

Recognizing Activities in Multiple Contexts using Transfer Learning

T.L.M. van Kasteren and **G. Englebienne** and **B.J.A. Kröse**

Intelligent Systems Lab Amsterdam (ISLA)
Science Park, Kruislaan 403, 1098 SJ
Amsterdam, The Netherlands

Abstract

Activities of daily living are good indicators of the health status of elderly. Therefore, automating the monitoring of these activities is a crucial step in future care giving. However, many models for activity recognition rely on labeled examples of activities for learning the model parameters. Due to the high variability of different contexts, parameters learned for one context can not automatically be used in another. In this paper, we present a method that allows us to transfer knowledge of activity recognition from one context to the next, a task called transfer learning. We show the effectiveness of our method using real world datasets.

Introduction

As the number of elderly people in our society increases so do the costs associated with hospitalization and nursing homes. We can reduce these costs by monitoring the health status of the elderly at home, and thus enabling them to live longer on their own. A good indicator of their health status is the ability to perform activities of daily living (ADLs), such as bathing, toileting and cooking (Katz et al. 1970). Therefore, automating the monitoring of these activities is a crucial step in future care giving.

In order to monitor activities a variety of sensors is available, particularly popular have been wireless sensor networks. Developments in sensing and network technology allow us to use wireless binary sensing nodes that are easy to install and run on batteries for several months. These nodes can be equipped with different kinds of sensors, to measure a door being opened, a toilet being flushed or the temperature above a stove rising. However, inferring the performed activity from this data is challenging, since the output of these sensors is noisy and ambiguous. The usual technique to overcome this ambiguity involves machine learning algorithms, which can model and recognize the sensor patterns generated from the activities in question.

Although the performance of activity recognition models is getting better and better, often models rely on labeled examples of activities for learning the model parameters (Tapia, Intille, and Larson 2004; van Kasteren and Kröse 2007). Due to the high variability in the layout of houses

and the way different individuals perform an activity, parameters learned for one house can not automatically be used in another. Ideally, we would have separate training data for each individual and house, however, because we are working with elderly we cannot expect each one of them to annotate their own personal dataset. Therefore, we would like a system which is able to use existing datasets from other houses in order to learn the parameters of the model for a new house. Using the knowledge gained in one problem and applying it to a different but related problem is called transfer learning (Mihalkova, Huynh, and Mooney 2007; Raina, Ng, and Koller 2006).

In this work, we present a method which uses unlabeled data from house A together with labeled data from house B, to learn the parameters of a model for activity recognition in house A. This involves two steps, first we need to find a way to map the two sensor networks to each other. Because the houses differ, the location of the sensors and the properties they measure will also differ. To be able to use the data from both houses we need to associate a sensor from house A to a sensor in house B. Second, we need to learn the parameters using the labeled and unlabeled data. We therefore use a semi-supervised learning algorithm to find the parameters of our model.

The remainder of this paper is organized as follows. In the next section we discuss related work. After that, we describe the various datasets that we used in this work. We then give a description of our model for activity recognition and describe our method for transfer learning. Then we discuss the experiments and results. Finally, we conclude by summing up our findings.

Related work

Activity recognition has been performed using static sensors (e.g. reed switches (Tapia, Intille, and Larson 2004), motion detectors (Logan et al. 2007), cameras (Duong et al. 2005)) and wearables (e.g. accelerometers (Lester et al. 2005), wrist worn RFID reader (Patterson et al. 2005)). The technologies differ from each other in terms of price, intrusiveness, ease to install and the type of data they output (Schmidt 2002). Models used for recognizing activities can be probabilistic (Duong et al. 2005; Patterson et al. 2005), logic based (Landwehr et al. 2007) or hand-crafted (Fogarty, Au, and Hudson 2006). Probabilistic

bilistic models are popular because sensor readings are noisy and activities are typically performed in a non-deterministic fashion. Hidden Markov models were used to perform activity recognition at an object usage level using a wrist worn RFID reader (Patterson et al. 2005). Hierarchical hidden Markov models were used to perform activity recognition from video (Duong et al. 2005). And a comparison between hidden Markov models and conditional random fields was made to show the effectiveness of generative and discriminative in activity recognition (van Kasteren et al. 2008).

All these models for activity recognition need labeled data to learn their parameters. In work by Wilson a method is proposed to make the collection of labeled data easier. Inhabitants of a house are shown a game-like environment on a display, in which past sensor readings are shown using easily understandable icons. By letting the user manually annotate each sequence of sensor readings, a personal dataset is created which can be used for training (Wilson, Long, and Atkeson 2005).

Previous work on transfer learning has been applied to various domains. Transfer learning in general refers to the problem of retaining and applying knowledge from one task to a new related task. This knowledge transfer either consists of the structure of a particular model being transferred, the parameters of the model being transferred or both. For example, in work by Mihalkova the structure in relationships of individuals in an academic department are mapped to the relationships in an international movie database (e.g. a professor is mapped to a director) (Mihalkova, Huynh, and Mooney 2007). While in work by Raina the prior over parameters in a text classification task was transferred by using the co-occurrence of words in a document (Raina, Ng, and Koller 2006).

Datasets

The sensor network we used was chosen according to two main criteria: ease of installation and minimal intrusion. It consists of wireless network nodes to which simple off-the-shelf magnetic reed switches were attached. The wireless network node has an analog and digital input. It sends an event when the state of the digital input changes or when some threshold of the analog input is violated. Special low-energy consuming radio technology, together with an energy saving sleeping mode result in a long battery life. The node can reach a data transmission rate of 4.8 kb/s, which is enough for the binary sensor data that we need to collect.

Two datasets were recorded using this sensor network. Activities were chosen beforehand based on the Katz ADL index, a commonly used tool in healthcare to assess cognitive and physical capabilities of an elderly person (Katz et al. 1970). The activities were annotated by the subjects themselves while performing the activities, in one case a bluetooth headset was used, in the other an activity diary.

House 1

Using our sensor network, we recorded a dataset in the house of a 26-year-old man. He lives alone in a three-room apartment where 14 state-change sensors were installed. The sensors were placed, amongst others, on doors, cupboards, the

refrigerator and in the toilet cistern (see fig. 1). Sensors were left unattended, collecting data for 28 days in the apartment. This resulted in 2120 sensor events.

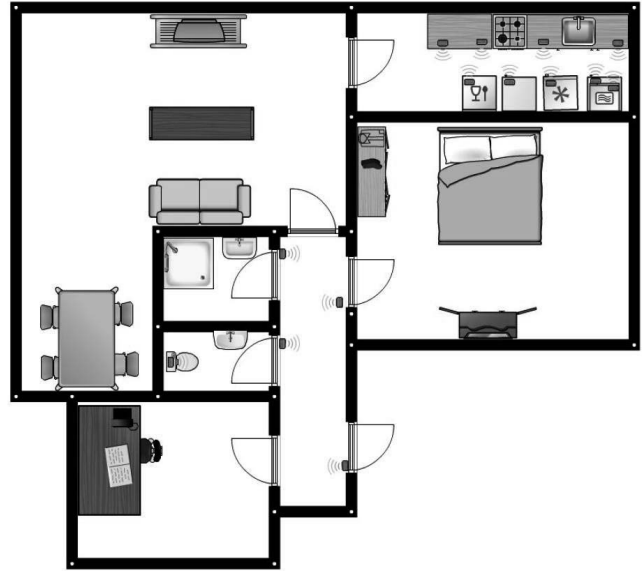


Figure 1: Floorplan of house 1, red rectangle boxes indicate sensor nodes.

Annotation was done by the subject himself at the same time the sensor data was recorded. A bluetooth headset combined with speech recognition software was used for annotation (van Kasteren et al. 2008). Seven different activities were annotated, namely: 'Out of house', 'Toileting', 'Showering', 'Sleeping', 'Preparing breakfast', 'Preparing dinner' and 'Preparing a beverage'. The timeslices at which nothing is annotated is collectively grouped in an extra activity called 'other activities'. Table 1 shows the number of separate instances of activities and the percentage of time each activity takes up in the data set. This table clearly shows how some activities occur very frequently (e.g. toileting), while others that occur less frequently typically have a longer duration and therefore take up more time (e.g. leaving and sleeping). A total of 245 activity instances were annotated by the subject.

	Number of instances	Percentage of time
Other acts.	-	11.5%
Out of house	34	56.4%
Toileting	114	1.0%
Showering	23	0.7%
Sleeping	24	29.0%
Breakfast	20	0.3%
Dinner	10	0.9%
Drink	20	0.2%

Table 1: Number of instances and percentage of time activities occur in the house 1 dataset.

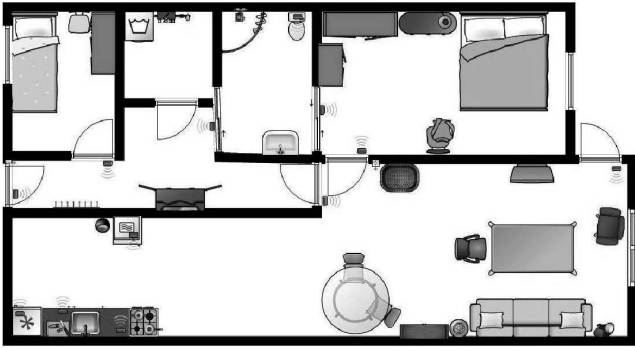


Figure 2: Floorplan of house 2, red rectangle boxes indicate sensor nodes.

House 2

A second dataset was recorded in the house of a 72-year-old woman. She lives alone in an apartment where 13 state-change sensors were installed. Locations of sensors include doors, cupboards, refrigerator and a toilet flush sensor (see fig. 2). Sensors were left unattended, collecting data for 6 days in the apartment. This resulted in 1318 sensor events.

Activities were annotated by the subject herself maintaining an activity diary. The start and end time of each activity was noted on a piece of paper at a resolution of 5 minutes. The same activities as in house 1 were annotated, table 2 shows the list of activities together with the number of separate instances and the percentage of time. A total of 124 activity instances were annotated by the subject.

	Number of instances	Percentage of time
Other acts.	-	35.6%
Out of house	55	28.8%
Toileting	39	2.0%
Showering	4	0.5%
Sleeping	5	31.2%
Breakfast	7	0.9%
Dinner	8	0.6%
Drink	6	0.4%

Table 2: Number of instances and percentage of time activities occur in the house 2 dataset.

Dataset comparison

Comparing the two datasets, we see the percentage of 'out of house' time in house 1 is higher than in house 2. In house 2 we see that the 'other activities' time is higher, while 'out of house' time is lower than in house 1. In terms of layout, house 1 only has a single door through which the house can be exited. House 2 has both a front and back door both of which the subject used to leave and enter the house. House 1 has the bathroom and toilet in a separate rooms, while house 2 has them in one room which can be accessed through two doors. The bedroom in house 1 only has one entry point,

while in house 2 there are two doors leading to the bedroom. A complete list of the sensors together with the function group they belong to is given in table 3. The function group shows a grouping of similar sensors, this information is later used by the transfer learning algorithm.

Because of the differences in layout of the houses and behavior of the inhabitants, it is not so straightforward to combine the two datasets to get improved results. However, our method for transfer learning allows us to use the labeled information from house 1 to improve the results on the unlabeled data of house 2.

Function Group	House 1	House 2
House Entrance	Front door	Front door Back door
Bedroom Entrance	Hall-Bed door	Live-Bed door Bath-Bed door
Bathroom Entrance	Hall-Bath door	Hall-Bath door Bath-Bed door
Toilet Entrance	Hall-Toilet door	Hall-Bath door Bath-Bed door
Toilet Flush	Toilet flush	Toilet flush
Kitchen Cooking	Microwave	Microwave
Kitchen Food	Fridge	Fridge
	Freezer	
	Cupboard #1	
Kitchen Drinks	Cupboard #2	Cupboard #1
Kitchen Plates	Cupboard #3	
Kitchen Utensils	Cupboard #4	
Kitchen Cleaning		Cupboard#2
Clean Clothes	Washmachine	
Clean Dish	Dishwasher	
Various		Study door
		Live-Hall door

Table 3: List of sensors from house 1 and house 2, ordered by their function. The 'Room-Room door' sensors refer to sensors on doors connecting the two rooms. The term 'Live' refers to the living room. Some sensors are listed more than once because they belong to multiple function groups.

Model for Activity Recognition

Before applying transfer learning we first discuss the model used for activity recognition, the effectiveness of this model was shown in (van Kasteren et al. 2008). In this model, the time series data obtained from the sensors is first divided in time slices of constant length Δt (fig. 3). We denote a sensor reading for time t as x_t^i , indicating whether sensor i fired at least once between time t and time $t + \Delta t$, with $x_t^i \in \{0, 1\}$. In the experiment section we introduce some other ways to represent sensor readings. In a house with N sensors installed, we define a binary observation vector $\vec{x}_t = (x_t^1, x_t^2, \dots, x_t^N)^T$. Using K activities, y_t denotes the activity at timeslice t , with $y_t \in \{1, \dots, K\}$. Our model finds a sequence of labels $\mathbf{y} = \{y_1, y_2, \dots, y_T\}$ that best explains the sequence of observations $\mathbf{x} = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_T\}$ for a total of T time steps.

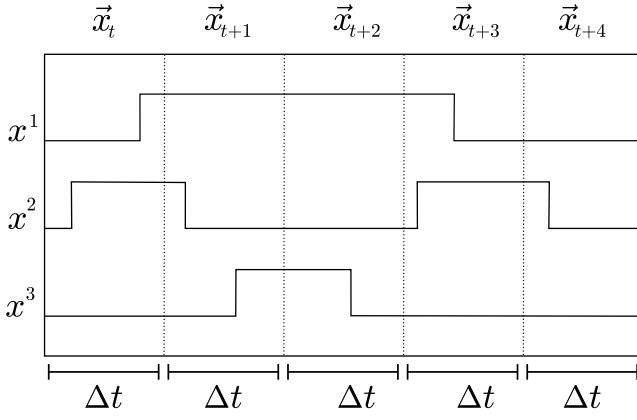


Figure 3: Showing the relation between sensor readings x^i and time intervals Δt .

Because of the noisy and ambiguous nature of the sensor readings, temporal probabilistic models are ideal for this task. We use a generative, parametric probabilistic model known as the Hidden Markov Model (HMM). The generative nature of this model allows us to incorporate unlabeled data, while the use of parameters allows for a compact representation.

Hidden Markov Model

The Hidden Markov Model (HMM) is a generative probabilistic model for sequential data, consisting of a hidden variable and an observable variable at each time step (fig. 4). In our case the hidden variable is the activity performed, and the observable variable is the vector of sensor readings. There are two independence assumptions that define this model, represented by the directed arrows in the figure.

- The hidden variable at time t , namely y_t , depends only on the previous hidden variable y_{t-1} (*Markov assumption* (Rabiner 1989)).
- The observable variable at time t , namely x_t , depends only on the hidden variable y_t at that time slice.

With these assumptions we can specify an HMM using three probability distributions: the distribution over initial states $\pi = \{\pi_k\}$, with $\pi_k = p(y_1 = k)$; the

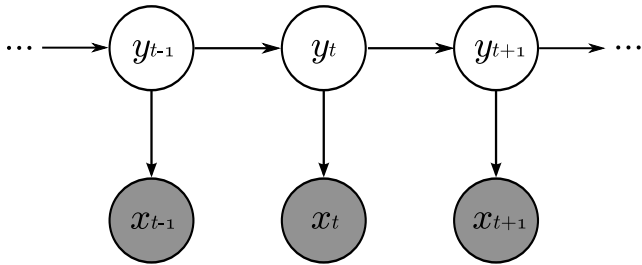


Figure 4: The graphical representation of a HMM. The shaded nodes represent observable variables, while the white nodes represent hidden ones.

state transition probability distribution $A = \{a_{kl}\}$, with $a_{kl} = p(y_t = l | y_{t-1} = k)$ representing the probability of going from state k to state l ; and the observation distribution $B = \{b_{il}\}$, with $b_{il} = p(x_t^i = 1 | y_t = l)$ indicating the probability that the state l would generate observation $x_t^i = 1$. Observations are binary and modeled as independent Bernoulli distributions.

Learning the parameters (π, A, B) of these distributions corresponds to maximizing the joint probability $p(\mathbf{x}, \mathbf{y})$ of the paired observation and label sequences in the training data. We can factorize the joint distribution in terms of the three distributions described above as follows (Bilmes 2006):

$$p(\mathbf{x}, \mathbf{y}) = \prod_{t=1}^T p(y_t | y_{t-1}) p(\vec{x}_t | y_t) \quad (1)$$

in which we write the distribution over initial states $p(y_1)$ as $p(y_1 | y_0)$, to simplify notation. We use the Expectation Maximization (EM) algorithm to estimate the maximum-likelihood parameters (Bishop 2006). The details of the EM algorithm and how we use it to perform transfer learning are discussed in the next section.

Transfer Learning

Our objective is to learn the parameters of the HMM, to perform activity recognition in a house A. However, in learning these parameters we only use unlabeled data from house A and labeled data from an independent house B. For house B we have a sequence of ground truth labels \mathbf{L}_B of length T_B and sensor data \mathbf{x}_B consisting of T_B binary observation vectors each of length N_B . For house A we have no labels, but have T_A binary observation vectors \mathbf{x}_A of N_A dimensions. Note the length of the two data sequences is likely to be different ($T_A \neq T_B$) and so is the number of sensors ($N_A \neq N_B$).

To reach our objective we need to do two things. First, we need to map the two sensor data matrices \mathbf{x}_A and \mathbf{x}_B . Because the layout of the houses differ it is possible that sensor x_A^i measures a different property than sensor x_B^i . Using the data as it is, would associate sensor x_A^i and x_B^i to a single parameter, even though they measure different properties. Therefore, we need to find a mapping $F(\mathbf{x}_A)$ and $F(\mathbf{x}_B)$ resulting in \mathbf{x}'_A and \mathbf{x}'_B respectively, so that x'^i_A and x'^i_B can be consistently associated to a single parameter. Although a perfect mapping is most likely impossible, since two houses differ, we can experimentally determine which mapping works best.

Second, we need to learn the parameters using the labeled and unlabeled data. Therefore, after applying the mapping we combine the two datasets into a single dataset of labeled and unlabeled data. A semi-supervised learning algorithm is then applied to find the maximum-likelihood parameters for the HMM.

Mapping the sensor data

We propose a number of manual mapping strategies $F_{mapping}$ based on the function groups shown in table 3.

Intersect: For each function group, similar sensors are matched and sensors that have no comparable sensor in the other house are disregarded. For example, of the 'house entrance' group only the front door sensors are mapped and the back door sensor in house 2 is disregarded.

Duplicate: For each function group, similar sensors are matched and sensors that have no comparable sensor in the other house use any other sensor that exists in the same function group. For example, the back door sensor in house 2 is matched with the front door sensor of house 1.

Union: For each function group, the union of all the sensors in the group is taken resulting in one sensor output per group per house. For example, the front and back door in house 2 are combined into a single sensor and matched with the front door sensor in house 1.

Semi-supervised learning of the parameters

Using one of the mappings described above, we get $\mathbf{x} = \{F_{mapping}(\mathbf{x}_A), F_{mapping}(\mathbf{x}_B)\}$ and $\mathbf{L} = \{\mathbf{L}_B\}$. We can now use the Expectation Maximization (EM) algorithm to find the maximum-likelihood parameters ($\theta = \{\pi, A, B\}$) for the HMM. This is an iterative process in which the current parameter values θ^{old} are used to find the expectation $Q(\theta, \theta^{old})$ (E-step) and the new parameter values θ^{new} are determined by maximizing the expectation $\theta^{new} = \arg \max_{\theta} Q(\theta, \theta^{old})$ (M-step).

This is a semi-supervised learning approach, in which we combine labeled and unlabeled training data, therefore our E-step is given by:

$$Q(\theta, \theta^{old}) = \sum_{\mathbf{y}} p(\mathbf{y} | \mathbf{x}, \mathbf{L}, \theta^{old}) \log p(\mathbf{x}, \mathbf{y} | \mathbf{L}, \theta) \quad (2)$$

where \mathbf{x} is the sequence of observed data, \mathbf{y} the sequence of hidden variables and \mathbf{L} the sequence of labeled ground truth. The posterior distribution is calculated using Bayes' rule:

$$p(\mathbf{y} | \mathbf{x}, \mathbf{L}, \theta) = \frac{p(\mathbf{x} | \mathbf{y}, \mathbf{L}, \theta)p(\mathbf{y} | \mathbf{L}, \theta)}{\sum_{\mathbf{y}} p(\mathbf{x} | \mathbf{y}, \mathbf{L}, \theta)p(\mathbf{y} | \mathbf{L}, \theta)} \quad (3)$$

where $p(\mathbf{x} | \mathbf{y}, \mathbf{L}, \theta)$ is the conditional probability of the data and $p(\mathbf{y} | \mathbf{L}, \theta)$ can be calculated by applying Bayes' rule once more:

$$p(\mathbf{y} | \mathbf{L}, \theta) = \frac{p(\mathbf{L} | \mathbf{y}, \theta)p(\mathbf{y} | \theta)}{\sum_{\mathbf{y}} p(\mathbf{L} | \mathbf{y}, \theta)p(\mathbf{y} | \theta)} \quad (4)$$

where $p(\mathbf{y} | \theta)$ is the probability given by the state transitions and $p(\mathbf{L} | \mathbf{y}, \theta)$ is given by:

$$p(\mathbf{L} | \mathbf{y}, \theta) = \prod_{t=1}^T p(L_t | y_t, \theta) \quad (5)$$

$$p(L_t | y_t, \theta) = \begin{cases} 0 & \text{if label exists and } L \neq y \\ 1 & \text{if label exists and } L = y \\ \frac{1}{|y|} & \text{if label doesn't exist} \end{cases} \quad (6)$$

Propagating the values of (6) into (5-3), we obtain that $p(y|x, L, \theta) = 1$ if $L = y$; 0 if $L \neq y$ and $p(y | x, \theta)$ if L is absent.

In the M-step we reestimate the parameters of the HMM, the parameters that maximize the expectation $Q(\theta, \theta^{old})$ are given by (Bilmes 1997):

$$\pi_k = p(y_1 = k | \mathbf{x}, \mathbf{L}, \theta^{old}) \quad (7)$$

$$a_{kl} = \frac{\sum_{t=2}^T p(y_t = l, y_{t-1} = k | \mathbf{x}, \mathbf{L}, \theta^{old})}{\sum_{t=2}^T p(y_{t-1} = k | \mathbf{x}, \mathbf{L}, \theta^{old})} \quad (8)$$

$$b_{il}(o) = \frac{\sum_{t=1}^T p(y_t = l, | x_t^i = o, \mathbf{L}, \theta^{old})}{\sum_{t=1}^T p(y_t = l, | \mathbf{L}, \theta^{old})} \quad (9)$$

Experiments

In this section we present the experimental results acquired in this work. We first describe our experimental setup, a description of the experiments and a presentation of the acquired results. Then we conclude the section with a discussion of the acquired results.

Setup

In our experiments the sensor readings are divided in data segments of length $\Delta t = 300$ seconds. Previous experiments have shown that smaller segments result in the same mean accuracy, but with smaller variance (van Kasteren and Kröse 2007). Due to the resolution at which the house 2 dataset was annotated, we did not use segments of smaller length.

The *raw* sensor representation gives a 1 when the sensor is firing and a 0 otherwise (fig. 5a). However, in this work we use the *change point* representation, which gives a 1 to timeslices where the sensor reading changes (fig. 5b) and the *last* sensor fired representation, gives a 1 for the sensor that changed state last and 0 for all other sensors (fig. 5c). The change and last sensor representation give the best results in activity recognition (van Kasteren et al. 2008).

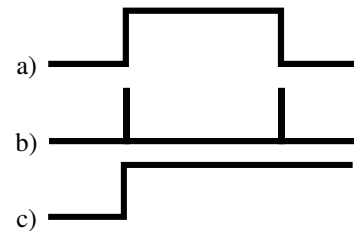


Figure 5: Example of sensor firing showing the a) *raw*, b) *change point* and c) *last* observation representation.

We evaluate the performance of our models by two measures, the time slice accuracy and the class accuracy. These measures are defined as follows:

$$\text{Time slice: } \frac{\sum_{n=1}^N [inferred(n)=true(n)]}{N}$$

$$\text{Class: } \frac{1}{C} \sum_{c=1}^C \left\{ \frac{\sum_{n=1}^{N_c} [inferred_c(n)=true_c(n)]}{N_c} \right\}$$

in which $[a = b]$ is a binary indicator giving 1 when true and 0 when false. N is the total number of time slices and C is the number of classes.

Measuring the time-slice accuracy is a typical way of evaluating time-series analysis. However, we also report the class average accuracy, which is a common technique in datasets with a dominant class. In these cases classifying all the test data as the dominant class yields good time-slice accuracy, but no useful output. The class average though would remain low, and therefore be representative of the actual model performance.

Experiment 1: Mapping comparison

In this experiment we wish to determine which mapping technique gives the best results in transfer learning. We use the labeled training data from house 1 combined with the unlabeled training data from house 2 for training. Discretizing the data resulted in 8002 labeled timeslices from house 1 and 1628 unlabeled timeslices from house 2. All three mapping techniques are tested and EM is performed using randomly initialized parameters. The parameters obtained through transfer learning are used by the HMM to perform inference on the house 2 dataset. The hidden Markov model used consists of a single hidden node taking one of eight possible state values (one for each activity) and 16, 26 or 18 binary observation nodes for the intersect, duplicate or union mapping, respectively.

Table 4 shows the resulting accuracies of all the mappings. We see that the 'union' mapping outperforms the other mappings.

Accuracy in %	Timeslice		Class	
	Mean	Std	Mean	Std
Intersect	45.4	9.4	43.5	8.2
Duplicate	52.3	7.1	45.7	6.4
Union	66.8	10.2	58.2	9.7

Table 4: Timeslice and class accuracies using our transfer learning method for the various kinds of mappings.

Experiment 2: Learning comparison

The goal of this experiment is to compare the performance of our method with a fully supervised approach and with a naive transfer learning approach. In the fully supervised approach we use labeled training data from house 1 and house 2 to learn the model parameters. We use a cross-validation technique called leave-one-day-out, in which we use one day for testing and the remaining days for training, cycling over all the days of the house 2 dataset. In the naive transfer learning approach we perform EM using *only* the house 1 labeled training data and use the learned parameters to perform inference on the house 2 dataset. In all approaches union mapping was used.

By comparing our method with the fully supervised approach, we see how close we are to achieving the best possible results using this model and this dataset. While the comparison with the naive approach shows us how much

our method improves the results over a straight forward approach.

Table 5 shows the accuracies for the various approaches. We see that our method improves the results in comparison with the naive approach.

Accuracy in %	Timeslice		Class	
	Mean	Std	Mean	Std
Supervised	70.9	9.4	64.6	8.0
Our method	66.8	10.2	58.2	9.7
Naive	64.6	11.8	54.5	3.2

Table 5: Timeslice and class accuracies using supervised, our method, or the naive approach.

The confusion matrix for the supervised approach can be found in table 6, for our method in table 8 and for the naive approach in table 7. We see that all three approaches completely fail to classify showering. Both our method and the naive approach fail to classify preparing breakfast. Furthermore, we see that the increase in performance of our method in comparison with the naive approach is mainly with respect to the 'other acts', 'prepare dinner' and 'prepare drink' activities.

Discussion

The experiments show that our method of transfer learning for activity recognition can be successfully applied using the union mapping. By combining the sensors in a function group we were able to effectively use all the sensors that are of importance to the various activities. Furthermore, it allowed us to transfer the knowledge of the correlations between the various sensors and the activities from one context to the next. The other mappings were less successful, because either important sensors were excluded or because important correlations were missed.

An important element of our method is that we use both labeled and unlabeled data during the EM iterations. An alternative approach would be to first learn the model parameters using only the labeled data from house 1 and then use these parameters to perform EM using only the unlabeled data from house 2. However, the problem with this approach is that after the initialization there are no constraints in leading the algorithm to the activities we are interested in. It is very well possible it will find a set of parameters that fit the data better, but will cause the model to recognize meaningless activities such as going to the hallway.

The confusion matrices from experiment 2 show us that our method mainly improved the recognition of the 'other acts.' activity. Because the 'other acts.' activity is not a clearly defined activity, but rather a collection of any activity that we do not monitor, it is very person specific. By using EM we are able to adjust the parameters to account for this person specific behavior accordingly. The confusion matrices also show us that the showering and the prepare breakfast activity are not recognized correctly at all. This is due to the high level of ambiguity in the sensor patterns of house 2. Showering is confused with toileting and 'other acts.',

	Other acts.	Out of House	Toileting	Showring	Sleeping	Breakfast	Dinner	Drink
Other acts.	37.0	34.7	4.3	0.6	17.6	2.8	2.4	0.6
Out of House	3.4	77.5	3.6	0.0	10.5	1.5	3.4	0.0
Toileting	0.0	0.0	95.7	2.9	0.0	0.0	1.4	0.0
Showring	7.7	0.0	92.3	0.0	0.0	0.0	0.0	0.0
Sleeping	0.0	0.0	2.0	0.0	98.0	0.0	0.0	0.0
Breakfast	26.1	0.0	17.4	0.0	0.0	34.8	21.7	0.0
Dinner	0.0	0.0	0.0	0.0	0.0	36.8	63.2	0.0
Drink	0.0	0.0	6.7	0.0	0.0	0.0	0.0	93.3

Table 6: Confusion Matrix for *supervised* approach using cross validation. The values are percentages.

while preparing breakfast is mainly confused with preparing dinner. This is because the toilet and the shower are shared in a single room, which makes the information from the door sensors not enough to separate the two activities. However, installing a simple humidity sensor would most likely solve this problem. The same applies to preparing breakfast, house 2 had relatively few number of sensors in the kitchen.

Conclusions

This paper introduces a method to use the knowledge about activity recognition from one context and apply it in a new context, so no labeled training data of the new context is required. A number of mappings were proposed and the effectiveness of these mappings was compared. We showed our method improves the performance of our activity recognition model over a naive approach. Our evaluation was performed using datasets recorded in two different houses. This gives us early experimental evidence that our method works, we are recording more datasets to evaluate our method in a more settings.

Acknowledgements

This work is part of the Context Awareness in Residence for Elders (CARE) project. The CARE project is partly funded by the Centre for Intelligent Observation Systems (CIOS) which is a collaboration between UvA and TNO, and partly by the EU Integrated Project COGNIRON (The Cognitive Robot Companion).

	Other acts.	Out of House	Toileting	Showring	Sleeping	Breakfast	Dinner	Drink
Other acts.	11.3	34.7	4.7	1.7	17.8	0.0	29.1	0.6
Out of House	0.8	76.1	4.4	0.0	10.5	0.0	7.6	0.6
Toileting	0.0	0.0	94.2	2.9	1.4	0.0	1.4	0.0
Showring	0.0	0.0	92.3	0.0	7.7	0.0	0.0	0.0
Sleeping	0.0	0.0	2.0	0.0	98.0	0.0	0.0	0.0
Breakfast	0.0	0.0	17.4	0.0	0.0	0.0	78.3	4.3
Dinner	5.3	0.0	10.5	0.0	0.0	26.3	52.6	5.3
Drink	0.0	0.0	6.7	0.0	0.0	0.0	0.0	93.3

Table 7: Confusion Matrix for the *naive* transfer learning approach. The values are percentages.

	Other acts.	Out of House	Toileting	Showring	Sleeping	Breakfast	Dinner	Drink
Other acts.	30.2	33.8	4.3	1.7	16.9	0.0	2.8	10.3
Out of House	0.2	71.9	3.1	0.0	10.5	0.0	0.4	13.9
Toileting	0.0	0.0	94.2	2.9	1.4	0.0	0.0	1.4
Showring	0.0	0.0	92.3	0.0	7.7	0.0	0.0	0.0
Sleeping	0.0	0.0	2.0	0.0	98.0	0.0	0.0	0.0
Breakfast	21.7	0.0	17.4	0.0	0.0	0.0	0.0	60.9
Dinner	0.0	0.0	15.8	0.0	0.0	5.3	68.4	10.5
Drink	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0

Table 8: Confusion Matrix for *our* transfer learning method. The values are percentages.

References

- Bilmes, J. 1997. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models.
- Bilmes, J. A. 2006. What HMMs Can Do. *IEICE Trans Inf Syst* E89-D(3):869–891.
- Bishop, C. M. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.
- Duong, T. V.; Bui, H. H.; Phung, D. Q.; and Venkatesh, S. 2005. Activity recognition and abnormality detection with the switching hidden semi-markov model. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, 838–845. Washington, DC, USA: IEEE Computer Society.
- Fogarty, J.; Au, C.; and Hudson, S. E. 2006. Sensing from the basement: a feasibility study of unobtrusive and low-cost home activity recognition. In *UIST '06: Proceedings of the 19th annual ACM symposium on User interface software and technology*, 91–100. New York, NY, USA: ACM Press.
- Katz, S.; Down, T.; Cash, H.; and et al. 1970. Progress in the development of the index of adl. *Gerontologist* 10:20–30.
- Landwehr, N.; Gutmann, B.; Thon, I.; Philipose, M.; and De Raedt, L. 2007. Relational transformation-based tagging for human activity recognition. In Malerba, D.; Appice, A.; and Ceci, M., eds., *Proceedings of the 6th International Workshop on Multi-relational Data Mining (MRDM07)*, 81–92.
- Lester, J.; Choudhury, T.; Kern, N.; Borriello, G.; and Hanford, B. 2005. A hybrid discriminative/generative approach for modeling human activities. In *IJCAI*, 766–772.
- Logan, B.; Healey, J.; Philipose, M.; Tapia, E. M.; and Intille, S. S. 2007. A long-term evaluation of sensing modalities for activity recognition. In *UbiComp '07*, 483–500.
- Mihalkova, L.; Huynh, T.; and Mooney, R. J. 2007. Mapping and revising markov logic networks for transfer learning. In *22nd AAAI Conference on Artificial Intelligence*, 608–614.
- Patterson, D. J.; Fox, D.; Kautz, H. A.; and Philipose, M. 2005. Fine-grained activity recognition by aggregating ab-

stract object usage. In *ISWC*, 44–51. IEEE Computer Society.

Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2):257–286.

Raina, R.; Ng, A. Y.; and Koller, D. 2006. Constructing informative priors using transfer learning. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, 713–720. New York, NY, USA: ACM.

Schmidt, A. 2002. *Ubiquitous Computing - Computing in Context*. Ph.D. Dissertation, Lancaster University.

Tapia, E. M.; Intille, S. S.; and Larson, K. 2004. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive Computing, Second International Conference, PERVASIVE 2004*, 158–175.

van Kasteren, T. L. M., and Kröse, B. J. A. 2007. Bayesian activity recognition in residence for elders. In *Intelligent Environments, 2007. IE 07. 3rd IET International Conference*, 209–212.

van Kasteren, T.; Noulas, A. K.; Englebienne, G.; and Kröse, B. 2008. Accurate activity recognition in a home setting. In *Tenth International Conference on Ubiquitous Computing (UbiComp'08)*.

Wilson, D. H.; Long, A. C.; and Atkeson, C. 2005. A context-aware recognition survey for data collection using ubiquitous sensors in the home. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, 1865–1868. New York, NY, USA: ACM Press.