether the approach is able to reliably recognize the different categories of activities, we performed another experiment using the real data where we merged the "prepare cereals" and "prepare curd cheese" plans into one "prepare food" activity as well as both clean table plans into one. We repeated the experiment and calculated precision, recall and accuracy, which resulted in the values shown in Table 3.

A plot of plan probabilities over time for the simulated data, real data and real data with merged "prepare food" and "clean table" plans are shown in Figure 4. We can mostly draw conclusions about different human activities using only context dependent spatial regions. Only the recognition of the "Drink water" activity is not recognized at all in some cases. We think is due to being the first activity, only few observations being given to the HHMM. Since we initialize the probabilities of all of our states of the HHMM uniformly, it has a hard time finding the correct activity at the beginning when only one or two observations

a) Simulated data

Activity	Prec. (%)	Recall (%)	Acc. (%)
Drink water	66.3	62.5	86.8
Prepare cereals	95.1	96.6	94.4
Prepare curd cheese	63.8	46.5	62.8
Clean table cereals	87.9	64.7	94.0
Clean table curd cheese	45.2	44.7	89.5
Prepare work	44.6	68.0	92.6

b) Real world data

Activity	Prec. (%)	Recall (%)	Acc. (%)
Drink water	35.9	37.0	76.4
Prepare cereals	51.9	67.5	62.9
Prepare curd cheese	34.8	25.0	63.0
Clean table cereals	68.4	23.2	82.3
Clean table curd cheese	85.8	34.1	84.9
Prepare work	63.4	91.3	92.6

Table 2. Average precision, recall and accuracy for 12 experiments of the simulated and real data using only locations with HHMMs.

Activity	Precision (%)	Recall (%)	Accuracy (%)
Drink water	64.0	61.6	87.0
Prepare food	69.4	68.7	73.8
Clean table	49.2	79.5	79.9
Prepare work	90.6	48.5	94.4

Table 3. Average precision, recall and accuracy for 12 experiments of the real data using only locations with HHMMs and combined plans.

are available. To improve recognition rates at the beginning, one could think of biasing probabilities for some activities at the initialization of our system since e.g a human will most likely have breakfast and *afterwards* clean the table.

Another way to increase recognition rates is the inclusion of object detections. Since one of our goals is to avoid equipping the environment extensively with sensors, RFID tags, etc., we investigated the use of only few object detections as a proof-of-concept showing that a combination of activity recognition solely based on locations in combination with few object detections can produce a very reliable activity recognition. We therefore bias the probabilities of the activity recognition by just observing which objects appear in an activity at all and which don't, i.e. if an object is detected to be used by the human, probabilities for activities the include the object are rewarded, while activities that do not include the object are penalized. We use this rather simple approach as a proof-of-concept, but there are also more elaborate systems based on sequences of object detections, for example by Buettner et al. [2]. We classified a detected object as "used by the human" when it was in reach of the human and changed its position since its last detection. Precision, recall and accuracy for the simulated dataset with partial object detections are shown in Table 4. Out of the 25 object

interactions of the user, on average 15 were detected. These detections already allow us to bias the activity recognition and increase the system performance (on average: Accuracy of 94.1 %).

Activity	Prec. (%)	Recall (%)	Acc. (%)
Drink water	90.2	73.5	94.2
Prepare cereals	96.9	98.1	96.4
Prepare curd cheese	97.9	88.4	91.0
Clean table cereals	84.1	73.4	96.2
Clean table curd cheese	61.7	48.2	91.3
Prepare work	68.8	86.5	95.7

Table 4. Average precision, recall and accuracy for 12 experiments of the simulated data with partial object detections using HHMMs.

6 Discussion

When modeling linear activities using unique locations, STPRs offer a cheap and easy way to model activities and perform activity recognition while providing a decent model to merge information about patterns of actions and their corresponding durations. Another advantage of STPRs is the possibility to explicitly model temporal sequences which is only partially possible using HMMs due to the Markov Assumtion which they are based on. This property makes STPRs also applicable in other applications like montoring, e.g. in applications when it also has to be accounted for how often specific subtasks have already been executed.

However, when it comes to non-linear sequences of tasks (caused by variations of sub-tasks) and locations that cannot be uniquely identified (e.g. due to overlapping gaussians in our spatial model), performance decreases and the use of HHMMs can significantly improve results. Activity recognition based on HHMMs performs slightly worse than other ones based on sequences of object detections in a heavily sensor equipped environment (which often have precision and recall-rates around 90 % [2] or average accuracy rates around 97 % [4]), but given the unintrusive and cheap sensor setting, our approach works remarkably well as long as good models of the activities are provided to generate the HHMM. One shortcoming of our current experiments is the manual generation of models for the HHMMs and STPRs based on ground truth data due to a lack of training data. To prove the generality of the approach, a cross validation would be necessary but due to the lacking availability of datasets and the already heavily time-consuming task of generating new datasets, sufficient training data has not been available at the time of the experiments.

Using only sequences of spatial regions as features introduces the limitation that activities that consist of similar patterns of locations, which potentially

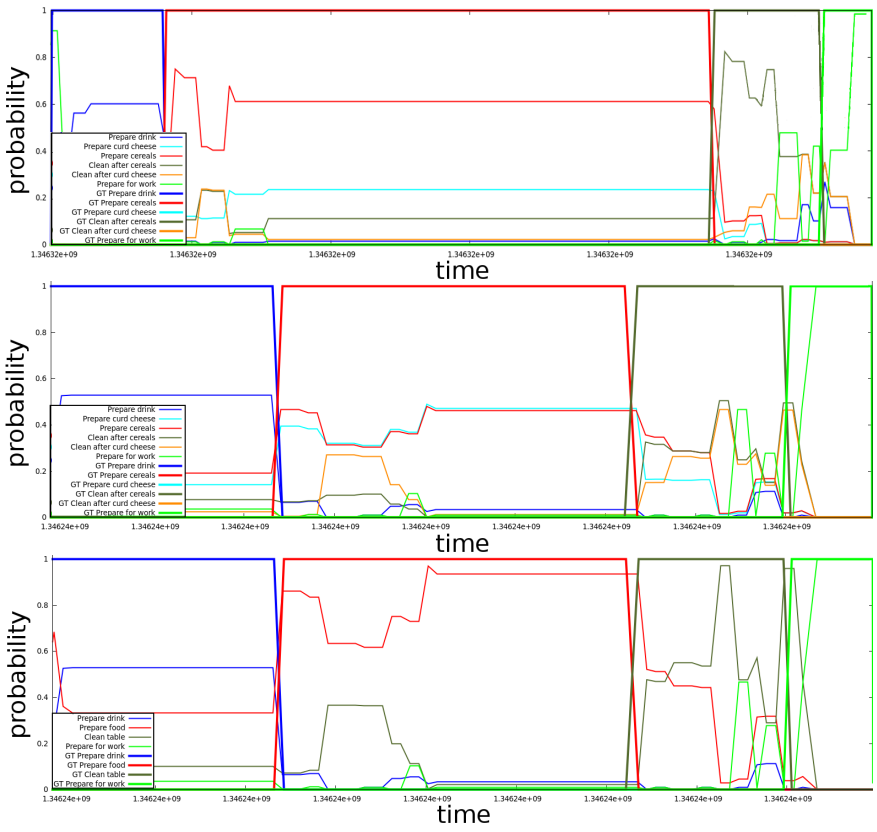

Fig. 4. Upper picture: Plan-probabilities over time of one morning routine of the simulated dataset using the online activity recognition without object detections. Middle picture: Plan probabilities for the real data. Here, the activity recognition has a hard time distinguishing between the different food preparing and cleaning activities due to their similarity. Lower picture: Plan probabilities of the same morning-routine as the middle picture, but in this case,the two food-preparing and the two clean-table activities are merged into a single one.

are close to each other, are hard to distinguish as we saw in the example of the different prepare-food and clean-table plans. The performance of our system may be improved to some extent with more features like object detections or by including prior information like the time and rough order in which activities are usually performed. One drawback of the use of HHMMs is the need for emission probabilities, which we currently learn in a training phase. We were able to estimate emission probabilities directly from the spatial model with comparable, but slightly less accurate results. Further investigations to estimate emission probabilities from the spatial model are part of our future work. We also intend to include non-activities into our HHMM by introducing an additional internal state that models non-labeled activities. Nevertheless, we were so far able to set up a live system for activity recognition using only data from one Kinect sensor (or two if object detections are used), thus offering a relatively cheap and non-intrusive way of performing activity recognition for mobile robots in spatially limited environments.

7 Conclusion

We presented an approach that performs activity recognition based on context dependent spatial regions in a kitchen environment using inexpensive depth cameras. We compared two different modeling and inference techniques and found that STPRs do not perform well in settings with different context dependent spatial regions that are located very close to each other due to a missing sensor model. To overcome this limitation and also account for variations in the partial order of activities, we set up a HHMM-based live system for activity recognition using a spatial model and evaluated it in a simulated and a real-world setting. The evaluations show that HHMM-based activity recognition with a Kinect using context dependent spatial regions outperforms STPRs and offers a decent approach for human activity recognition.

References

1. Nico Blodow, Lucian Cosmin Goron, Zoltan-Csaba Marton, Dejan Pangercic, Thomas Rühr, Moritz Tenorth, and Michael Beetz. Autonomous semantic mapping for robots performing everyday manipulation tasks in kitchen environments. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, September, 25–30 2011. Accepted for publication.
2. M. Buettner, R. Prasad, M. Philipose, and D. Wetherall. Recognizing daily activities with rfid-based sensors. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 51–60. ACM, 2009.
3. H.H. Bui, D.Q. Phung, and S. Venkatesh. Hierarchical hidden markov models with general state hierarchy. In *Proceedings of the National Conference on Artificial Intelligence*, pages 324–329. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004.

4. T.V. Duong, H.H. Bui, D.Q. Phung, and S. Venkatesh. Activity recognition and abnormality detection with the switching hidden semi-markov model. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 838–845. Ieee, 2005.
5. G. Echeverria, N. Lassabe, A. Degroote, and S. Lemaignan. Modular openrobots simulation engine: Morse. In *Proceedings of the IEEE ICRA*, 2011.
6. S. Fine, Y. Singer, and N. Tishby. The hierarchical hidden markov model: Analysis and applications. *Machine learning*, 32(1):41–62, 1998.
7. Stephen S Intille, Kent Larson, Emmanuel Munguia Tapia, Jennifer S Beaudin, Pallavi Kaushik, Jason Nawyn, and Randy Rockinson. Using a live-in laboratory for ubiquitous computing research. In *Pervasive Computing*, pages 349–365. Springer, 2006.
8. Michael Karg and Alexandra Kirsch. Acquisition and Use of Transferable, Spatio-Temporal Plan Representations for Human-Robot Interaction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.
9. Matthew Klenk, Nick Hawes, and Kate Lockwood. Representing and reasoning about spatial regions defined by context. In *AAAI Fall 2011 Symposium on Advances in Cognitive Systems*, 2011.
10. Lin Liao, Dieter Fox, and Henry Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *Int. J. Rob. Res.*, 26(1):119–134, 2007.
11. B. Logan, J. Healey, M. Philipose, E.M. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Proceedings of the 9th international conference on Ubiquitous computing*, pages 483–500. Springer-Verlag, 2007.
12. T.L. Mitzner, C.A. Smarr, J.M. Beer, T.L. Chen, J.M. Springman, A. Prakash, C.C. Kemp, and W.A. Rogers. Older adults' acceptance of assistive robots for the home. 2011.
13. N.T. Nguyen, H.H. Bui, S. Venkatsh, and G. West. Recognizing and monitoring high-level behaviors in complex spatial environments. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–620. IEEE, 2003.
14. N.T. Nguyen, D.Q. Phung, S. Venkatesh, and H. Bui. Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 955–960. IEEE, 2005.
15. Mike Perkowitz, Matthai Philipose, Kenneth Fishkin, and Donald J. Patterson. Mining models of human activities from the web. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 573–582. ACM, 2004.
16. D.Q. Phung, H.H. Bui, S. Venkatesh, et al. Hierarchical semi-markov conditional random fields for recursive sequential data. *Arxiv preprint arXiv:1009.2009*, 2010.
17. Freek Stulp, Andreas Fedrizzi, and Michael Beetz. Action-related place-based mobile manipulation. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2009.
18. J. Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 842–849. IEEE, 2012.
19. A. Tapus, M.J. Mataric, and B. Scassellati. Socially assistive robotics. *IEEE Robotics and Automation Magazine*, 14(1):35, 2007.

20. Moritz Tenorth, Lars Kunze, Dominik Jain, and Michael Beetz. KNOWROB-MAP – Knowledge-Linked Semantic Object Maps. In *Proceedings of 2010 IEEE-RAS International Conference on Humanoid Robots*, Nashville, TN, USA, December 6-8 2010.
21. Moritz Tenorth, Daniel Nyga, and Michael Beetz. Understanding and Executing Instructions for Everyday Manipulation Tasks from the World Wide Web. In *IEEE International Conference on Robotics and Automation (ICRA).*, pages 1486–1491, 2010.
22. D.J. Townsend and T.G. Bever. *Sentence comprehension: The integration of habits and rules*. The MIT Press, 2001.
23. M. Weiland, A. Smaill, and P. Nelson. Learning musical pitch structures with hierarchical hidden markov models. *Journees dInformatique Musical*, 2005.
24. Li Yujian and Liu Bo. A normalized levenshtein distance metric. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1091 –1095, june 2007.
25. H. Zender, O. Martínez Mozos, P. Jensfelt, G.J.M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 2008.