This is the published version:


Available from Deakin Research Online:

http://hdl.handle.net/10536/DRO/DU:30044623

Reproduced with the kind permissions of the copyright owner.

Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Copyright : 2005, IEEE
Factored State-Abstract Hidden Markov Models for Activity Recognition Using Pervasive Multi-modal Sensors

# Dung T. Tran 1, Dinh Q. Phung 1, Hung H. Bui 2, Svetla Venkatesh 1

Department of Computing, Curtin University of Technology
GPO Box U1987, Perth, WA 6845, Australia

Email: {trand,phungquo,svetha}@cs.curtin.edu.au
Email: bui@ai.sri.com

Abstract

Current probabilistic models for activity recognition do not incorporate much sensory input data due to the problem of state space explosion. In this paper, we propose a model for activity recognition, called the Factored State-Abstract Hidden Markov Model (FS-AHMM) to allow us to integrate many sensors for improving recognition performance. The proposed FS-AHMM is an extension of the Abstract Hidden Markov Model which applies the concept of factored state representations to compactly represent the state transitions. The parameters of the FS-AHMM are estimated using the EM algorithm from the data acquired through multiple multi-modal sensors and cameras. The model is evaluated and compared with other existing models on real-world data. The results show that the proposed model outperforms other models and that the integrated sensor information helps in recognizing activity more accurately.

1. INTRODUCTION

The field of human activity recognition has received much attention from the research community due to the growing needs of automatic and sophisticated surveillance such as aged care monitoring [16] and smart homes [8]. Basically, the purpose of activity recognition is to infer people’s behaviours from low-level data acquired through sensors in a given setting, based on which other critical decisions are made. For instance, in smart home environments for aged care monitoring [8], based on the information provided by cameras and other pervasive sensors, the system needs to automatically monitor the occupant and determine when they need assistance, raising an alarm if required. These environments are complicated because of the vast number of sensors, resulting in a large state space.

Much research into activity recognition has been carried out and typical methods used are probabilistic models [10] for modelling and recognizing complex behaviours. Flat models such as the Hidden Markov Model (HMM), the Coupled Hidden Markov Model (CHMM) [3] and the Variable Length Markov Model (VHMM) [7] are quite efficient at recognizing simple activities but are not able to capture the hierarchical structure of complex activities. The Layered Hidden Markov Model (LHMM) [13], the Abstract Hidden Markov Model (AHMM) [5], [4] and the Hierarchical Hidden Markov Model (HHMM) [6], [11] have been developed to address these problems by introducing a hierarchical probabilistic model in which each level represents an activity at a particular level of abstraction. In [5], [4], Bui et al. introduce a robust model, the AHMM, a framework for probabilistic plan recognition which efficiently exploits the hierarchical structure in policy execution. Nguyen [12] and Osentoski [14] have applied this powerful model for behaviour recognition in complex home environments based on the position data. Liao et al. [9], [15] employed slightly different models to AHMMs in inferring and predicting users’ mode of transportation from GPSs (Global Positioning System), which is also applied by Yin [17] to recognize high-level goals in indoor environments.

The underlying problem with these models is that the state space can be very large, especially in the case where many state variables other than position are monitored. This makes the number of parameters in the transition model extremely large, and thus difficult to estimate from a reasonable amount of training data. In [1], [2], Boutillier et al. introduced the concept of factored representation to represent the effects of action on state. It exploits the fact that, in the general case, only some state variables are affected and the rest stay the same when performing an action. Thus, state transitions can be represented more compactly by representing each state variable in a node rather than by combining them into a very large node. In other words, the state is factorized into state variables. Though proposed in [5], the work did not provide details of implementation. Our work will investigate the use of factored states in detail with an AHMM to address the issue of activity recognition in pervasive environments.

The aim of this paper is three-fold: (a) to explore the factored representation in AHMMs, (b) to construct a probabilistic model based on AHMMs using factored states for recognizing complex behaviours in a smart home setting, we name this model the factored state AHMM (FS-AHMM), (c) to compare the model with other existing models and evaluate its performance in terms of accuracy and robustness. We argue
that with the use of factored states, the FS-AHMM will yield better recognition results than existing models.

The contribution of our work is two-fold: (1) we apply the concept of factored states in extending AHMMs that have been proven to be powerful in recognizing complex behaviours [12], [14]. As a result, the AHMM's inherent problem of an explosion of state space is efficiently addressed. (2) We provide a framework that is able to integrate multi-modal sensors, for example trajectory data and other multi-modal sensor data, whilst in previous works [12], [14], [17], [15], only trajectory data is used. Incorporating multi-modal sensor data will reinforce the recognition performance of the model.

The layout of the rest of the paper is as follows. Section 2 introduces the concept of factored state representations and outlines the proposed FS-AHMM that applies this concept in detail. Section 3 presents the implementation of the model, describes the experiments carried out with the model for activity recognition and the results are then discussed. Finally, section 4 summarises our work and its contribution.

2. THE FACTORED STATE - ABSTRACT HIDDEN MARKOV MODEL

A. Abstract Hidden Markov Model

The Abstract Hidden Markov Model, proposed by Bui et al. [5], [4], is a probabilistic framework for modeling people behaviour in a stochastic environment at multiple levels of abstraction that is able to capture the hierarchical structure of behaviours. A high behaviour \( \pi^t \) is decomposed into a sequence of simpler behaviours at lower levels and in turn, behaviours at low-levels are refined into a sequence of lower level behaviours \( \pi^{t-1} \). When a high-level behaviour is executed, it calls recursively a lower level behaviour with some probability until reaching the level 1 behaviours (primitive behaviours). A called low-level behaviour terminates with some probability at some certain states \( s \). Figure 1 illustrates the DBN representation of the AHMM with two layers in two time slices: \( t = 1 \) and \( t \).

B. The concept of factored state representation

The factored representation is proposed by Boutilier et al. [1], [2] to compactly represent the effects on state when performing actions. In general, the state space is substantial because each state is the product of its state components, \( s = (s^1, s^2, \ldots, s^D) \). It is therefore very costly to represent the state transitions of an action. As illustrated in Figure 2(a), in the non-factored form, the network of state transitions of each action is fully connected and in this way, the state transition matrix is very huge. To deal with this problem, Boutilier [2] introduces the concept of factored representations to exploit the fact that, in the general case, there are independencies among state variables that hold in certain contexts. It also means that some redundant probabilistic links can be removed from the state transition network. Thus, it is more beneficial to factorize the state into state variables than to combine them into a large node where each state variable is represented as a different node. As shown in Figure 2(b), the state transition network is simplified when certain independencies are used which will result in reducing the complexity of the action’s network as well as the number of parameters of the model.

We take advantage of this and apply this factored representation to represent the state transitions in the AHMM. This enables the new model to integrate several monitored variables rather than position state \( p_t \) only. In the rest of this section, we present the FS-AHMM in detail.

C. DBN Representation

The FS-AHMM model is an extended version of the AHMM, described in [4], in which the state is factored into state variables. A FS-AHMM can be represented as a Dynamic Bayesian Network as shown in Figure 3 with two consecutive time slices. In comparison with the AHMM, the amalgamated state \( s_t \) is now factorized into other state variables: \( s_t = (p_t, w^1_t, \ldots, w^D_t) \), in which each state variable represents the hidden state of a monitored sensor and \( D \) is the number of sensor devices used in the model. At each time slice \( t \) of the DBN, the set of variable nodes \( X_t \) includes \( \pi_t, e_t, p_t, w^1_t, \ldots, w^D_t, o^p_t, o^{w^1}_t, \ldots, o^{w^D}_t \) which represent the activity nodes, the terminating variable nodes, the hidden state nodes of position and sensor devices and the observation nodes of the state variables respectively. The links represent the causal relationships between the nodes. As a consequence of using the factored state representation, some redundant links between state variables of two consecutive time slices are removed instead of being fully connected as in the non-factored representation. For shortness, we denote vector \( w = (w^1, \ldots, w^D) \) and vector \( o^w = (o^{w^1}, \ldots, o^{w^D}) \) from later on.

D. Parameters of the FS-AHMM

Similar to the AHMM, the set of parameters of the FS-AHMM model, denoted as \( \theta \), includes the set of initial parameters \( I \), the set of transition parameters \( A \) and the set of observation models \( B: \theta = (I, A, B) \). The parameters are defined as

![DBN representation of AHMM with two layers in two time slices](image-url)
Fig. 2: (a) DBN of the state transition in non-factored form. The transition network is fully connected. (b) DBN of the state transition in factored form. The network of state transitions is partially connected.

Fig. 3: DBN representation of FS-AHMM with two time slices: $t-1$ and $t$. follows.

#### Initial parameters $I$

1) Initial activity probability $I_a$: the probability that activity $\pi$ is selected at the beginning of the execution of the top activity.
2) Initial position state probability $I_p$: the probability that activity $\pi$ starts at the landmark position $p$.
3) Initial device state probability $I_w$: the probability that the status of device sensor $i^{th}$ is ON when activity $\pi$ is started.

#### Transition parameters $A$

1) Terminating probability $T_{e=\text{true}}^\pi$: the probability that activity $\pi$ terminates at position state $p$ and the corresponding current device states are $w$. The terminating probability is initialized to a high value if the position state is in the vicinity of particular landmarks such as the fridge, stove or dining chair, and assigned to a small value otherwise.
2) Activity transition probability $A_{w}^{\pi',e,p,w}$: the probability that the previous activity $\pi$ is switched to activity $\pi'$

given the current state variables $e, p, w$. It is defined as:

$$A_{w}^{\pi',e,p,w} = \begin{cases} \delta(\pi', \pi) & \text{if } e=\text{false} \\ P(\pi'|p, w) & \text{if } e=\text{true} \end{cases}$$

where $\delta$ is the Kronecker function. Each $\pi$ is transitted with high probability near particular landmarks and with low probability otherwise.

3) Position state transition probability $A_{p}^{w':w}$: the probability that position state $p'$ is moved from $p$ when performing activity $\pi$.

4) Device state transition $A_{w}^{w'}$: the probability that the state of the device $i^{th}$ is switched from $w^i$ to $w'^i$ when performing activity $\pi$ at position state $p$.

#### Observation model $B$

1) Movement observation model $B^e$: the probability of emitting position observation $p'$ given the true position state is $p$.
2) Sensor device observation model $B^w$: the probability of observing the state of the device $i^{th}$ when the true state is $w^i$.

#### The number of parameters of the state transition model

The number of parameters of the state transition model in the case of using the non-factored representation is $|\mathcal{P}|(|\mathcal{W}| + |\mathcal{D}|)^2$ versus $|\mathcal{P}|(|\mathcal{W}| + |\mathcal{D}|)^2$ when using the factored representation. Thus, the number of parameters in the case of the factored state model is greatly reduced, especially when $D$ is large.

#### E. Inference and learning in FS-AHMM

##### Inference

The task of inference in the DBN model is to compute the filtering distribution $P(X_t|O_{1:T})$ where $X_t$ is the set of hidden variable nodes in the time slice $t$ and $O_{1:T}$ is the observation sequence up to time $t$. In the case of the FS-AHMM, $X_t$ and $O_t$ are defined as follows: $X_t \triangleq (\pi_t, e_t, p_t, w_t)$ and $O_t \triangleq (o^{w}_t, o^{e}_t)$. The inference method used in this model is the exact inference method that is formulated in a similar manner to the forward-backward algorithm in the HMM [10].

The inference procedure will produce the filtering distributions: $\gamma_t(X); \xi_t(X_{t-1}, X_t)$, denoted as: $\gamma_t(X) \triangleq Pr(X_t = X|O_{1:T})$ and $\xi_t(X_{t-1}, X_t) \triangleq Pr(X_{t-1}, X_t|O_{1:T})$ that is later
used to estimate the parameters in the learning procedure.

**Learning**

The task of learning in the DBN model is to estimate the parameters of the model \( \theta \) with the set of training data \( O = (O^k, k = 1 \ldots N) \), using the Expectation-Maximization (EM) algorithm. The EM objective is to find a maximum likelihood (ML) estimation \( \theta^* = \arg \max_{\theta} Pr(O|\theta) \). The EM algorithm is an iterative procedure to estimate \( \theta^* \) consisting of an E-step and M-step, which is guaranteed to converge to a local optimal point. It first computes the expected sufficient statistics (ESS) for the parameters using the smoothing distributions, \( \gamma, \xi \), of the inference procedure and then re-estimates the parameters by normalizing the ESSs for these parameters.

**3. EXPERIMENTAL RESULTS**

In this section, the proposed FS-AHMM model is applied to address the issue of activity recognition in pervasive environments.

A. **Environment and activity description**

1) **Environment description:** The proposed model is used for recognizing complex daily activities in a smart home environment. The inputs of the model are trajectory data obtained from a camera tracking system and sensor data emitted from sensor devices, which are mounted on objects that occupants are likely to interact with as a consequence of an activity in daily living. The camera tracking system outputs the trajectory of the occupant as a sequence of \((x_i, y_i)\) coordinates which are then mapped into position state \( p \) of the environment. As shown in Figure 4, the home environment is divided into a grid of 7x4 cells where each cell has a size of 1m x 1m. Therefore, the environment has a total of 28 position states \( p \) from 1, 2, ..., 28. The objects in the home are also arranged as illustrated in the figure. Their locations are equivalent to special landmarks, the lightly shaded cells.

Due to the inherent noise in the environment, the observed value of the true position state is occasionally one of the neighboring cells of the true position state. The camera observation model is learned separately by comparing the observed value with its labeled ground-truth.

![Fig. 4: The layout of the environment.](image)

2) **Sensor description:** In our smart home environment, we installed several multi-modal sensors mounted on objects at some landmarks. The state of the sensor is changed when a certain activity associated with it is activated. We use two types of “state-change” sensors. Reed switches are placed on objects that the occupant is likely to interact with such as the fridge, stove, etc and pressure mats are placed at designated locations.

In the experiments, we use four sensors laid out as in Figure 4, \( w_1, w_2, w_3, w_4 \). Three of them are reed switches attached to the fridge, stove and the cupboard respectively. The other is a pressure mat which covers the dining chair. These sensor devices will provide information to indicate whether the occupant is interacting with the object. That is, the sensors mounted on the fridge, stove and cupboard specify whether the fridge or cupboard is being opened or closed and whether the stove is being turned on or off. The pressure mat informs us as to whether or not the occupant is sitting on the dining chair.

The sensor devices also suffer from inherent noise of the environment and the sensors themselves and the observation models of sensors are learned separately as for the camera.

3) **Activity description:** We use the FS-AHMM for recognizing the set of complex morning daily activities: [1] a milk and cereal breakfast, [2] a scrambled eggs on toast breakfast, [3] a fruit and yogurt breakfast. Each of these top-level activities consists of a set of less complex activities, which correspond to variable \( \pi \), in a hierarchical manner and in turn, each of these activities involves getting to the landmark and operating device. The top-level activities are described in detail as follows.

**Activity [1]:** Getting milk (door→fridge→take out milk→to table→put down milk) → Getting cereal (table→cupboard→take out cereal→to table) → Eating (have breakfast) → Washing-leaving (to sink→to door).

**Activity [2]:** Preparing scrambled eggs on toast (door→fridge→take out eggs→to stove→cook on stove→to table) → Eating (have breakfast) → Washing-leaving (to sink→to door).

**Activity [3]:** Preparing fruit (door→fridge→take out fruit→to cupboard→take out plate→to table) → Getting yogurt (table→fridge→take out yogurt→to table) → Eating (have breakfast) → Washing-leaving (to sink→to door).

Each top-level activity is executed in the order described above and recorded in a sequence of observations with the length of time from 80 seconds to 100 seconds. The walking pattern of each activity varies, meaning that there is confusion in the trajectory between activities.

**B. Data and evaluation**

1) **Data:** We collected 75 labeled sequences, with 25 labeled sequences for each top-level activity, in which each sequence contains position information returned from the camera tracking system and event data emitted from the sensor devices. For instance, a data sequence might be the sequence
of tuples having the form: \( (p = 13, w^1 = 1, w^2 = 0, w^3 = 0, w^4 = 0) \), which indicates that the occupant is observed at position state 13, the fridge is opened and the status of the stove, cupboard and dining chair is OFF respectively. For each top-level behaviour, 25 data sequences are divided randomly into two sets of sequences, a training set of 12 sequences and a testing set of 13 sequences. The model for that activity is then trained using an EM algorithm.

2) Evaluation: After training, three sets of parameters for three models, \( \theta_1, \theta_2, \theta_3 \), are obtained. The recognition performance of the FS-AHMM is evaluated at two levels of activity, one at the top-level and the other at the level of less complex activity. At both levels, recognition performance is evaluated on two criteria: accuracy rate and early detection which are defined as follows. Accuracy rate is the ratio of the number of testing sequences that the system recognizes correctly to the total number of testing sequences, and early detection is the ratio of the period of time the testing sequence is recognized correctly to the time length of that testing sequence. In the problem of activity recognition, the higher the accuracy rate and the lower the early detection, the better the recognition performance of the model.

At the top-level activity, for each labeled testing sequence \( O^{test} \), the log likelihood of that testing sequence with each of set of parameters associated with each model \( \theta_i \), \( i = 1 \ldots 3 \), is computed and the maximum likelihood model is chosen as the winning model. For example, as illustrated in Figure 5, the testing sequence best fits model 3 and it is detected correctly from the time \( t = 9 \), meaning that early detection is \( 9/82 = 10.95\% \). We perform the same experiments with the flat HMM and the AHMM to compare our FS-AHMM against these models.

At the lower level, we do online filtering to evaluate the recognition performance of the lower-level activity \( \pi \). For this, the quantity \( P(\pi_i | O_{1:T}) \) needs to be computed to segment activity \( \pi \) from the whole testing sequence \( O_{1:T} \). It can be derived from \( \alpha_t(X) \equiv P_t(X_t = X | O_{1:t}) \), by marginalizing \( \alpha_t \) out unnecessary variables in \( X_t \). The activity \( \pi_i \) with the highest probability in online filtering is then compared with the ground-truth to determine the recognition performance at the lower level. To prove for robustness, we pre-segment the top-level activity into lower-level activities and use an HMM to train each lower-level activity to compare the obtained result of the HMM with that of the FS-AHMM.

C. Results and discussion

1) Performance of the FS-AHMM vs. that of the flat HMM: In this experiment, both the FS-AHMM and the flat HMM are evaluated in the case of using trajectory data and sensor data. As can be seen from Table 1, the accuracy rate of the FS-AHMM is much better than that of the flat HMM (98.50% versus 70.00%). It also shows that the FS-AHMM is able to detect the winning activity earlier than the flat HMM, FS-AHMM’s early detection of 31.90% as compared to the flat HMM’s of 45.36%. Thus, our proposed FS-AHMM outperforms the flat HMM in terms of both the accuracy rate and early detection.

<table>
<thead>
<tr>
<th>Table 1: The FS-AHMM vs. The Flat HMM.</th>
</tr>
</thead>
<tbody>
<tr>
<td>flat HMM</td>
</tr>
<tr>
<td>Accuracy</td>
</tr>
<tr>
<td>Early detection</td>
</tr>
</tbody>
</table>

2) Performance with trajectory data alone vs. trajectory and sensor data: In this experiment, we compare the performance of models with trajectory data alone (flat HMM and AHMM) with that of models supplemented with sensors (FS-AHMM and flat HMM). Table 2 shows that the performance of the flat HMM in three cases of using trajectory data alone, sensor data alone and trajectory data incorporated with sensors in recognizing the complex activities is low. The accuracy rate of the flat HMM with trajectory integrated with sensors is higher than that of flat models with trajectory or sensors alone (70.00% vs 60.60% and 60.50%). The early detection of the flat HMM incorporated with sensors is lower than that with trajectory only (50.45%, 45.36% vs 26.30%). Both the FS-AHMM and the AHMM obtain a good accuracy rate, which is much better than the flat HMM, 98.50%, 100% and 87.80% respectively. The accuracy rate of the FS-AHMM supported with sensor data is significantly improved in comparison with that of the AHMM with trajectory only, but its early detection is a bit lower, 31.90% vs 27.04%. The above results show that the flat HMM is unsuitable for our activity recognition setting, even with the help of sensors, since it cannot capture the hierarchical structure in complex activities. The FS-AHMM is able to capture the hierarchy in these complex activities and with the support of sensors, its recognition performance is greatly reinforced. Also, the supplemental use of sensors makes the proposed model less dependent on the trajectory data.

Figure 6 illustrates the results of filtering lower-level activities from the whole course of a top-level activity plotted over time for both the AHMM with trajectory only and the FS-AHMM. As can be seen in Figure 6, the AHMM does not segment some lower-level activities correctly but the FS-AHMM does with the support of sensor information. It implies that incorporating sensor information really improves the performance of recognition using the AHMM.

3) Performance of the FS-AHMM in combination of trajectory and sensors.
TABLE 2: PERFORMANCE OF MODELS WITH TRAJECTORY ALONE VS THAT OF MODELS INTEGRATED WITH SENSORS.

<table>
<thead>
<tr>
<th></th>
<th>Trajectory only</th>
<th>Sensors alone</th>
<th>Incorporated with sensors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Flat HMM A HMM</td>
<td>Flat HMM FS-AHMM</td>
<td>Flat HMM FS-AHMM</td>
</tr>
<tr>
<td>Accuracy</td>
<td>60.60%/ 87.80%</td>
<td>60.50%/ 100%</td>
<td>70.00%/ 98.50%</td>
</tr>
<tr>
<td>Early detection</td>
<td>26.30%/ 27.04%</td>
<td>50.45%/ 41.9%</td>
<td>45.36%/ 31.90%</td>
</tr>
</tbody>
</table>

Fig. 6: Activity filtering with: (a) AHMM, (b) FS-AHMM.

Varying sensor noise: This experiment measures the performance of the FS-AHMM when sensor noise, including both environment noise and sensor itself modeled by sensor model $B_{u,t}^{n}$, is varied. Table 3 shows that the early detection when using sensors alone is very low, from 41.0% to 56.81%, but when integrated with trajectory, it improves considerably (from 26.7% to 31.0% ). Moreover, the results also indicate that the trajectory data and sensors can mutually support the performance of the model. For instance, when the noise of sensors is high, 25%, the performance of the FS-AHMM when using sensors only is low too, 40%. But with the support of trajectory, the performance of the FS-AHMM is much better at 93.39%.

Missing sensor reading: In this experiment, we measure the performance of the model when one sensor is assumed to be unavailable. It can be seen from Table 4 that the performance of the FS-AHMM under the above condition is still reasonable compared with the FS-AHMM in the case of normal operation, 96.96% versus 98.5%.

TABLE 3: PERFORMANCE OF FS-AHMM WITH VARYING SENSOR NOISE.

<table>
<thead>
<tr>
<th>Sensor noise</th>
<th>0.5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
<th>25%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensors</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>alone E</td>
<td>100%</td>
<td>100%</td>
<td>93.93%</td>
<td>93.93%</td>
<td>40.00%</td>
</tr>
<tr>
<td>Trajectory</td>
<td>41.90%</td>
<td>52.90%</td>
<td>56.80%</td>
<td>55.80%</td>
<td>25.10%</td>
</tr>
<tr>
<td>and sensors</td>
<td>98.50%</td>
<td>93.39%</td>
<td>93.39%</td>
<td>93.39%</td>
<td>93.39%</td>
</tr>
<tr>
<td></td>
<td>31.90%</td>
<td>29.50%</td>
<td>28.50%</td>
<td>26.70%</td>
<td>27.40%</td>
</tr>
</tbody>
</table>

TABLE 4: PERFORMANCE OF FS-AHMM IN CASE OF MISSING SENSOR READING.

<table>
<thead>
<tr>
<th>FS-AHMM with</th>
<th>Full sensor reading</th>
<th>One missing sensor reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>98.50%</td>
<td>96.96%</td>
</tr>
<tr>
<td>Early detection</td>
<td>31.5%</td>
<td>30.80%</td>
</tr>
</tbody>
</table>

4. CONCLUSION

In conclusion, we have presented our FS-AHMM in solving the problem of activity recognition in pervasive environments with multi-modal sensors. We propose the FS-AHMM as a framework for integrating multi-modal sensors to reinforce the activity recognition performance. The model is constructed on the AHMM with the use of factored representations and can be trained from data captured in the smart home lab. The results have demonstrated that the proposed model is more efficient and robust in comparison with other existing models, the AHMM and the flat HMM, in recognizing complex activity. In addition, it has also been shown that sensor data, incorporated into our model, provides richer information for activity recognition rather than trajectory data only, which is used in most other existing models.

REFERENCES