USER BEHAVIOR RECOGNITION FOR AN AUTOMATIC PROMPTING SYSTEM

A Structured Approach based on Task Analysis

Christian Peters, Thomas Hermann and Sven Wachsmuth
CITEC, Bielefeld University, Bielefeld, Germany

Keywords: User behavior recognition, Task analysis, Bayesian network, Bayesian filtering.

Abstract: In this paper, we describe a structured approach for user behavior recognition in an automatic prompting system that assists users with cognitive disabilities in the task of brushing their teeth. We analyze the brushing task using qualitative data analysis. The results are a hierarchical decomposition of the task and the identification of environmental configurations during subtasks. We develop a hierarchical recognition framework based on the results of task analysis: We extract a set of features from multimodal sensors which are discretized into the environmental configuration in terms of states of objects involved in the brushing task. We classify subtasks using a Bayesian Network (BN) classifier and a Bayesian Filtering approach. We compare three variants of the BN using different observation models (IU, NaiveBayes and Holistic) with a maximum-margin classifier (multi-class SVM). We present recognition results on 18 trials with regular users and found the BN with a NaiveBayes observation model to produce the best recognition rates of 84.5% on avg.

1 INTRODUCTION

Cognitive assistive technology aims at developing systems which support persons with cognitive disabilities in the execution of activities of daily living (ADLs). Such persons mostly have problems in accomplishing ADLs on their own and need assistance to perform such tasks successfully. Automatic prompting systems can provide assistance and keep elderly people or persons with cognitive disabilities further in their own homes which leads to an increase of independence of the persons and a relief of caregiver burden.

Recognizing ADLs is an important problem for the design of automatic prompting systems. However, there is currently no systematic approach to design recognition components for different daily activities like brushing teeth or preparing coffee, etc. Recognizing such activities is a challenging problem: Firstly, ADLs usually involve a variety of subtasks which can be combined in a flexible way to execute the task successfully. Hence, an automatic prompting system has to identify the different subtasks independently of their time of occurrence. Secondly, the variance in the execution of subtasks is huge, especially for persons with cognitive disabilities, but also for regular persons. Thirdly, the recognition procedure has to deal with a low number of training examples because obtaining training data in a complex, real-world scenario is very hard.

In this paper, we describe a structured approach to user behavior recognition in an automatic prompting system at the exemplary task of *brushing_teeth*. We are going from a systematic analysis of the task to the relevant activities and environmental states of objects involved in *brushing_teeth*. Therefore, we use Interaction Unit (IU) analysis proposed in (Ryu and Monk, 2009) as a method for qualitative data analysis. IU analysis is utilized for different design decisions: Firstly, we decompose the brushing task into important subtasks which we aim to recognize in our framework. Secondly, we extract environmental configurations in terms of states of objects manipulated during subtasks, e.g. the position of the mug or the towel. We discretize features extracted from sensory information into environmental configurations as an intermediate representation. We abstract from recognizing specific movements by tracking objects or the user's hands due to the huge variance in execution. Instead, we classify subtasks based on environmental configurations using a Bayesian Network (BN). Each time we observe an environmental configuration, we update our belief (probability distribution over subtasks) using Bayesian Filtering. In our recognition frame-

Peters C., Hermann T. and Wachsmuth S. (2012).
USER BEHAVIOR RECOGNITION FOR AN AUTOMATIC PROMPTING SYSTEM - A Structured Approach based on Task Analysis.
In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, pages 162-171
DOI: 10.5220/0003773601620171

work, we compare BNs with three different structures and, hence, different independence assumptions between variables: (i) *IU* where the structure is purely based on the results of IU analysis (ii) *NaiveBayes* where environmental states of objects obtained in the IU analysis are treated independently of each other (iii) *Holistic* where environmental states are treated as a single environmental observation. We underline that the results of IU analysis are integrated into the BN structure, either completely as for *IU* or partially as for *NaiveBayes* and *Holistic*. We compare our approach to a multi-class Support Vector Machine on a dataset of 18 brushing trials conducted by regular persons. Evaluating our framework with regular users is feasible in a first development cycle where data of persons with cognitive disabilities is very hard to acquire: We consider the target group in the development of the recognition framework because IU analysis is conducted on videos of persons with cognitive disabilities in a residential home setting. Since we abstract from the recognition of specific movements by tracking objects or the user's hands, data of regular users can be used for testing our framework because regular users show similar characteristics in the execution of the task. However, we aim to evaluate our framework with persons with cognitive disabilities in the future.

The remainder of the paper is structured as follows: Section 2 gives an overview of relevant related work. In section 3, we describe IU analysis as a method of qualitative data analysis. Section 4 shows the feature extraction process from sensory information. Our recognition framework is described in section 5, followed by results and a conclusion in section 6 and 7, respectively.

2 RELATED WORK

Recognizing complex activities in real-world scenarios is a crucial step in the development of cognitive assistive technology. Much work is done recognizing behaviors based purely on (i) visual sensors like cameras (Hoey et al., 2010) (Moore et al., 1999) and (ii) wearable sensors which are directly attached to the user (Yang et al., 2009) (Subramanya et al., 2006).

Several approaches based on visual sensors aim to recognize human behaviors using movement trajectories of objects or the user's hands (Hoey et al., 2010), (Moore et al., 1999), (Nguyen et al., 2005), (Pusiol et al., 2008). Due to the large variance in task execution, it is very hard to distinguish between similar user behaviors based on movement trajectories only. In this work, we abstract from recognizing movement trajectories, but classify user behaviors based on environmental states of objects involved in the brushing task. Object-based behavior recognition was done e.g. by (Wu et al., 2007) and (Patterson et al., 2005) using RFID-tagged objects and a Dynamic Bayesian Network (DBN) for classification. DBNs and Hidden Markov Models (HMMs) as a special variant of DBNs are widely used in user behavior recognition: (Oliver et al., 2002) use hierarchical HMMs for behavior recognition in an office environment, (Galata et al., 2001) classify dance movements with variable length Markov models. (Robertson and Reid, 2006) use HMMs to recognize user behaviors in an urban surveillance and a sports scenario.

Most approaches for user behavior recognition in such scenarios are modelled using common-sense knowledge without further analyzing the task and the recognition problem. Here, we apply a structured approach of retrieving relevant information on which we develop our recognition framework. We use Interaction Unit (IU) analysis proposed in (Ryu and Monk, 2009) as a method for qualitative data analysis to obtain both relevant user behaviors to be recognized as well as environmental configurations describing states of objects. IU analysis was used in a similar context in (Hoey et al., 2011) in order to facilitate the specification process of an automatic prompting system using a Partially Observable Markov Decision Process (POMDP). The following section describes how we use IU analysis in the development of our recognition framework.

3 INTERACTION UNIT ANALYSIS

User behavior recognition in an everyday task like brushing teeth is a challenging problem: The task consists of several subtasks that can be combined in a flexible order to execute the task successfully. The analysis of the task and the subtasks is an important step in the development of a recognition framework. In this work, we apply Interaction Unit (IU) analysis proposed in (Ryu and Monk, 2009). IU analysis models interaction by describing the conjunction of cognitive and environmental pre- and postconditions for individual actions. We apply IU analysis on 23 videos recorded at our cooperation partner *Haus Bersaba*, a residential home belonging to the clerical foundation *v. Bodelschwinghsche Stiftungen Bethel* in Bielefeld, Germany. In *Haus Bersaba*, users with cognitive disabilities like Alzheimer's Disease, Dementia, Autistic Spectrum Disorder, Epilepsy, etc. permanently live. Each video shows one trial of a user brushing his/her teeth while being observed and supported by a caregiver. The caregiver's assistance is needed if the user is not able to proceed in task execution. In this work, we are particularly interested in two aspects of IU analysis: Firstly, the decomposition of the task into subtasks which we will call *user behaviors* in the following. Secondly, the environmental conditions associated with the user behaviors. Table 1 shows the results of the IU analysis for brushing teeth. The brushing task is decomposed into seven user behaviors as described in column *UB*: *paste_on_brush*, *fill_mug*, *rinse_mouth*, *brush_teeth*, *clean_mug*, *clean_brush* and *use_towel*. Each user behavior is further subdivided into single steps described in column *UB steps*. *rinse_mouth* for example consists of three steps: mug is moved to the face, the user rinses his/her mouth and the user moves the mug away from the face. Column *Current Environment* shows the environmental states as a precondition of single user behavior steps. Performing the step then changes the environmental state, for example in the first step of *paste_on_brush*: The toothpaste tube is on the counter and taking the tube changes the toothpaste location to 'in hand'. We abstract from the recognition of single steps, but infer the user's behavior based on the environmental configuration which is expressed by states of objects manipulated during a behavior. From column *Current Environment*, we extract five discrete random variables describing important object states: *mug_position*, *towel_position*, *paste_movement*, *brush_movement* and *tap_condition*. Since using the toothpaste is not bound to a specific region, we abstract from the exact location and use the movement of the paste.

Table 1: Results of the IU analysis for brushing teeth. Column "UB" describes the different subtasks involved in the brushing task. Column "UB steps" lists the ideal steps to execute the according subtask. Column "Current Environment" shows the environmental configuration in terms of states of objects involved in a particular step. TT - toothpaste tube.

UB	Current Environment	UB steps
paste_on_brush	TT on counter	take TT from counter
	TT closed in hand	alter TT to open
	brush on counter	take brush from counter
	brush and TT in hand	spread paste on brush
	TT is open	alter TT to closed
	TT closed in hand	give TT to counter
	TT on counter, brush in hand	
fill_mug	mug empty	give mug to tap
	mug at tap, tap off	alter tap to on
	mug at tap, tap on	alter tap to off
	mug filled	
rinse_mouth	mug filled	give mug to face
	mug at face	rinse
	mug else	give mug to counter
	mug counter	
brush_teeth	brush with paste in hand	give brush to face
	brush at face	brush all teeth
	brush at face, teeth clean	take brush from face
	brush not at face	
clean_mug	mug dirty at counter	give mug to tap
	mug dirty at tap, tap off	alter tap to on
	mug dirty at tap, tap on	clean mug
	mug clean at tap, tap on	alter tap to off
	mug clean at tap, tap off	give mug to counter
	mug clean at counter	
clean_brush	brush dirty	give brush to tap
	brush dirty at tap, tap off	alter tap to on
	brush dirty at tap, tap on	clean brush
	brush clean at tap, tap on	alter tap to off
	brush clean at tap, tap off	give brush to counter
	brush clean at counter	
use_towel	towel at hook, mouth wet	give towel to face
	towel at face, mouth wet	dry mouth
	towel at face, mouth dry	give towel to hook
	towel at hook	

Table 2 shows the state variables and their according discrete values.

For *paste_movement* and *tap_condition*, a binary random variable with values no/yes and off/on, respectively, is adequate. For *brush_movement*, we have the states no, yes_sink and yes_face. The latter ones are important to discriminate between the user behaviors *paste_on_brush* and *brush_teeth* based on the movement of the brush. The values of the variables *mug_position* and *towel_position* are the different regions identified in column *Current Environment* where the mug and towel appear during task execution. *No_hyp* is used if no hypothesis about the mug/towel position is available.

In the following sections, we describe how we integrate the state space given in table 2 as an intermediate representation in our recognition framework: The decomposition of the task provides the important user behaviors we aim to recognize. We classify the user behaviors based on the environmental state space. We assess the state space on features extracted from sensory information described in the following section.

Table 2: Environmental state space with five discrete random variables extracted from the environmental configuration in table 1.

State Variable	Values
mug_position	counter, tap, face, else, no_hyp
towel_position	hook, face, else, no_hyp
paste_movement	no, yes
brush_movement	no, yes_sink, yes_face
tap_condition	off, on

4 SENSORS AND FEATURE EXTRACTION

We built a washstand setup which we equipped with a set of sensors for environmental perception. We use a combination of unobtrusive sensors installed in the environment and tools in order to extract features from which we assess the environmental configuration. We don't attach any wearable sensors to the user directly, because we don't want to disturb in task execution. The following list gives an overview of the sensors.

Cameras. We use two cameras observing the scene from both overhead and frontal perspective as shown in figure 1. The cameras grab images of resolution 480x640 (overhead) and 500x620 (frontal) with 30Hz each.

Flow Sensor. A binary flow sensor is installed at the water pipe. The sensor measures whether the water flow is off or on.

9-dof Brush Module. We equipped the toothbrush with a 9-dof module including a 3-axis accelerometer measuring gravitational acceleration, a 3-axis gyroscope measuring orientation of the brush and a 3-axis magnetometer measuring the earth's magnetic field.

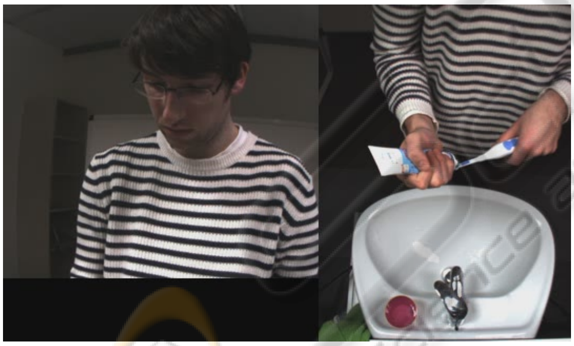

Figure 1: Example images of the frontal camera (left) and overhead camera (right), respectively.

As a first step of the recognition framework features extracted from sensory information are discretized into environmental state variables given in table 2. In order to assess the values of the state variables, we extract a set of 19 features: 4 features each for the position of the mug and the towel, 1 feature indicating the water flow, 1 feature describing the movement of the paste and 9 features for the movement of the brush. The feature extraction is described in detail in the following: From the camera images, we extract the position of the mug and towel using a color distribution detector. Since the procedure is similar for both objects, we describe the detection of the mug exemplarily: The color distribution of the objects is learned in a supervised manner based on sample images. The result of the detector is a hypothesis about the object's position for each camera image given by a bounding box. We use the x and y coordinate of the bounding box center as features. Since we have a detector on both the overhead and frontal image, we have 4 features (x and y position on frontal and overhead image) for the mug. The 4 features for the towel are calculated accordingly. For estimating the movement of the paste, we found the number of edge pixels in the counter region of the overhead image to be a valuable feature: If the paste is in the counter region, the paste produces an increased number of edge pixels compared to the case when the paste is outside the counter region due to a manipulation by the user. Hence, we employ the number of edge pixels in the counter region as an indicator whether the paste is moving. Since we have a static camera, the counter region is predefined in our setup. The flow sensor returns a binary feature with 1 indicating water flow and 0 indicating no water flow. For brush movement detection, we extract 9 features from the brush module: acceleration, orientation and the earth's magnetic field in x,y and z direction each.

In the following section, we describe our recognition framework in detail.

5 APPROACH

The IU analysis decomposes a task into different user behaviors. Each user behavior is further subdivided into single steps which are described in terms of environmental states. Hence, the IU analysis structures the task into a hierarchy of user behaviors and combines semantic information about the user behavior with environmental states. In our approach, we make use of the hierarchical structure obtained in the IU analysis: We use a two-layered framework for user behavior recognition modeling the hierarchical structure as shown in figure 2. The features $f_1 ... f_{19}$ described in the previous section are discretized into an intermediate representation of state space variables

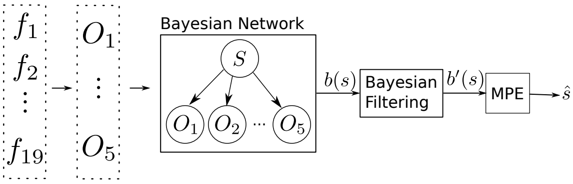

Figure 2: Overview of the hierarchical classification framework. See text for a detailed description.

$O_1...O_5$ given in table 2. A Bayesian Network classifies user behaviors (denoted with variable S) based on the state space variables into a belief b, a discrete probability distribution over user behaviors. We apply Bayesian Filtering to update b to a consecutive belief b'. In every time step, we choose the most probable explanation from b' which is the user behavior $\hat{s}$ with the highest probability. In the following sections, we describe the feature discretization and the Bayesian Network classification using a Bayesian Filtering in more detail.

5.1 Feature Discretization

We discretize the features $f_1...f_{19}$ into the five discrete random variables $O_1...O_5$ which correspond to the variables *mug_position*, *towel_position*, *paste_movement*, *brush_movement* and *tap_condition* described in table 2. The variables encode the environmental configuration obtained in the IU analysis which is described in table 1. We will denote $O_1...O_5$ as *observation variables* in the following. Each observation variable O_i is estimated using a single classification scheme:

mug_position. We determine the value of *mug_position* using the results of the mug detector. The detector provides a hypothesis in terms of the center position (x,y) of a bounding box for each camera image. Since we have a mug detector on the frontal and overhead image, we choose the hypothesis with the highest confidence in order to get a single hypothesis. The confidence is the number of pixels in the bounding box agreeing to the learned color distribution. We compare the (x,y)-position of the chosen hypothesis to a predefined set of static image regions which are the *counter*, *tap* and *face* region. In our approach, the whole frontal image is the *face* region since we don't explicitly detect the user's face due to two reasons: Firstly, the user's face is occluded by objects or the user's hands for a certain amount of time during the brushing task which makes the face recognition error-prone. Secondly, some user's lean forward during the brushing task. Hence, their faces disappear completely from the frontal image which makes a face recognition unreasonable. Every image point not in the *counter*, *tap* or *face* region is considered for the *else* region. If the center point is in one of the regions, the variable *mug_position* is set to the according value. If the detector doesn't return a hypotheses for both images, the variable is set to *no_hyp*.

towel_position. The position of the towel is determined similarly to the position of the mug. The static image regions used for towel detection are the *face* and *else* region as described in the previous section. Additionally, two areas left and right of the *counter* region are treated as a common region *hook* where the towel is usually hang up. Similar to *mug_position*, we set *towel_position* to *no_hyp* if the detector does not produce a valid hypothesis for one of the images.

paste_movement. If the paste is in the *counter* region, the number of edge pixels are increased compared to the case when the paste is outside the *counter* region due to a manipulation by the user. Hence, the number of edge pixels in the *counter* region indicates whether the paste is used or not. *Paste_movement* is detected by simply thresholding this number: If the number of edge pixels is below the threshold t_1, the variable is set to *yes*, otherwise *no*.

brush_movement. The movement of the brush is classified into the values *no*, *yes_face* and *yes_sink* using a two-step classifier: Firstly, we classify whether the brush is moving at all: We compare the orientation of the brush given by the 3-dimensional gyroscope data at time t with the mean orientation over the last 3 time steps. If the difference is above a threshold t_2, we estimate that the brush is moving. In order to distinguish between the user behaviors *brush_teeth* and *paste_on_brush* (the brush is moving in both behaviors), we use a Support Vector Machine (SVM) for a more fine-grained classification into the two classes *yes_sink* and *yes_face*: The input features are the 9-dimensional features from the brush module: acceleration, orientation and the earth's magnetic field in x, y and z direction each. The SVM with an RBF kernel is trained with a leave-one-out cross-validation scheme on manually labeled sample data. The γ and C parameters of the RBF kernel are calculated using an extensive grid search in the parameter space.

tap_condition. The discretization of the *tap_condition* is trivial: If the flow sensor returns 0, *tap_condition* is set to *off*, otherwise to *on*.

We classify user behaviors based on the discretized observation variables using a Bayesian Network and a Bayesian Filtering approach as described in the following section.

5.2 Bayesian Network Classification

In our framework, we aim to recognize the user behaviors obtained from the IU analysis in table 1. We subsume the user behaviors *fill_mug* and *clean_mug* to a common user behavior *rinse_mug* because the relevant observation variables as well as the according states are nearly identical for both user behaviors. In a regular trial, user behaviors don't follow exactly on each other, but mostly alternate with transition behaviors, for example the user's hand approaches or leaves a manipulated object. We consider these transition behaviors by adding a user behavior *nothing* which we treat as any other user behavior in our recognition model. In this work, we use a Bayesian Network (BN) to classify user behaviors based on the observations $O_1 ... O_5$. A BN models a joint probability distribution of random variables. Conditional independence relations between variables are modelled using a directed acyclic graph. A BN is ideally suited to model the structural relations between user behaviors denoted by the random variable S and relevant observation variables $O_1 ... O_5$. On a higher level, the inclusion of an observation variable O_i in the BN of behavior s describes that O_i is relevant for classifying behavior s. The relevance relations arise from the results of IU analysis: For example for *paste_on_brush*, relevant observation variables according to table 1 are *paste_movement* and *brush_movement*. All other observation variables are not relevant according to IU analysis and are not regarded in the BN. The BN for *paste_on_brush* is shown in figure 3 (a). For each user behavior, we maintain a BN with a structure according to table 3 where we list relevant observation variables for each user behavior. IU analysis does not provide a structural relationship for behavior *nothing* because IU analysis doesn't model transition behaviors explicitly as user behaviors. Since the transition behaviors can occur between all behaviors, we declare all observation variables relevant for classifying *nothing*. We denote the approach using relevance relations in the BN structure as *IU*-based. We compare the *IU-based* approach with a *NaiveBayes* approach where all observation variables are relevant for each user behavior and each observation variable O_i is conditionally independent given the user behavior:

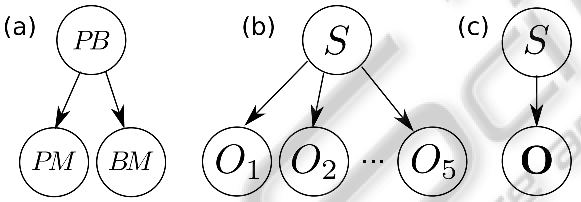

Figure 3: Bayesian Networks with three different structure: (a) IU-based structure: Example BN for user behavior *paste_on_brush*: *BM* and *PM* are brush and paste movement, respectively. (b) NaiveBayes (c) Holistic.

Table 3: User behaviors and relevant observation variables according to IU analysis.

User Behavior	Relevant Observation Variables
paste_on_brush	paste_mov, brush_mov
rinse_mug	mug_pos, tap_cond
rinse_mouth	mug_pos
brush_teeth	brush_mov
clean_brush	brush_mov, tap_cond
use_towel	towel_pos
nothing	all obs. variables

$$P(O_1, ..., O_5, S) = \prod_{i=1}^{5} P(O_i|S) \qquad (1)$$

The BN with *NaiveBayes* structure shown in Figure 3 (b) has the ability to deal with small training sets since the probability of each O_i depends only on the user behavior S. This is important in our work, because some user behaviors like *clean_brush* are rare compared to other behaviors and the acquisition of data in our scenario is very hard. A disadvantage of *NaiveBayes* is the influence of irrelevant variables in the classification of behaviors. The position of the mug for example should be irrelevant for the classification of *use_towel*. Hence, the probability of *mug_position* might be very low and decreases the overall probability in the product in equation 1. We try to overcome this side effect by using a BN with a *Holistic* structure: We subsume the observations $O_1 ... O_5$ in a vector $\mathbf{O}$ and treat $\mathbf{O}$ as a single observation as shown in figure 3 (c). The BN with *Holistic* structure is prone to faulty observations which happen occasionally in the discretization of features into observation variables. In the *Holistic* BN, faulty observations lead to rapid changes in the belief b from one time step to the next. This is not desirable in our scenario, because transitions between user behaviors are rather smooth due to the nature of the task. Hence, we extend our framework with a transition model which takes into account the belief of the preceding time step. This results in a Bayesian Filtering approach similar to the forward algorithm in a Hidden Markov Model as the simplest type of a Dynamic Bayesian Network. The belief b is updated to a consecutive belief b' for each user behavior s' as shown in equation 2:

$$b'(s') = \frac{O(s', o) \cdot \sum_{s \in S} T(s', s) \cdot b(s)}{C} \qquad (2)$$

with the normalization term $C = \sum_{s' \in S} O(s', o) \cdot \sum_{s \in S} T(s', s) \cdot b(s)$. $O(s', o)$ is the probability of making observation o when the user behavior is s'. For the *IU-based* approach, $O(s', o) = \prod_{i_{s'}} P(O_{i_{s'}}|s')$ with $i_{s'}$ are the variables i which are relevant for user behavior s' according to IU analysis. For *NaiveBayes* and *Holistic*, $O(s', o) = \prod_{i=1}^{5} P(O_i|s')$ and $O(s', o) =$

$P(\mathbf{O}|s')$, respectively. $T(s', s) = P(s'|s)$ is the probability of a state transition from user behavior s to user behavior s'. The observation model $O(s', o)$ is learned on manually annotated training data using Maximum Likelihood (ML) estimation.

$$P_{ML}(O_i = o_j | S = s_k) = \frac{n_{ij}}{N_{ik}} \tag{3}$$

where n_{ij} is the number of observations of variable i with value j and N_{ik} is the number of observations of variable i when user behavior is k. We apply a leave-one-trial-out cross validation scheme to estimate the parameters: The test set consists of data of a single trial and the residual data forms the training set. Learning the transition model T from data is similar to learning the observation model:

$$P_{ML}(S = s' | S = s) = \frac{n_{ss'}}{N_{s'}} \tag{4}$$

where $n_{ss'}$ is the number of transitions from user behavior s to s' and $N_{s'}$ is the total number of transitions to s'. The ML estimation results in a very peaked state transition distribution: The probability of self transitions is very high. Transitions from one state to another have a very low probability because the number of occurrences of different user behaviors is small compared to the length. Hence, the transition model in equation 2 leads to smooth state transitions between user behaviors because single faulty observations can't rapidly change the entire belief from one time step to the next. This is a desirable behavior in our system.

The advantage of the proposed hierarchical framework is two-fold: Firstly, we can combine trivial and sophisticated classifiers in the discretization of features into observation variables since each observation variable can be treated independently. Secondly, classifiers can easily be exchanged or added to our framework for different tasks analyzed with IU analysis.

We compare our hierarchical framework using observation variables as an intermediate representation with a completely feature-driven classification approach using a multi-class Support Vector Machine as described in the following section.

5.3 Multi-Class SVM

We use a multi-class Support Vector Machine (MC-SVM) for classifying user behaviors. The MC-SVM method classifies the 19-dimensional feature vector $f_1 \dots f_{19}$ into the six user behaviors given in table 1 and behavior *nothing*. The features are normalized to the interval $[-1, 1]$. The SVM with an RBF kernel is trained with a leave-one-trial-out cross-validation scheme on manually labeled sample data. The parameters γ and C of the RBF kernel are calculated using an extensive grid search in the parameter space. The classical MC-SVM predicts the output label for a given test sample. In our recognition framework, we use a Bayesian Filtering approach based on the probability distribution over user behaviors. In order to compare MC-SVM to our approach, we apply an adapted form of MC-SVM proposed in (Wu et al., 2004) which provides probability estimates over classes instead of a single class label.

6 RESULTS

We show results for the approaches described in the previous section on a dataset of 18 trials. Each trial shows a single brushing task. In this work, the trials are performed by regular users. Since we abstract from the recognition of specific movements by tracking objects or the user's hands, data of regular users can be used for evaluating our framework in a first development cycle because regular users show similar characteristics in terms of a flexible and highly user-dependent execution of the task. However, we aim to conduct experiments with persons with cognitive disabilities in the future. The 18 trials were performed by 9 users where 2 users conducted 4 trials each, 3 users conducted 2 trials each and 4 users conducted a single trial each. Table 4 shows the total number of occurrences for each user behavior in the trial data. Since *nothing* usually alternates with any other user behavior in a regular trial it occurs much more frequently than any other behavior. Besides the total number of occurrences, the average lengths of user behaviors vary extremely as shown in table 4: *rinse_mouth* for example has an average length of 58 frames compared to *brush_teeth* with an average of 1980 frames. Furthermore, this results in a huge difference in the number of training data for each user behavior. The huge variance in average lengths as well as small amount of training data for certain be-

Table 4: Table shows for each user behavior the total number of occurrences ”nrUB”, total number of frames and the average length ”nrFrames (avg)” in the trial data.

User Behavior	nrUB	nrFrames (avg)
paste_on_brush	16	1922 (120)
rinse_mug	20	1818 (91)
rinse_mouth	29	1681 (58)
brush_teeth	18	35641 (1980)
clean_brush	24	3428 (143)
use_towel	13	1922 (148)
nothing	137	22426 (164)

haviors make the recognition problem very challenging. Table 5 shows the classification rates for individual user behaviors and average rates. **IU**, **NB** and **HO** denote the three approaches using a Bayesian Network with different structures: For **IU**, the structure is given by the relevant observation variables for each user behavior obtained from the IU analysis, **NB** is the NaiveBayes and **HO** denotes the holistic structure as described in section 5.2. **SVM** denotes the approach using a multi-class Support Vector Machine (section 5.3). We compare the approaches with two different variants of our recognition framework: **OT** describes the Bayesian Filtering variant where the belief b is updated according to equation 2 using the observation and the transition model as mentioned in section 5.2. **O** describes the variant using the Bayesian Network without a transition model between behaviors. The average classification rates for NB, HO and SVM are slightly increased in OT compared to O except for IU where rates are similar. Obviously, the transition model can deal with faulty observations by suppressing rapid belief changes from one time step to the next which increases the classification rates. Instead, the transition model favors smooth belief changes which is desirable for our system due to the nature of the underlying task. In the following, we concentrate on the analysis of OT to compare the approaches in more detail.

Table 5: Comparison of classification rates for the different approaches: OT - Bayesian Filtering, O - Bayesian Network classifier without transition model. IU - BN structure obtained in IU analysis, NB - NaiveBayes, HO - Holistic denote the different BN structure approaches, SVM - multi-class Support Vector Machine, RMg - rinse_mug, UT - use_towel, PB - paste_on_brush, RMth - rinse_mouth, BT - brush_teeth, CB - clean_brush, N - nothing.

approach		RMg	UT	PB	RMth	BT	CB	N	Avg
O	IU	73.1	88.1	75.2	81.4	**79**	48.1	25.9	67.3
	NB	**91.6**	**89.5**	**90.4**	**91**	71.3	**87.6**	47.4	**81.3**
	HO	74.2	89.1	77.9	75.7	72	84.1	49.5	74.7
	SVM	72.4	71	66.2	75.1	53	77.1	**54.4**	67
OT	IU	70.4	91.3	72.1	80.3	**89.9**	38	28.6	67.2
	NB	**94.1**	**95.1**	**85.4**	**95.7**	81.7	**88.8**	50.4	**84.5**
	HO	79	94.4	75.3	79.3	81.9	85.5	55	78.6
	SVM	64.5	68.1	64	55.1	82.9	61.1	**74**	68.2

The NB method leads to the highest classification rates with an average of 84.5%. With 68.2%, the accuracy of the SVM method is similar to the IU method with 67.2%, but worse than NB with 84.5%. The excellent result for the NB method shows that our systematic approach for user behavior recognition based on IU analysis is feasible: In the NB method as well as the HO and IU method, the results of the IU analysis are integrated in terms of environmental configurations on which the classification of user behaviors is based.

For IU, NB, HO and SVM, the classification rates for *nothing* are highly decreased in comparison to other user behaviors, especially in IU where the structure of the Bayesian Network classifier is obtained in terms of relevant observation variables. For *nothing*, all observation variables are relevant which decreases the classification rate for *nothing* compared to other user behaviors where IU analysis provides a specific set of relevant variables for each user behavior. The classification rates for single user behaviors are also decreased in IU compared to NB and HO. Furthermore, user behaviors that have an equal or similar set of relevant observation variables are mixed up in the IU method as shown in the confusion matrix in table 6. *clean_brush* (for which brush_movement and tap_condition are relevant) is misclassified as *brush_teeth* (brush_movement) with 34.2% and *paste_on_brush* (brush_movement and paste_movement) with 24.1%. Obviously, the residual variables not in the set of relevant variables seem to be important for distinguishing user behaviors where the same set of objects are manipulated. The confusion matrix of NB given in table 7 underlines the assumption: *clean_brush* is misclassified as *brush_teeth* with only 1.3% and *paste_on_brush* with 3.5%. Both NB and HO make use of the full information available by incorporating all observation variables for each user

Table 6: Confusion matrix for BN classifier with IU structure in the Bayesian Filtering approach. See table 5 for abbreviations.

	RMg	UT	PB	RMth	BT	CB	N
RMg	70.4	1.4	6.3	3.5	2.4	16.1	0
UT	0	91.3	5.9	0	1.1	0	1.6
PB	0	1.4	72.1	0	21.6	0	4.8
RMth	0	0	6.4	80.3	13.3	0	0
BT	0	0	10.1	0	89.9	0	0
CB	0	3.6	24.1	0	34.2	38.1	0
N	1.7	7.1	24.9	12.7	23.5	1.6	28.6

Table 7: Confusion matrix for BN classifier with NB structure in the Bayesian Filtering approach. See table 5 for abbreviations.

	RMg	UT	PB	RMth	BT	CB	N
RMg	94.1	0	0.1	1.2	0	2	2.5
UT	0.1	95.2	1	0	0.1	0	3.7
PB	0	0.6	85.4	0.1	6.8	0	7.1
RMth	1.1	0	0.2	95.7	1.2	0	1.8
BT	0.2	0	12.2	0.5	81.7	0	5.4
CB	0.1	0	3.5	6.3	1.3	88.8	0.1
N	1.9	5	14.6	8.9	15.7	3.8	50.4

behavior. However, NB has a higher average rate with 84.5% compared to HO with 78.6% as shown in table 5. Apparently, NB is more suited to deal with small amounts of training data for certain user behaviors in our scenario: Due to the conditional independence assumption in the NB approach, the probabilities for each observation variable given the user behavior can be calculated independently of the other observation variables. This leads to a more accurate prediction of the underlying probabilities and a higher classification rate compared to the HO method where the observation variables are subsumed in a single observation. As shown in table 5, the NB produces excellent classification results for single user behaviors showing a huge difference in length according to table 4. The classification rates range from 81.7% for *brush_teeth* to 95.7% for *rinse_mouth*. Only *nothing* has a decreased rate of 50.4%.

Our results show that our hierarchical classification framework based on the results of IU analysis is well suited to approach the recognition problem in our scenario. Our framework can deal well with the specific requirements of small amounts of training data for certain behaviors and arbitrary behavior lengths.

7 CONCLUSIONS

In this paper, we focus on the challenging problem of user behavior recognition in a real-world scenario. We use a structured approach to develop a recognition framework for an automatic prompting system assisting persons with cognitive disabilities in the everyday task of brushing teeth. We analyze the task using IU analysis. We identify user behaviors which are important to complete the task successfully as well as environmental configurations of objects involved in the task. User behaviors are classified based on environmental configurations using a Bayesian Network (BN) in a Bayesian Filtering approach.

We present recognition results on 18 trials performed by regular users. In future work, we aim to test our recognition framework in a study with persons with cognitive disabilities. An average recognition rate of 84.5% using a BN with a *NaiveBayes* structure shows that our framework is suitable to user behavior recognition for an automatic prompting system in a complex real-world scenario: The Bayesian Filtering approach can deal with the specific requirements like small amount of training data for user behaviors and arbitrary behavior lengths. Furthermore, the framework is applicable to other tasks and easily extendable with different classifiers due to the hierarchical structure.

ACKNOWLEDGEMENTS

The authors would like to thank the inhabitants and caregivers of Haus Bersaba for their high motivation to cooperate in the project 'Task Assistance for Persons with Cognitive Disabilities' (TAPeD) of the Cognitive Interaction Technology Center of Excellence (CITEC), Bielefeld University.

REFERENCES

Galata, A., Johnson, N., and Hogg, D. (2001). Learning variable-length Markov models of behavior. *Computer Vision and Image Understanding*, 81(3):398–413.

Hoey, J., Ploetz, T., Jackson, D., Monk, A., Pham, C., and Olivier, P. (2011). Rapid specification and automated generation of prompting systems to assist people with dementia. *Pervasive and Mobile Computing*, 7(3):299 – 318.

Hoey, J., Poupart, P., Bertoldi, A. v., Craig, T., Boutilier, C., and Mihailidis, A. (2010). Automated handwashing assistance for persons with dementia using video and a partially observable markov decision process. *Computer Vision and Image Understanding*, 114:503–519.

Moore, D., Essa, I., and Hayes, M. (1999). Object Spaces: Context Management for Human Activity Recognition. In *AVBPA'99, 2nd Int. Conf. on Audio-Visual Biometric Person Authentication*, Washington, DC.

Nguyen, N., Phung, D., Venkatesh, S., and Bui, H. (2005). Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. In *CVPR'05, Int. Conf. on Computer Vision and Pattern Recognition*, volume 2.

Oliver, N., Horvitz, E., and Garg, A. (2002). Layered representations for human activity recognition. In *ICMI'02, Int. Conf. on Multimodal Interfaces*, Pittsburgh, PA.

Patterson, D. J., Fox, D., Kautz, H., and Philipose, M. (2005). Fine-grained activity recognition by aggregating abstract object usage. In *ISWC'05, IEEE Int. Symposium on Wearable Computers*, Washington, DC, USA.

Pusiol, G., Patino, L., Bremond, F., Thonnat, M., and Suresh, S. (2008). Optimizing Trajectories Clustering for Activity Recognition. In *MLVMA'08, 1st Int. Workshop on Machine Learning for Vision-based Motion Analysis*, Marseille, France.

Robertson, N. and Reid, I. (2006). A general method for human activity recognition in video. *Computer Vision and Image Understanding*, 104:232–248.

Ryu, H. and Monk, A. (2009). Interaction Unit Analysis: A New Interaction Design Framework. *Human-Computer Interaction*, 24(4).

Subramanya, A., Raj, A., Bilmes, J. A., and Fox, D. (2006). Recognizing activities and spatial context using wearable sensors. In *UAI '06, 22nd Conference on Uncertainty in Artificial Intelligence*, Cambridge,MA, USA.

Wu, J., Osuntogun, A., Choudhury, T., Philipose, M., and Rehg, J. (2007). A scalable approach to activity recognition based on object use. In *ICCV'07. IEEE 11th International Conference on Computer Vision*.

Wu, T.-F., Lin, C.-J., and Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *Machine Learning Research*, 5:975–1005.

Yang, A. Y., Jafari, R., Sastry, S. S., and Bajcsy, R. (2009). Distributed recognition of human actions using wearable motion sensor networks. *Ambient Intelligence and Smart Environments*, 1:103–115.