Semi-supervised learning of a POMDP model of Patient-Caregiver Interactions

Jesse Hoey, Dept. of Computer Science, University of Toronto, Toronto, Canada

Pascal Poupart, School of Computer Science, University of Waterloo, Waterloo, Canada

Craig Boutilier, Dept. of Computer Science, University of Toronto, Toronto, Canada

Alex Mihailidis, Dept. of Occupational Therapy, University of Toronto, Toronto, Canada

Abstract

This paper presents a decision theoretic model of interactions between assistive technology and users during activities of daily living. The model is a partially observable Markov decision process whose goal is to monitor a user, assist the user during each activity, maintain indicators of overall user health, and adapt to changes. The key idea behind the model is that it is relatively easy to specify, and can be applied to many activities of daily living with little modification. The key contribution of this paper is to show how such a model can be learned without knowing the classes of behaviors of the user a priori. *This semi-supervised learning will enable assistive technologies to be applied ubiquitously for many different activities. We give some results from a preliminary version of the model for the task of handwashing.*

1 Introduction

Older adults living with cognitive disabilities (such as Alzheimer's disease or other forms of dementia) have difficulty completing activities of daily living (ADLs), and are usually assisted by a human caregiver who prompts them when necessary. Assistive technology will allow this elderly population to age-in-place by non-invasively monitoring users in their homes during ADLs, providing guidance or assistance when necessary [15].

The user's progress in an ADL is characterised by the interaction between three principal elements. We will illustrate these elements using the example of a person with moderate dementia being aided by a human caregiver during handwashing, as shown in Figure 1. First, the *task state* is a characterisation of the high-level state of the user, and is related to the goals in the task. For example, the handwashing ADL can be described by task variables such as *hands wet* and *hands dirty* which characterise the set of task states. Second, the *behavior* of the user is the course of action the user takes to change the task state. Common behaviors in the handwashing ADL may be things like *rinsing hands* or *using soap*. Third, the *caregiver's action* is what the caregiver does to help the user through the ADL. In Figure 1, the caregiver prompts the patient to use the soap prior to frame 1395. The patient duly follows the prompt and performs a behavior of putting soap on their hands up to frame 1745. This behavior causes a change in the task state: the patient's hands become soapy. The *task state* is what we are really interested in: it tells us about the progress the user is making in the task, and a utility function can be defined over it.

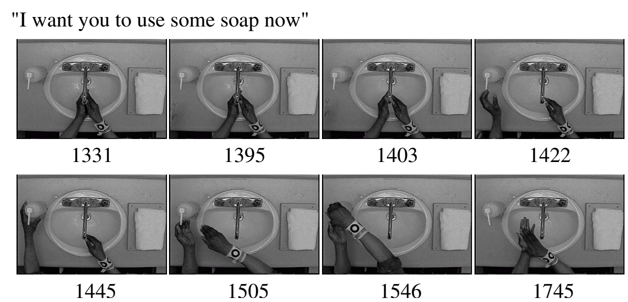


Figure 1: Example sequence in which a patient is prompted to put soap on their hands.

Our goal is then to design a system that will monitor the *task state* by observing the *behavior* of the user as a reaction to the *actions* of the caregiver. Assuming the *task state* is defined, the standard approach is to first use expert knowledge to define the behaviors that will occur in the ADL, and to train a classifier to recognise these statically defined behaviors using supervised learning. A second learning phase then builds a model of the relationship between the behaviors and the task. The problem with this approach is the need to specify behaviors *a priori*. This is a labor-intensive task requiring expert knowledge. Further, different individuals perform the same behaviors in different ways, exhibit different behaviors to perform the same task, and change their behaviors over time, usually as a result of their changing state of health. These considerations make it very difficult in many cases to define a single set of recognisable behaviors, a fact that is emphasised in the rehabilitation literature extensively [15]. Further, the recognition of all possible behaviors (as specified by some expert), may be computationally wasteful during interaction with particular users, who may only require assistance for a small number of aspects of a task. A system that can learn which behaviors are necessary to recognise in order to achieve task completion avoids this wasteful recognition process.

Our approach is to use a learning method that *discovers* the behaviors that are being exhibited, and learns their relationship to the task simultaneously. These relationships allow the model to predict the state of the task, and to monitor the user's progress. Ultimately, these predictions will be used to optimize an automated prompting strategy by maximising some notion of utility over the possible outcomes given visual observations of the user. In this paper, we approach the intermediate goal of learning the model that can monitor the progress of the user in the ADL from video observations. As such, the rewards and costs in the POMDP will not play a role in the work described herein.

Our previous work in this domain is on the specification and solution of a particular POMDP model [3], but ignores the learning problem by using only prior knowledge of the domain. This previous work showed how the domain can be modeled using a POMDP, how approximate solutions to the model can yield high quality policies of action, and how these policies can be used in a clinical setting [3]. The focus of the current paper is to discuss the associated learning problem.[1]

The learning method we present has the dual advantage of not requiring extensive *a priori* knowledge and training, and of being able to adapt to different users in different situations. The model is a partially observable Markov decision process (POMDP), with an observation function that relates entire video sequences to *behaviors* using a dynamic Bayesian network. The learning method clusters video sequences in training data in which only the *task* states are labeled, thereby learning a set of *behaviors*, and the relationship of the learned behaviors to the *task* states. This approach emphasises the notion that the behaviors are not actually what we are interested in recognising. Rather, we would like to predict the task state, because then we can choose actions that optimize expected return from some utility function defined over the task states. Eventually, these models can be applied to different ADLs by only changing the *task states*, which are more readily(i.e., objectively) definable than behaviors by human designers [12].

The paper first describes a general POMDP model for the patient–caregiver interaction during ADLs, including the observation function and the methods for learning the parameters of the POMDP. A simplified version for the handwashing task is then presented, followed by results on 17 sequences of video of a patient being prompted by a human caregiver. We finish the paper by describing related work and discussing future directions for this research.

[1]In fact, we are only discussing the learning of model parameters. Learning or eliciting the reward function is the subject of our current research.

2 General Model

A discrete-time POMDP consists of: a finite set S of states; a finite set A of actions; a stochastic transition model $\Pr : S \times A \to \Delta(S)$, with $\Pr(t|s,a)$ denoting the probability of moving from state s to t when action a is taken; a finite observation set Z; a stochastic observation model with $\Pr(z|s)$ denoting the probability of observation z in state s; and a reward function assigning real-valued reward $R(t,a)$ with taking action a and transiting to state t. Figure 2(a) shows the POMDP as a Bayesian network. Given a specific POMDP, the goal is to find a *policy* that maximizes the expected discounted sum of rewards attained by the system. The system state is not known with certainty, and therefore a policy maps either *belief states* (i.e., distributions over s) or action-observation *histories* into choices of actions. We refer to [16] for an overview of POMDP concepts and algorithms. Our research into finding policies of actions for specific POMDPs of this type is described in [3, 11]. In this paper, we address the associated problem of learning the transition and observation functions.

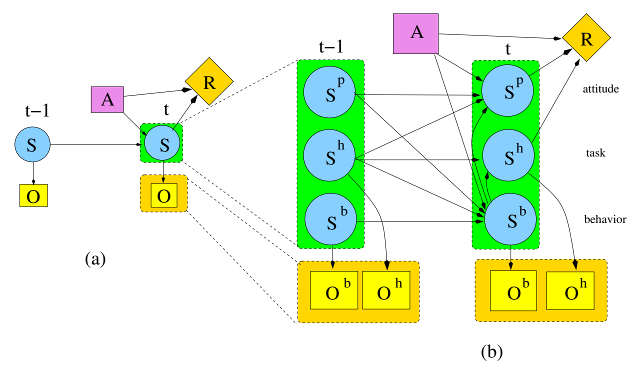


Figure 2: (a) Two time slices of general POMDP. The state, S, is modified by action A, and produces observation O. (b) Two slices of factored POMDP for ADL understanding. The state, S, is factored into three sets of variables, *task* (S^h), *attitude* (S^p) and *behavior* (S^b). Conditional independencies have been introduced.

Figure 2(b) shows the same model, except that the state, S, has been factored into three sets of variables: *task* (S^h), *attitude* (S^p) and *behavior* (S^b). During handwashing, the *task* variables may indicate whether the water is flowing, or whether the hands are soapy. The cognitive state of the user, including the level of dementia and the current responsiveness, for example, is given by a set of user *attitude* variables. The *attitude* variables essentially describe internal properties of the user that generalise across tasks. The task states and the user's attitude are changed by the user's *behavior*, S^b. *Task* and *behavior* variables generate observations, O^h and O^b, respectively. The transition function here, $P(S_t|S_{t-1}, A_{t-1}) =$

$P(S_t^b, S_t^p, S_t^h | S_{t-1}^b, S_{t-1}^p, S_{t-1}^h, A_{t-1})$, is written as a product of three terms, as follows. $P(S_t^b | S_{t-1}, A_{t-1})$ gives the expected behavior of the user given the previous state and the system action. $P(S_t^p | S_t^b, S_{t-1}^p, S_{t-1}^h, A_{t-1})$ gives the expected user state given the current behavior, the previous attitude and task states and the system action. $P(S_t^h | S_{t-1}^h, S_t^b)$ gives the expected task state given the current behavior and the previous task state. Notice that the only conditional independencies introduced here are in this last distribution: the task state is independent of the attitude, S^p, and the action, A. The idea is that changes in the *task* states are caused by the *behaviors* of the user, independently of the user's attitude or the system's actions. The action, A, only affects the *behavior* and *attitude* of the user, which in turn may cause changes to the *task*.

The observations $O = \{O^h, O^b\}$ are generated by the *task* and *behavior* variables, S^h and S^b, respectively, through some observation functions $P(O^h|S^h)$ and $P(O^b|S^b)$. These distributions can be of many different types, depending on what the observations are. In general, however, the time scales at which observations occur will be of much shorter duration than those at which task or attitude variables change values. Observations of behaviors will typically be frames from some video camera (at 30Hz), or some segments of an audio stream (at around 10kHz), whereas the task states will only change every few seconds. For example, during handwashing, a typical behavior may be "putting soap on hands", which may take a few seconds to perform, and result in 30 or more video frame observations, but only cause a single change of the *task* state: the hands become "soapy".

An example of a task observation function is the relationship between the task variable *water_on* and the signal from a switch or impeller during the handwashing task [3]. The following section describes a behavior observation model, $P(O^b|S^b)$, which is designed for the task of handwashing, but is fairly general in its applicability.

2.1 Behavior Observation model

The observation function, $P(O^b|S^b)$, encodes the relationship between the unobservable behaviors, S^b, and the observations O^b, which in this case are derived features from sequences of video frames. The parameterisation of this function follows the generative model developed in [10] for constructing temporally and spatially abstract descriptions of video sequences. Figure 3 shows the model as a Bayesian network being used to assess two frames from a sequence in which a patient is putting soap on his hands.

Our raw measurements consist of the video frames, I, and the optical flow fields between consecutive images, v.[2] The observation function must first spatially summarise

[2]Optical flow is computed using the method of Simoncelli [25].

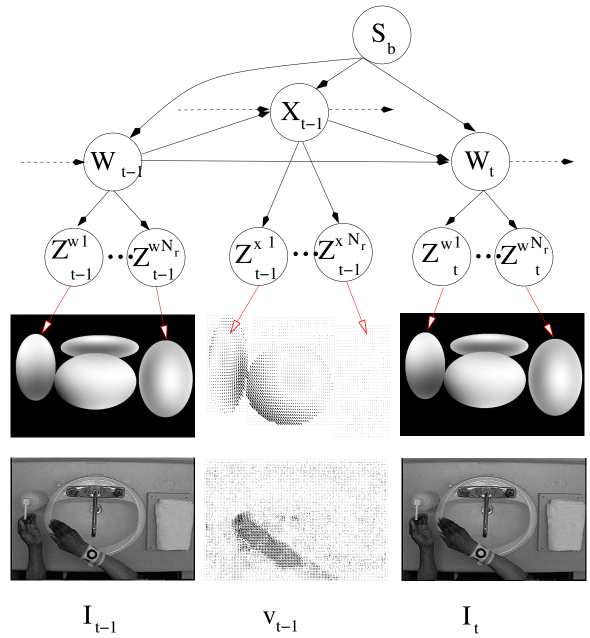


Figure 3: Dynamic Bayesian network for modeling observations of images and flows over regions as a function of S^b, the patient behavior. Along the bottom are the frames and the flow field computed using Simoncelli's multi-scale method [25]. Z^w and Z^x are the feature vector projections over statically defined regions. The reconstructed images and flow fields given Z^w and Z^x are shown below them ($N_z = 12$). W and X are discrete variables describing the instantaneous configuration and dynamics through Gaussian distributions over Z^w and Z^x.

both of these quantities, then temporally compress the entire sequence to a distribution over high level descriptors, S^b. The vertical chains in Figure 3 induce distributions over the multivariate random variables, W and X, from the projections Z^w and Z^x, respectively. W and X correspond to classes of instantaneous (per frame) configuration and dynamics of the region(s) of interest in the training data. For example, the configuration classes may correspond to characteristic hand poses, such as "reaching for the soap" during handwashing. The dynamics classes are motion classes, and may correspond to, for example, motion during the patient reaching for the soap. The mapping from frames and flow fields to W and X occurs as follows.

First, we assume a static set of N_r image regions ($N_r = 4$ in Figure 3). The grayscale image, I, and the optical flow field, v, within each region are projected to a pre-determined set of N_z 2D basis functions, yielding N_z dimensional feature vectors, $Z^{w,r}$ and $Z^{x,r}$, respectively, for each region, $r \in 1 \ldots N_r$. The concatenation of the $2 \times N_r$ feature vectors at each time step then form the observations, $\mathbf{O} = \{Z_1^w, Z_1^x, \ldots, Z_T^w, Z_T^x\}$, where $Z^w =$

$\{Z^{w,1} \ldots Z^{w,N_r}\}$, and $Z^x = \{Z^{x,1} \ldots Z^{x,N_r}\}$. Using a pre-determined basis set defers any commitment to particular types of motion to higher levels of processing, without affecting computational efficiency. We use the basis of Zernike polynomials, which have useful properties for modeling flow fields [10] and images [27]. This basis set is a complete and orthogonal set of polynomials over an elliptical region, such that Z^w and Z^x can be used to reconstruct images and flow fields to an arbitrary degree of accuracy, given sufficient basis projections. The basis functions are ordered by their spatial frequencies, such that low orders represent gross structure in images and flow fields, and higher orders represent more complex structures. The second row of images in Figure 3 shows the reconstructed images and flow fields over the defined regions given the projections Z^w and Z^x, with $N_z = 12$.

The distributions of each of the feature vectors (for configuration, Z^w, and dynamics, Z^x) are modeled by a mixture of Gaussian distributions, where the mixture components are labeled as states of W and X. The mixture models at this stage also include feature weights as priors on the cluster means. These feature weights obviate the choice of which basis functions are most useful for classification. The dynamics and configuration variables, X and W, each form Markovian chains, which are coupled. This coupling makes these chains coupled hidden Markov models, or ***CHMM***s [4]. Temporal abstraction is achieved using a mixture model at the high level, where the mixture components (states of S^b) are coupled hidden Markov models. Thus, each state of the high level display descriptor, S^b, *generates* a coupled hidden Markov model. The CHMM, in turn, generates a sequence of configuration and dynamics feature vectors, Z^x and Z^x, respectively, which can be used to reconstruct the associated images and flow fields.

This mixture model computes the likelihood, $P(\mathbf{O} = \mathbf{o}|S^b = s^b)$, of a sequence, $\mathbf{o} = \{z_1^x \ldots z_T^x, z_1^w \ldots z_T^w\}$, given the behavior, s^b, using $P(\mathbf{o}|s^b) = \sum_{kl} \alpha_{Tkl}$, where $\alpha_T^{kl} = P(X_T = k, W_T = l, z_1^x \ldots z_T^x, z_1^w \ldots z_T^w|s^b)$ is the *forward* variable, defined recursively

$$\alpha_t^{kl} = \phi_t^{x,k}\phi_t^{w,l}\sum_{ij}\theta_t^{x,ijk}\theta_t^{w,jkl}\alpha_{t-1,ij} \tag{1}$$

$$\alpha_1^{kl} = \phi_1^{x,k}\phi_1^{w,l}\pi^{x,kl}\pi^{w,l} \tag{2}$$

where where the parameters of the model are the transition probabilities in the coupled chains,

$$\theta^{x,ijk} = Pr(X_t = k|W_{t-1} = j, X_{t-1} = i, s^b) \tag{3}$$

$$\theta^{w,jkl} = Pr(W_t = l|X_{t-1} = k, W_{t-1} = j, s^b) \tag{4}$$

the initial state probabilities in the coupled chains,

$$\pi^{x,kl} = Pr(X_1 = k|W_1 = l, s^b) \tag{5}$$

$$\pi^{w,l} = Pr(W_1 = l|s^b) \tag{6}$$

and the observation probabilities of the projections of flow field and image regions given dynamics and configuration states, respectively.

$$\phi_t^{x,k} = P(z_t^x|X_t = k) \tag{7}$$

$$\phi_t^{w,l} = P(z_t^w|W_t = l) \tag{8}$$

Finally, we can compute the belief state at time t from the belief state at time $t-1$, $Pr(s_{t-1})$, the caregiver action a_t and the video sequence $\mathbf{o}_t$, using:[3]

$$P(s_t|a_t, \mathbf{o}_t) \propto \sum_{s_{t-1}} P(s_t|a_t, s_{t-1})P(\mathbf{o}_t|s_t^b)P(s_{t-1}) \tag{9}$$

where $P(s_t|a_t, s_{t-1})$ is the POMDP transition function defined in Section 2, and $P(\mathbf{o}_t|s_t^b)$ is the observation likelihood described above.

Since the video sequences are sometimes fairly long (over 1000 frames), the probability mass usually collapses nearly entirely to a single behavior model, assigning exactly zero to the others. This causes problems for the state distribution monitoring since it sees these behaviors as "impossible" given the data. We resolve this issue by dividing the logarithms of the likelihoods by the sequence lengths. This is similar to the heuristics described in [1] for resolving the imbalance between transition and observation probabilities.

2.2 Learning

The parameters of the model are learned from partially labeled training data. Our focus is to discover models of behaviors, and so we will assume that all variables other than the *behavior* are given an assignment in the annotations. Although the *task* variables are typically simple to annotate, the *attitude* variables require expert knowledge or must be learned from data along with the behaviors. In this work, we assume attitude variables are assessed by an expert.

To learn the parameters of the transition function $P(S_{t+1}|S_t, A)$ and the observation function $P(O|S)$, we apply the expectation-maximization (EM) algorithm [2] as outlined below. It is important to stress that the learning takes place over the *entire* model simultaneously: both the output distributions, including the mixtures of coupled HMMs, and the high-level POMDP transition functions are all learned from data during the process. The learning classifies the input video sequences into a spatially and temporally abstract finite set of *behaviors*, and learns the relationship between these high-level behavior descriptors, the state of the patient's attitude and of the task, and the prompt. The POMDP parameters are learned by finding the

[3] Note the abuse of notation by the subscript t used for the temporal index at both levels in the hierarchy. The meaning is clear from the context, however.

model that maximizes the posterior density of all observations and the model. Denote the set of N_o training video sequences $\omega = \{\mathbf{o_1} \ldots \mathbf{o_{N_o}}\}$, the associated sequences of caregiver prompts $a = \{\mathbf{a_1} \ldots \mathbf{a_{N_o}}\}$, and the annotation sequences of attitude and task states $s^p = \{\mathbf{s^p_1} \ldots \mathbf{s^p_{N_o}}\}$ and $s^h = \{\mathbf{s^h_1} \ldots \mathbf{s^h_{N_o}}\}$, respectively. Then, the learning problem is to find the set of parameters θ^* that maximizes $P(o, s^p, s^h, a, \theta)$, subject to constraints on the parameters, which involves integrating over all possible patient behaviors s^b.

The EM algorithm eases this maximization by writing it

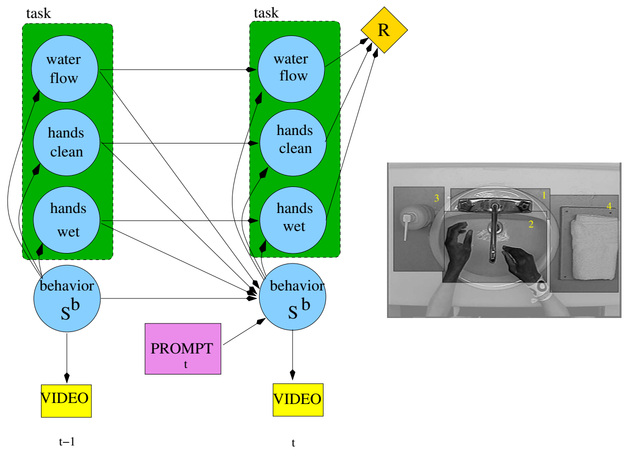
$$\arg\max_{\theta} \left[\sum_{i=1}^{N_o} \sum_{\mathbf{S^b}=\mathbf{s^b}} P(\mathbf{s^b}|\mathbf{o_i}\mathbf{s^p_i}\mathbf{s^h_i}\theta') \log P(\mathbf{o_i}\mathbf{s_i}|\theta) + \log P(\mathbf{\Theta}) \right]$$

The "E" step of the EM algorithm is to compute the expectation over the hidden behaviors, $P(\mathbf{s^b}|\mathbf{o_i}\mathbf{s^p_i}\mathbf{s^h_i}\theta')$, given θ', a current guess of the parameter values. The "M" step is then to perform the maximization which, in this case, can be computed analytically by taking derivatives with respect to each parameter, setting them to zero and solving for the parameters [9, 2].

The updates to the output CHMM distributions are very similar to those for a normal HMM with Gaussian outputs, except that evidence is propagated backwards and forwards through both X and W chains. The output distribution updates include learning the feature weights on the dimensions of the feature vectors Z^x and Z^w. These updates are included by putting priors and hyperpriors on the means and covariances of the output distributions, and then updating the priors based on the usefulness of each feature dimension at separating the classes in X and W. This feature weight learning is based on the method of [6], and is described in more detail in [9].

3 Handwashing Model

We will demonstrate how to apply the model in Section 2 to a particular activity of daily living (ADL), handwashing. In this ADL, the patient needs to get his hands clean by progressing through stages that include using soap, turning the water on and off, rinsing and drying his hands. A caregiver monitors the progress of the patient, issuing reminders or prompts at appropriate times. In this simplified model, shown in Figure 4(a), we only use the task variables, disregarding the patient's attitude and timing.[4] The states of the handwashing task are defined by the variables *hands_clean*, which can be $\{$*dirty, soapy, clean*$\}$, *hands_wet*, which can be $\{$*wet, dry*$\}$, and *water_flow*, which can be $\{$*on, off*$\}$. We assume the hands start *dirty* and *dry*, and the goal is to get them *clean* and *dry*, which can only happen if they become *soapy* and *wet* at some intermediate time. The water starts *off* and must be *off* for task completion. The PROMPTs are the possible reminders that can be given to the patient in the form of audible cues, and one null action where the caregiver waits. The cues correspond to the canonical steps of handwashing: *turn on water*, *wet hands*, *use soap*, *dry hands* and *turn off water*.



(a) (b)

Figure 4: (a) Two time slices of factored POMDP for ADL monitoring. (b) Regions for the handwashing task.

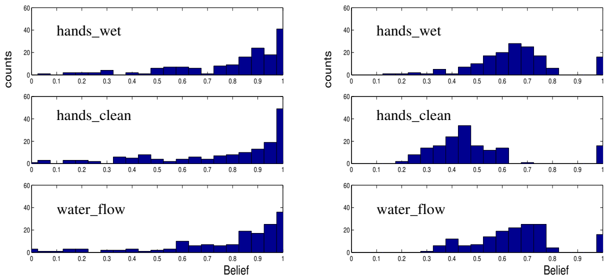
The mixture of CHMMs function outlined in Section 2 is used with four static regions of the image, as shown in Figure 4(b). These regions were manually specified, and correspond roughly to four major regions of interest in the sink area: the taps, the sink, the soap and the towel.

The training data is a subset of that used in [17]. It consists of 23 trials with the same caregiver and the same patient performing the handwashing task on 23 different days. We selected 18 of these sequences in which the patient was sleeveless. The addition of sleeves would mean the number of configuration states would likely have to double. One further sequence was dropped due to excessive caregiver presence in the video at the beginning. The patterned bracelet worn by the patient was used for the pattern recognition algorithm of [17], and is not necessary for our system, as the representation disregards it.

The 17 sequences were annotated by one of the authors. The water flow and the state of the hands was monitored and recorded along with the caregiver's actions. Temporal segmentation is achieved by simply cutting the sequence where the annotations recorded a change of task state. This temporal segmentation remains static throughout. Uninformative prior transition and observation models were used.

We performed a cross-validation experiment in which one sequence (the *test sequence*) is removed from the training set, and the labels on this test sequence are hidden. The

[4] Other aspects of our work investigate these variables [3], but do not do any learning.



(a) (b)

Figure 5: Histograms of marginal belief in actual (annotated) values of the variables *hands_wet* , *hands_clean* and *water_flow*. (a) with video observations (b) with uninformative video observations ($P(S^b|O)$ is uniform).

model is trained on the 17 remaining sequences, and then used to estimate the task state on the *test sequence* (using Equation 9). The estimates can then be compared with the (hidden) annotations.[5] This process is repeated for each possible test sequence. The learned behaviors typically fell into five categories corresponding to the five major things that the patients did during handwashing: nothing (staying still), turning the water on or off (these appear the same), rinsing the hands, drying the hands and using the soap.

In order to gauge how well the correct state is being monitored, we can look at the marginal belief the model has in each of the annotated values of the task variables, *hands_wet*, *hands_clean* and *water_flow*. Figure 5(a) shows histograms of these marginal beliefs over all 18 left-out test sequences. For comparison, Figure 5(b) shows the beliefs if the video observations were uninformative (so the distribution $P(S^b|O)$ is uniform). We see that, on average, the observation function is learning behaviors that allow for more accurate belief monitoring.

We also compared the most likely state of the probability distribution to the (hidden) annotation at each time step. There are 192 sequences in total, each of the handwashing trials having about a dozen changes of state. The fraction of correct maximum likelihood guesses is estimated to be 0.91 for *hands_wet*, 0.87 for *hands_clean* and 0.88 for the *water_flow* (for $N_z = 6$). Note that, even if the POMDP model does not correctly guess the most likely task state, it may still be able to choose optimal actions.

Figures 6–9 show examples of belief propagation on a test sequence. In this sequence, the patient manages to get as far as turning on the water (Figure 6) and wetting their hands before the caregiver has to step in and guide him for the rest of the task.

[5] However, note that one of the strengths of these kinds of models is their ability to maintain a belief distribution, and this comparison of the belief distribution with the annotations is not a direct measurement of the model's ability to monitor belief and ultimately choose correct actions.

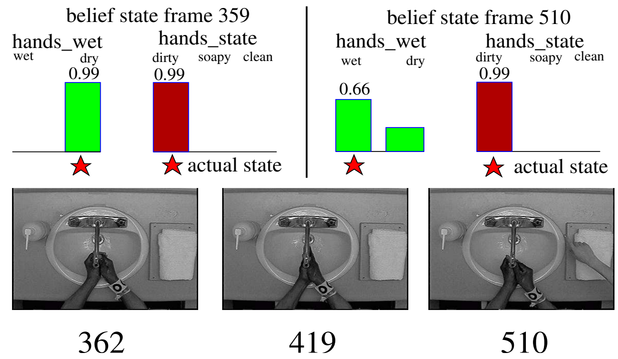


Figure 6: Belief propagation over a sequence where the patient wets their hands. No prompt was given to initiate this behavior. The state distribution is shown along the top. The patients hands are initially believed to be *dry* and *dirty* at frame 359, but the observed behavior shifts the belief over *hands_wet* towards *wet*. The actual task state is indicated by a star.

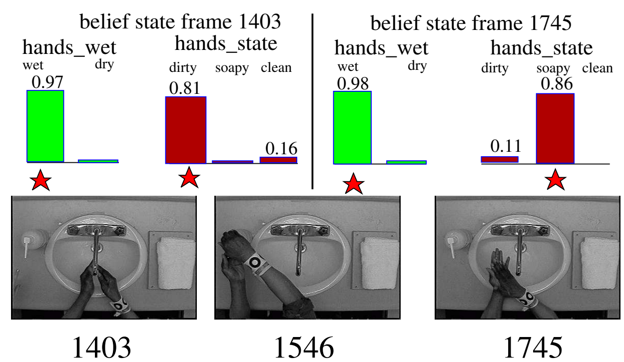


Figure 7: Belief propagation over a sequence where the patient uses soap. The caregiver prompted "I want you to use some soap now" at frames 1340-1390.

We start at frame 362 in Figure 6, in what is essentially the initial state distribution, which is that the patient's hands are dry and dirty. The patient then rubs his hands under the water until about frame 510 when the caregiver prompts him to turn on some cold water. We compute the probability of this sequence given each possible behavior model, and use these estimates to update the state distribution at frame 510, in which the hands are believed to be wet (with probability 0.66), and still dirty. Notice how the caregiver's hand enters the picture towards the end of this sequence. The model is able to deal with this confusing factor.

The caregiver prompts just prior to frame 1400, which leads the patient to put some soap on his hands, which he does until about frame 1745, shown in Figure 7. The state distribution changes accordingly, giving high weight to *hands_clean=soapy*.

Figure 8 gives a further example during the same test

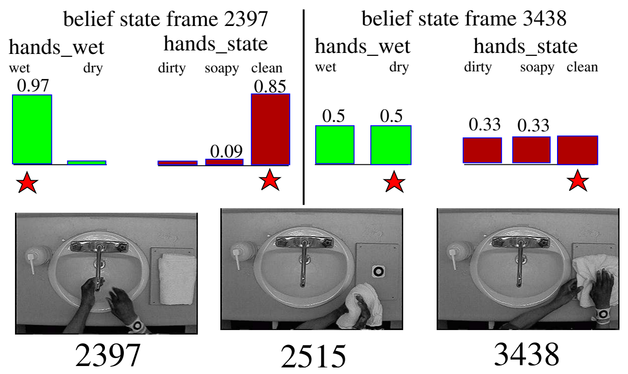


Figure 8: Belief propagation over a sequence where the patient dries their hands. The caregiver prompted "dry your hands now" at frames 2360-2390.

trial, in which the patient is prompted to dry their hands (Figure 8). This behavior shifts the state distribution away from *hands_wet*=*wet* and *hands_clean*=*clean*, but the resulting distribution is even across states. This happens due to lack of training data to support this kind of event[6], and could be resolved by using more informative priors. This is not necessarily a problem for the POMDP, since this distribution could still be used to appropriately select actions.

Figure 9 shows an example of a sequence in which the state distribution is not updated in correspondence with the annotations. The behavior appears as a rinsing behavior, and so the belief is updated so that the hands are most probably clean, not soapy. However, the hands are not under the water, and the hands do not actually get rinsed. This type of error could be corrected with some 3D information in the data, such as from a stereo camera.

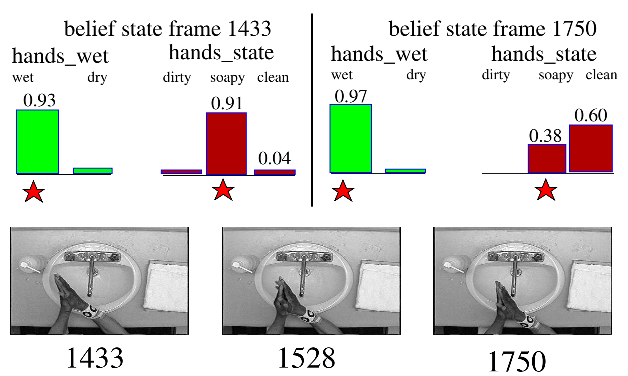


Figure 9: Belief propagation over a sequence where soap is used. The belief shifts from hands being *soapy* to *clean*.

[6]There is only 1 training sequence in which the patient dries his hands *before* turning off the water, and in that case, he was not prompted to do so as he was in this test sequence.

4 Related Work

Assistive technologies have primarily been investigated outside the artificial intelligence community, and are reviewed in LoPresti *et al.*[15]. Most relevant to our work, a system for monitoring handwashing using a ceiling-mounted camera demonstrated a significant reduction in caregiver burden [17]. The patient was required to wear a patterned bracelet, the location of which was determined by a pattern recognition algorithm. The resulting location was then input to a neural network for the recognition of predefined behaviors. This system was invasive and was not learned from data.

Several intelligent systems that use AI and ubiquitous computing techniques are currently being developed for the older adult. These include the Aware Home Project [18], the Assisted Cognition Project [13] and the Nursebot Project [21]. These new projects are similar to the work described in this paper in the sense that they attempt to incorporate AI and a decision-theoretic approach to overcome the shortcomings of current approaches. In particular, the Autominder System [22], one aspect of the Nursebot Project, applies a POMDP in the development of the planning and scheduling aspect of the system [21]. These systems do not incorporate advanced prompting techniques and algorithms, but rather are being developed as scheduling and memory aids. POMDPs have also been used for modeling interactions between humans in an office environment, e.g. during meetings and cooperative tasks [28].

These computerised assistive technologies, however, do not investigate the learning problem in detail. However, in related work, there has been significant progress in learning patterns of activity from a person's positional data. For example, data mining techniques have been used to discover sequences of activities from discrete data [7], and hierarchical hidden Markov models (HHMMs) have been used to explain GPS data of outdoor transportation patterns [14]. We are learning a model similar to the HHMM, but explicitly add system actions and model video sequences directly instead of only positional data. Other researchers use supervised techniques to build models of meeting dynamics [24], office activity [19], and other in-home activities [8].

Our previous work showed how to learn the parameters of a partially observable Markov decision process, while *discovering* models of facial displays and hand gestures in game-playing domains[10]. This work differed from previous work in human motion analysis in that it does not only attempt to *recognize* either purported characteristic behaviors, or the pre-defined atomic units which make up such behaviors [26, 5]. Instead, we make no prior assumptions about the number of type of behaviors that are present, learning this directly from data. However, the resulting model was used as a Markov decision process (MDP) by taking the most likely inferred behavior as the true state. This paper generalises that work by relaxing this observable constraint in the test data, and begins to look at the problem of learning and using the partially observable model. This paper also applies the POMDP model to the domain of ADL monitoring for the elderly.

5 Conclusion and Future Work

We have shown how to learn the parameters of a two-level model of patient–caregiver interactions during an activity of daily living. We described a simple observation function that recognises patient behaviors, and showed how it can be used to monitor the high-level state of the task. The primary advantage of the kinds of models we describe is that they do not require training data labeled with patient behaviors.

Our future work in this domain will progress along three fronts. First, from a model learning perspective, we want to investigate more complex POMDP models, with the addition of the patient's mental state, timing, and models for other ADLs. Eliciting rewards and cost functions for the POMDP is a critical part of this task, as is the ability for the system to learn *in situ*, for which we are looking into Bayesian reinforcement learning techniques. Second, from a computer vision perspective, we want to include more sophisticated models for spatial segmentation and tracking. Third, from a decision theoretic perspective, we want to investigate the solution of the POMDP models to yield policies of action. The POMDP models we are learning are intractable, but recent advances in approximate solution techniques promise to yield efficient and accurate policies [11, 23, 20]. These solution methods will enable value-directed learning of behavior categories, allowing us to learn models with only those behaviors that are *useful* to the task.

The type of learning we have presented in this paper will automatically define *behaviors* based on what goals the system has, and the attitude of the user. For example, a particular handwashing user may only need help remembering to turn the water off. For this person, only the composite behavior of "washing hands" (including soaping, rinsing and drying), and the primitive behavior of "turning the water off" need to be recognised. An interesting subject of future research is this tradeoff is between the complexity of the observation function $P(O|S)$ and the complexity of the task state, S^h. The more complex we make $P(O|S)$, the less complex the task state needs to be, but the less control the system will be able to exercise. Investigation of this tradeoff is an interesting avenue for future research.

References

[1] Y. Bengio. Markovian models for sequential data. *Neural Computing Surveys*, 2:129–162, 1999.

[2] Y. Bengio and P. Frasconi. Input-output HMMs for sequence processing. *IEEE Transactions on Neural Networks*, 7(5):1231–1249, September 1996.

[3] J. Boger, P. Poupart, J. Hoey, C. Boutilier, G. Fernie, and A. Mihailidis. A decision-theoretic approach to task assistance for persons with dementia. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Edinburgh, Scotland, July 2005. to appear.

[4] M. Brand, N. Oliver, and A. Pentland. Coupled hidden Markov models for complex action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 994–999, Puerto Rico, 1997.

[5] C. Bregler. Learning and recognising human dynamics in video sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 568–574, Puerto Rico, 1997.

[6] P. Carbonetto, N. de Freitas, P. Gustafson, and N. Thompson. Bayesian feature weighting for unsupervised learning with application to object recognition. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, pages 122–128, Key West, Florida, January 2003.

[7] V. Guralnik and K. Z. Haigh. Learning models of human behaviour with sequential patterns. In *Proceedings of the AAAI-02 Workshop on Automation as Caregiver*, pages 24–30, July 2002.

[8] R. Hamid, Y. Huang, and I. Essa. Argmode - activity recognition using graphical models. In *Proceedings of the CVPR Workshop on Detection and Recognition of Events in Video (EVENT)*, pages 38–44, Madison, WI, June 2003.

[9] J. Hoey and J. J. Little. Decision theoretic modeling of human facial displays. Technical Report TR-04-02, Department of Computer Science, 2004.

[10] J. Hoey and J. J. Little. Value-directed learning of gestures and facial displays. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1026–1033, Washington, DC, 2004.

[11] J. Hoey and P. Poupart. Solving POMDPs with continuous or large discrete observation spaces. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Edinburgh, July 2005. to appear.

[12] J. Hoey, P. Poupart, C. Boutilier, and A. Mihailidis. POMDP models for assistive technology. IATSL Technical Report, 2005.

[13] H. Kautz, L. Arnstein, G. Borriello, O. Etzioni, , and D. Fox. An overview of the assisted cognition project. In *Proceedings of the AAAI-2002 Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care*, pages 60–65, July 2002.

[14] L. Liao, D. Fox, , and H. Kautz. Learning and inferring transportation routines. In *Proc Nineteenth National Conference on Artificial Intelligenc (AAAI)*, pages 348–353, San Jose, CA, 2004.

[15] E. F. LoPresti, A. Mihailidis, and N. Kirsch. Assistive technology for cognitive rehabilitation: State of the art. *Neuropsychological Rehabilitation*, 14(1/2):5–39, 2004.

[16] W. S. Lovejoy. A survey of algorithmic methods for partially observed Markov decision processes. *Annals of Operations Research*, 28:47–66, 1991.

[17] A. Mihailidis. *The Development of an Intelligent Cognitive Orthosis to Facilitate Handwashing for Persons with Moderate- to- Severe Dementia*. PhD thesis, University of Strathclyde, 2002.

[18] E. Mynatt, I. Essa, and W. Rogers. Increasing the opportunities for aging in place. In *ACM Conference on Universal Usability (CUU)*, pages 65–71, Arlington, Virginia, 2000.

[19] N. T. Nguyen, H. H. Bui, S. Venkatesh, and G. West. Recognising and monitoring high-level behaviours in complex spatial environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 620–625, Madison, WI, June 2003.

[20] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1025–1032, Acapulco, Mexico, August 2003.

[21] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun. Towards robotic assistants in nursing homes: Challenges and results. *Robotics and Autonomous Systems*, 42(3–4), 2003.

[22] M. E. Pollack, L. Brown, D. Colbry, C. E. McCarthy, C. Orosz, B. Peintner, S. Ramakrishnan, and I. Tsamardinos. Autominder: An intelligent cognitive orthotic system for people with memory impairment. *Robotics and Autonomous Systems*, 44:273–282, 2003.

[23] P. Poupart and C. Boutilier. VDCBPI: an approximate scalable algorithm for large scale pomdps. In *Advances in Neural Information Processing Systems 17 (NIPS)*, pages 1081–1088, Vancouver, BC, 2004.

[24] P. E. Rybski and M. M. Veloso. Using sparse visual data to model human activities in meetings. In *Proc. Modeling Other Agents from Observations (MOO 2004)*, pages 9–16, New York, NY, July 2004.

[25] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger. Probability distributions of optical flow. In *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 310–315, Mauii, Hawaii, USA, 1991.

[26] T. Starner and A. P. Pentland. Visual recognition of American Sign Language using hidden Markov models. In *International Workshop on Automatic Face and Gesture Recognition*, pages 189–194, Zurich, Switzerland, 1995.

[27] C.-H. Teh and R. T. Chin. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):496–513, July 1988.

[28] P. Varakantham, R. Maheswaran, and M. Tambe. Exploiting belief bounds: Practical POMDPs for personal assistant agents. In *Proceedings of the International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS)*, Utrecht, Netherlands, July 2005. to appear.