

# Intelligent Healthcare Service Provisioning Using Ontology with Low-Level Sensory Data

**Asad Masood Khattak, Zeeshan Pervez, Sungyoun Lee and Young-Koo Lee**

Ubiquitous Computing Laboratory, Kyung Hee University, 446-701, South Korea

[e-mail: {asad.masood, zeeshan, sylee}@oslab.khu.ac.kr, yklee@khu.ac.kr]

\*Corresponding author: Sungyoung Lee

*Received May 9, 2011; revised August 31, 2011; accepted August 3, 2011;  
published November 29, 2011*

---

## **Abstract**

Ubiquitous Healthcare (u-Healthcare) is the intelligent delivery of healthcare services to users anytime and anywhere. To provide robust healthcare services, recognition of patient daily life activities is required. Context information in combination with user real-time daily life activities can help in the provision of more personalized services, service suggestions, and changes in system behavior based on user profile for better healthcare services. In this paper, we focus on the intelligent manipulation of activities using the Context-aware Activity Manipulation Engine (CAME) core of the Human Activity Recognition Engine (HARE). The activities are recognized using video-based, wearable sensor-based, and location-based activity recognition engines. An ontology-based activity fusion with subject profile information for personalized system response is achieved. CAME receives real-time low level activities and infers higher level activities, situation analysis, personalized service suggestions, and makes appropriate decisions. A two-phase filtering technique is applied for intelligent processing of information (represented in ontology) and making appropriate decisions based on rules (incorporating expert knowledge). The experimental results for intelligent processing of activity information showed relatively better accuracy. Moreover, CAME is extended with activity filters and T-Box inference that resulted in better accuracy and response time in comparison to initial results of CAME.

---

**Keywords:** Ubiquitous healthcare, activity recognition, ontology, inference engine, ontology matching

---

A preliminary version of this paper appeared in The 10th IEEE Annual International Symposium on Applications and the Internet (SAINT 2010), Seoul, Korea, July 2010. This version includes a concrete analysis, implementation and results of the overall system. This work was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agent)" (NIPA-2011-(C1090-1121-0003).

**DOI: 10.3837/tiis.2011.11.008**

## 1. Introduction

As the human lifespan increases, people are becoming more interested in living a healthy life, which results in high-cost healthcare systems and services. Maintaining good quality and widely available healthcare services at a minimal cost is challenging [1]. Home healthcare systems are becoming a more important form of healthcare service delivery. The management, maintenance, and coordination of healthcare services, educating users, and empowerment of individuals to manage their own health are the main focus. To support this, a powerful, flexible, and cost-effective infrastructure is required for healthcare services that can fulfill the vision of *ubiquitous healthcare (u-healthcare)*. Cloud Computing can potentially provide a huge cost savings, flexibility, high-throughput, and ease of use for different services [2] as well as for healthcare services. For this reason, we have developed an architecture, called Secured Wireless Sensor Network (WSN)-integrated Cloud Computing for u-LifeCare (SC<sup>3</sup>) [3]. Different wireless sensors are deployed that collect real-time data that is transmitted to a Cloud Server through a Cloud Gateway. Based on this real-time data collected by different sensors, SC<sup>3</sup> provides real-time home care and safety monitoring services, an information sharing and exchange facility, emergency connection services, and patient monitoring and care services.

One of the main components of SC<sup>3</sup> is the Human Activity Recognition Engine (HARE) [1] (see Fig. 1). This engine is necessary to provide improved daily medical care and real-time reaction to medical emergencies, and identifying patient activities (i.e., Activity Recognition (AR)) is a prerequisite. Low level activities are defined as simple motions or actions detected by the sensing device that are very general and unclear in meaning such as hand movement. High level activities are the linkage of low level activities in a sequence with reference to context to make it more understandable. For example, in the process of making tea, picking up the cup, boiling water, using sugar, and using a spoon are all low level activities and are unclear when interpreted separately. However, when these activities are sequenced together and are used in context with the kitchen location, then they comprise “making tea” as a high level activity.

The existing systems are based on a simple condition and action model [4], not using context information including time, location, and user profile. In some cases, the existing systems use imperfect low level context information [5], which makes the system results unpredictable. Their focus is more on environmental sensors (e.g., smoke detector, infrared control, and GPRS modem) rather than on real-time human activity. Traditional systems are moving from location or time based reminder systems toward context-aware activity recognition. However, humans rely on several modalities including the five classical senses and other senses such as thermoception (temperature) and equilibrioception (balance and acceleration) together with context information such as location, and time for everyday tasks. Currently, to the best of our knowledge, there is no systematic way to integrate multi-modalities such as profile information, vision with motion, environment, location, and time to infer human intentions, whereas the traditional systems are based on simple activity with a condition and action model. Our focus in this paper is on CAME [6] as a component of our proposed HARE [1] (see Fig. 1) that is beyond the limitations of traditional systems. We propose the integration of all the activities detected using different types of sensors together along with context and profile information of the subject. We model the activities in domain ontology within the explicit context of the activity and execution pattern. In addition, we apply semantic reasoning to infer high level activities (user intention for performed activity) and use

it for decision making. This will help in enhancing capabilities of healthcare systems to facilitate more personalized recommendations and decision making, and it has tremendous value for intelligent/efficient service provisioning.

For the CAME implementation, we use different sources of information to avoid the possibility of missing information or imperfect context information [5]. For context representation and profile information, we use an ontology and have developed a semantic structure for information representation. Ontology is formally defined as *an explicit and formal specification of a shared conceptualization* [7].

Sensors are deployed to collect real-time data about a person's activities and environmental information. Then with the use of an ontology (containing expert knowledge of the medical domain and user profile information) these detected activities are intelligently manipulated to infer higher level activities and also to make a situational analysis. The experimental results of the match-making process of CAME yielded good results. Rule-based filtering for situation analysis and decision making has verified our claims, and the results of activity recognition and manipulation are very encouraging in terms of accuracy. We have also extended our proposed CAME [6] to the analysis and decision making process. The extended CAME uses both A-Box (instance level) and T-Box (structure level) inference techniques that confer better accuracy compared to the traditional CAME. A filter is also implemented in extended CAME to filter out an "unknown activity" during the match making phase. This not only improves the accuracy of the extended CAME compared to the conventional CAME, but also results in better system response time.

This paper is arranged as follows. Section 2 is the background study of existing systems. Section 3 discusses activity recognition techniques used for the detection of low level activity. In Section 4 we discuss our proposed CAME architecture that takes the output of activity recognition engines and uses it for decision making. Section 5 presents our results and a discussion of these results. We conclude our discussion in Section 6.

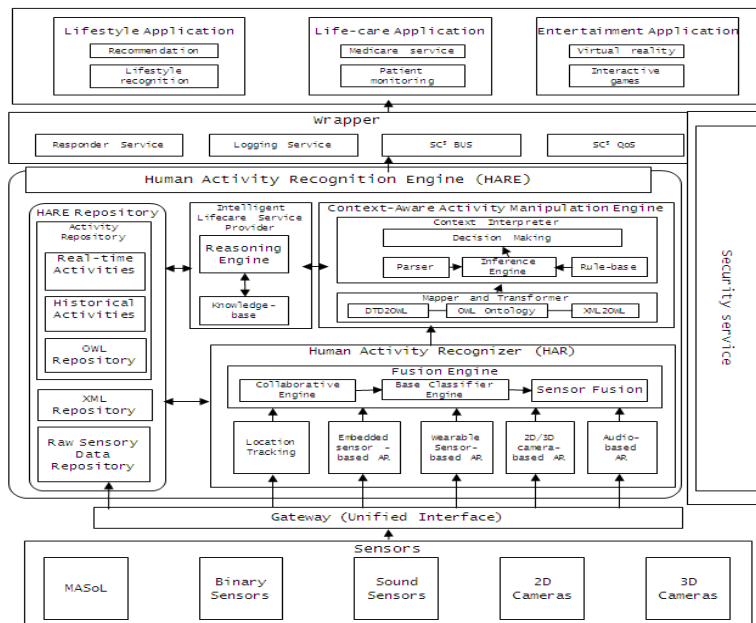


Fig. 1. HARE system model [1].

## 2. Background Study

Numerous developments in industry and academia have already begun or are being used currently to facilitate better healthcare. In July 2008, the Ministry for Health, Welfare, and Family Affairs, Korea released *u-Care System for a Solitary Senior Citizen (SSC)*. SSC monitors human health at home and provides limited services such as 24 hours×365 days safety monitoring services for a SSC, emergency-connection services, and information sharing services. *Microsoft* and *Google* are two pioneers who have brought Cloud healthcare platforms to reality for healthcare applications and services with low cost and increased performance.

Microsoft developed a platform to store and maintain health and fitness information, called *Health Vault* [8]. It is a Cloud service that helps people collect, store, and share their personal health information. Google provides a personal health information centralization service known as *Google Health* [9]. The service allows Google users to volunteer their health records into the *Google Health* system. Volunteered information can include health conditions, medications, allergies, and lab results. *Google Health* uses the information to provide the user with a merged health record, information on conditions, and possible interactions between drugs, conditions, and allergies. The *Unified Cloud Interface (UCI)* standardization [10] or Cloud broker serves as a common interface for interaction with remote platforms, systems, networks, data, identity, applications, and services. UCI is composed of a semantic specification and ontology. The ontology provides the actual model descriptions, while the specification defines the details for integration with other management models.

Various wireless technologies are currently being used in healthcare. CodeBlue [11] is one of the technologies developed by Harvard Sensor Network Lab. CodeBlue is based on a publish/subscribe model for its different services. CodeBlue mainly supports physicians and nurses that monitor patients; however, some of the focus is shifting toward research on providing reminders for the elderly to perform daily life activities [12]. These are mostly plan-based approaches to decide when and how to prompt subjects effectively, and are thus centered around time-based activities. To overcome the limitations of this system, a location-based reminder system was introduced [13], where a key element for reminders in this system is the subject's location. But in fact, context for reminders is more important than simple location or time and context includes both location and time as subsets. *ComMotion* [14] is an example of a context-aware system that supports reminder applications that use only one sensor and are mainly based on time. It addresses how and when to prompt the subjects. *HYCARE* [15] is the most recent reminder system that takes context into consideration, and it develops a novel scheduling mechanism that can coordinate with different reminder services and remedy possible conflicts.

The idea in [16] is based on the Markov process for decision making (a decision model capable of taking into account the uncertain effects of an action with the tradeoffs of both short-term and long-term objectives). The system is designed to monitor elderly dementia patients and provide them autonomous guidance to complete their activities of daily life. The authors of that study focused on facilitating hand washing activity using a video camera. A conceptual model/space is developed for hand washing and then used by the system when the activity is being performed by patients, and it provides reminders for different steps from the conceptual model/space. The system in [4] is a more realistic system that not only uses ontology to incorporate context in intelligent processing of the collected information, but also focuses on the information collected from sensors such as a smoke detector, GPRS modem, infrared control and X10 appliances that actually facilitate more in-home care for the person than person healthcare. This system is based on the Event-Condition-Action (ECA) model;

however, for support in healthcare, the system needs to collect data on the activities performed by humans in addition to environmental information.

The systems discussed above do not use real-time activities or only use a single type of real-time activity performed by the subject and then generate reminders or even make decisions based on that information. They only consider the context to the level of time and location, which results in inflexible system behaviour. These systems can mostly be categorized as reminder systems or homecare systems, but the important thing is to facilitate healthcare where these systems fail to perform. The existing Cloud-based Healthcare system does not integrate wireless sensor networks, which is necessary to obtain real-time information on patients and/or the environment in order to monitor and analyze emergency situations.

### 3. Activity Recognition

An accurate and robust human Activity Recognition (AR) and Manipulation system is necessary to achieve a better understanding of a situation and for decision making in u-healthcare environments. Many research groups are addressing the problem of determining Activities of Daily Life (ADL), where they mostly focus on one or a few activities [14], use one or few techniques [5][15][16], and often lack robustness to identify those activities in complex situations. Our proposed AR system incorporates video [17], accelerometer [18][19], location [20], and physiological data to improve the robustness and scope of ADL capabilities. For example, it will recognize many complex activities such as taking pills, exercising, watching TV, eating, teeth brushing, falling down, and heart attacks. To provide intelligent service recommendation and situation analysis, contextual information (expert knowledge) is used with the help of ontology to eliminate the possibility of erroneous information generated from heterogeneous devices operating in that environment. The Human Activity Recognition Engine (HARE) shown in Fig. 1 is composed of such sub-components. The components are used to detect activities. Detected activities are then used by CAME for situation analysis and decision making. Description of these components is given below.

The accuracy of the video-based AR depends significantly on the accuracy of human body segmentation [21]. The proposed system in [22] is used, but its accuracy is still questionable. Our methodology [23] is to incorporate an evolving term based on the Bhattacharyya distance [24] to the CV energy functional such that not only the differences within each region are minimized, but the distance between the two regions is maximized as well (for details please refer to [17]). After obtaining a set of body silhouettes segmented from a sequence of images, ICA (independent component analysis) [25 and 26] is applied to get the motion features of that sequence. ICA focuses on local features rather than global features as in PCA (principal component analysis) [26]. The extracted features are then symbolized using vector quantization algorithms such as K-mean clustering [27]. Finally, the symbol sequence is used to generate a codebook of vectors for training and recognition.

Based on [28], a recognition model called “semi-Markov Conditional Random Fields (semiCRF)” [18] is being developed for activity recognition. Furthermore, the developed model is a novel algorithm that helps to reduce the complexity of the training phase by more than ten times in comparison with the original work. The algorithm is proposed for computing gradients of the target function by extending [28]. It reduces the complexity of computing each gradient from  $O(NTM^2D)$  to  $O(3TM(M+D)+NTD)$ , where T, M, D, and N are the length of the input sequence, number of label values, and maximum duration of a label, and the number of gradients, respectively. This technique is used to identify most of the activities that are used in the later portion of the paper for system evaluation. For details on the overall workings of the

proposed algorithm, please refer to [18].

The objective of location tracking is to provide location information about the object of interest. Video-based AR fails in situations such as crowded tracked objects or because of privacy requirements (e.g., no cameras are allowed in bedrooms and bathrooms). Some approaches have been proposed using radio frequency (RF) [29][30] where the learning phase is followed and the result of inference is directly the location. A Neural Network is applied for our system implementation, not to infer the location directly but to infer the distances to the beacons, and then use Push-Pull Estimation (PPE) [20] to determine the location. This technique is in fact developed for an outdoor environment; with a high density of tracked objects and few beacons, the localization accuracy can be improved significantly. For the indoor tracking problem, it is still a good candidate solver because it is a successive refining method; the result of the last inference is used as an initial guess for the next inference, which is prohibited in the previous system.

#### 4. Context-aware Activity Manipulation Engine

The use of ontology in activity recognition is a relatively new area of research. It helps in better understanding the activity in a given context. Activities recognized with the help of different sensors (i.e., body, location, motion, and video sensors) are low level activities and cannot be used for analysis and decision making. With the help of ontology, where we use the contextual information and link all the related activities in a chain, and customized rules we get higher level activities that are more usable for decision making.

We define an ontology comprising the domain expert information representation as follows:

$$O = \{C, S, I, R, A\}$$

$O$  is the ontology containing  $C$  concepts with  $I$  instances connected with one another using  $S$  slots.  $R$  is the restrictions applied on different  $C$ ,  $I$ , and  $S$ .  $A$  is the auxiliary axioms and annotations used to enrich information in the knowledgebase that can later be used in match making and decision making processes. The concepts represent different diseases, symptoms, daily life activities that are permitted in the context of a particular disease, detailed patient information including profile information, and patient's daily activity schedule. Ontology with the rules helps in properly extracting the higher level activity of activities set in a series, e.g., low level activities such as bending, sitting, jumping, and walking using ontology will result in a higher level activity, e.g., exercising (see Fig. 2; the Description Logic rule is given in Fig. 3). A detailed representation of the activities and related information are given in Fig. 4.

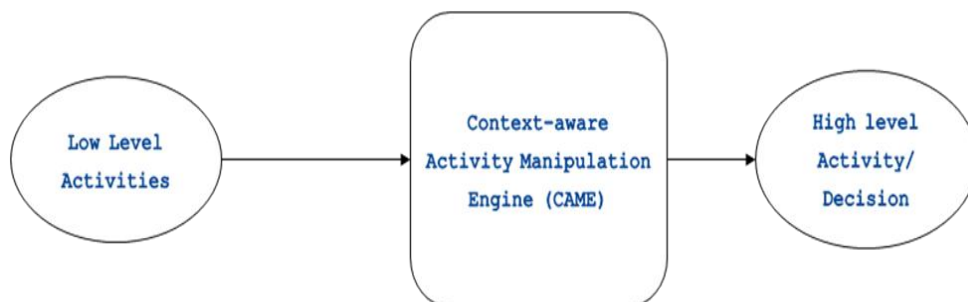


Fig. 2. Abstract model of CAME for activity manipulation.

Rule	
Exercise $\sqsubseteq$ $\forall$ Activity $\sqcap$ Activity.performedBy.Person = 1 Person $\sqcap$	
( $\exists$ Activity.hasContents(bending) $\sqcup$ $\exists$ Activity.hasContents(jacking) $\sqcup$	
$\exists$ Activity.hasContents(jumping) $\sqcup$ $\exists$ Activity.hasContents(runing) $\sqcup$	
$\exists$ Activity.hasContents(skipping) $\sqcup$ $\exists$ Activity.hasContents(sliding) $\sqcup$	
$\exists$ Activity.hasContents(walking) $\sqcup$ $\exists$ Activity.hasContents(waving) = $\exists$ 2	
Activity.distinctContents	

**Fig. 3.** DL rule for exercising based on detected activities.

CAME is one of the main components of HARE. It is the process of inferring high level activities from low level activities recognized by different sensors. A component-based framework architecture diagram of CAME is shown in [Fig. 5](#) with extensions, while the detailed description of all the components is given below.

A *Knowledgebase (KB)* serves as the backbone of CAME. It is responsible for proper communication of information among all the components of CAME. It stores all possible types of activities that a human body can perform in different contexts/situations along with information about different activity priorities for different users and groups of users. Proper engineering of the KB is most important in CAME development. To engineer the KB we have to look at the same problem from different angles. We have developed a Knowledgebase for activity representation and have divided activities into *Temporal* and *Non-Temporal Activity* classes. In the same way we have modelled all the other concepts in KB related to activity as well as to the subject that performs the activities.

```

activityOnto:Person_Instance_654
a    activityOnto:Person ;
activityOnto:has activityOnto:Patient , activityOnto:PhD , activityOnto:High_Age;
activityOnto:hasAge                68;
activityOnto:hasDesignation        activityOnto:Professor;
activityOnto:hasID                 654;
activityOnto:hasName               "Mr. Raazi".

activityOnto:Activity_Instance_20090614140013345
a    activityOnto:Activity ;
activityOnto:hasConsequentAction   activityOnto:Rule;
activityOnto:hasID                 345;
activityOnto:hasName               "Running";
activityOnto:hasType               "Motion";
activityOnto:isA                   activityOnto:Room_Instance_Living;
activityOnto:performedAtTime       2010:11:14:14:00:13;
activityOnto:performedBy           activityOnto:Person_Instance_654.

activityOnto:Activity_Instance_20090614140013347
a    activityOnto:Activity ;
activityOnto:hasConsequentAction   activityOnto:Rule;
activityOnto:hasID                 347;
activityOnto:hasName               "Walking";
activityOnto:hasType               "Motion";
activityOnto:isA                   activityOnto:Room_Instance_Living;
activityOnto:performedAtTime       2010:11:14:14:00:18;
activityOnto:performedBy           activityOnto:Person_Instance_654.

activityOnto:Activity_Instance_20090614140013346
a    activityOnto:Activity ;
activityOnto:hasConsequentAction   activityOnto:Rule;
activityOnto:hasID                 346;
activityOnto:hasName               "Jumping";
activityOnto:hasType               "Motion";
activityOnto:isA                   activityOnto:Room_Instance_Living;
activityOnto:performedAtTime       2010:11:14:14:00:15;
activityOnto:performedBy           activityOnto:Person_Instance_654.

```

```

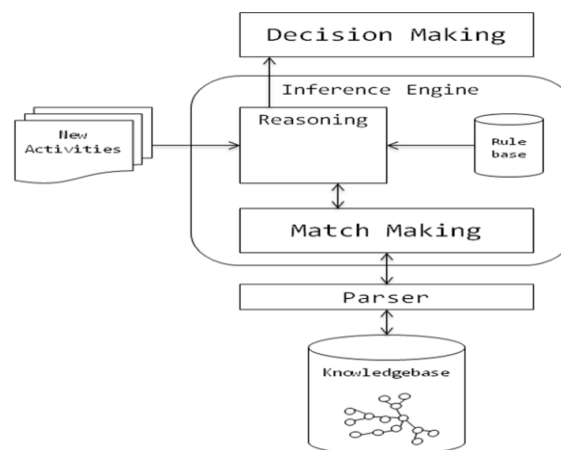
activityOnto:Activity_Instance_20090614140013348
a  activityOnto:Activity ;
activityOnto:hasConsequentAction  activityOnto:Rule;
activityOnto:hasID                  348;
activityOnto:hasName                "Bending";
activityOnto:hasType                "Motion";
activityOnto:isA                    activityOnto:Room_Instance_Living;
activityOnto:performedAtTime       2010:11:14:14:00:22;
activityOnto:performedBy           activityOnto:Person_Instance_654.

```

**Fig. 4.** OWL representation (using N3 notation) of activities that result in an exercise activity

Activity related information is extracted from the XML and Text files generated by *Activity Recognition Engines*. Thus the activity needs to be formally represented in a predefined semantic structure [31]. For this reason, CAME formally represents (see Fig. 4) the activities that are extracted in the previous step, while the representation is provided by the KB. Fig. 4 is the N3 representation of newly detected activities (with activity labels) including the context information, i.e., type of activity, time the activity is performed, location where activity is performed, sensor used, and subject who performed the activity. Each subject has its own historical meta information and profile information that contribute to decision making, which is linked with the subject instance.

The activities detected are then verified for two reasons: (1) to verify the consistency of the newly recognized activity against the KB developed for the activities, and (2) after consistency verification, to enforce existence verification for the activity, i.e., is this activity already present in the Knowledgebase or not? If it is not present then it is populated in the KB. In order to manipulate information from the KB, *Parser* is responsible for properly handling all operations regarding that matter. The Parser normally communicates with the Activity Representation component to properly represent the activity; it also parses the KB for a variety of different reasons such as activity verification and decision making. The Parser is also used to populate the KB for newly recognized activities.



**Fig. 5.** Context-aware Activity Manipulation Engine (CAME); inferring high level activities from low level activities.

In order to understand the context of an activity and to extract high level activities from low level activities recognized by various sensors, an *Inference Engine* is required for the analysis of these activities and to make proper personalized decisions on behalf of human users. This requires activity information with respect to contextual information and infers high level



activities. To facilitate the decision making, we have incorporated expert (medical doctor) knowledge with the help of the description logic rules given in [Table 2](#). We developed a Two Phase Algorithm (see Algorithm-1) to infer high level activities and make decisions. Two-phase filtering for decision making is used since only the results of match making are insufficient in healthcare systems. The first phase is the match making process. Here, newly detected activity is matched against the already existing activities in the KB and for this purpose we use semantic matching techniques. The match making process incorporates different information and at various levels, such as: the activity, its possible “before and after” activities, the time the activity is performed, the location of the performed activity, the subject performing the activity, and the subject’s medical history that might be useful for activity manipulation (e.g., disease, symptoms, age, and doctor prescription). The second phase is the rule-based filtering of the matched results returned from the KB for the newly detected activity using the inference engine. After the inference process, the system can make decisions or give suggestions for or against different activities based on the description logic rules incorporating expert knowledge. Thus the *Decision Making* module is responsible for performing/executing actions against the suggestions made by the *Inference Engine*.

**Algorithm CAME (Process Activity):**

This algorithm predefines the matching threshold of activity to  $\psi = 0.70$ .

**Input:** Human activities ontology  $O$  containing profile information.

**Input:** Instance of newly detected activity  $A$ ,  $A \in O$ , which lists activities in the ontology  $O$ .

**Input:** Decision rules  $R$  used for decision making.

**Output:** Higher level activity  $HA$  and/or decision commands  $DC$  based on the inference results.

1. /\* Receive newly detected activities from activity recognition engines containing subject and sensor information \*/
2. GetActivity( $A$ )
3. /\* Parse ontology model for all the activities, filter out unknown activities, initiate match making process with defined threshold, calculate semantic affinity and return matched activities in a sub-ontology model \*/
4.  $O \leftarrow \text{Filter}(\text{Parse}(O), \neg \text{“Unknown Activity”})$
5.  $O_{\text{sub}} \leftarrow \text{SemanticAffinity}(A, O, \psi)$
6. /\* Read rules from the rule file and maintain in an array  $R$  \*/
7.  $R[] \leftarrow \text{ReadRules}(\text{Rules File})$
8. /\* Start inference process using forward chaining \*/
9. **foreach**  $R_{\Delta} \in R[]$  **do**
10.     /\* Process for decision command \*/
11.     **If**  $R_{\Delta} : A \sqcap \exists O_{\text{sub}}.\text{Activity} \sqcap \exists O_{\text{sub}}.\text{Action}$  **then**
12.         /\* Store the decision command \*/
13.          $DC[] \leftarrow \{x \mid \langle R_{\Delta}, x \rangle \text{Action}\}$
14.     **else**
15.         /\* Process for higher level activity \*/
16.          $HA[] \leftarrow O \sqcap \{x \mid \langle A, x \rangle \text{hasAssociation}\}$
17.     **Endif**
18. **End**
19. /\* Execute most valued commands and store higher activities\*/
20.  $\text{Execute}(DC[\psi])$
21.  $O' \leftarrow O + HA[\psi]$
22. **End**

**Algorithm 1.** CAME algorithm for activity manipulation.

## 5. Implementation and Results

We deployed our system on a Cloud to support Alzheimer's patients. Our general system deployment design is shown in Fig. 6. The patient's home includes a kitchen, a bedroom and a living room. The overall installed devices are: MASol Sensor, Biosensor, MICAZ, Camera, PDA, Cell Phone, Phone, Laptop, and Computer. The Mobile Activity Sensor Logger (MASoL) serves in the infrastructure layer to collect and monitor human and environmental information. MASoL is our own developed sensor logger that contains 13 axis sensors, which gathers and stores raw activity data. For convenience, all kinds of sensors are integrated into one board with flexible software architecture for efficient data collection. A biosensor is used to detect the biological readings of the human body such as blood pressure and blood glucose. MICAZ is a low power consuming device used in the wireless sensor network that has the additional capability to function as a router. It is used primarily for monitoring temperature, barometric pressure, and high-speed sensor data transmission.

The MASoL, wearable-sensors, video cameras, and location tracking engines are deployed in the patient's home to collect sensory data and images. We have also deployed motion sensors for simple motion detection of the patient. This information is then used by CAME for situational analysis and decision making. We deploy a Cloud gateway in the living room to collect data from all sensors and cameras. It connects to the Cloud via a high-speed Internet router. Doctors, nurses, and the patient's relatives (e.g., his daughter) can easily access this information via a Web2.0 interface.

Ontology defines formal semantics for information, allowing information to be processable by computer systems and system agents. It defines real-world semantics for resources, allowing them to link machine processable content in a meaningful way based on consensual terminology. Researchers have different approaches to using ontology for the introduction of context in the system. In [4], the authors only focused on the location and time information of an activity. In our approach, we use the location and time information in addition to user profile information and information about the environment in which the user is currently located in order to make the system more personalized.

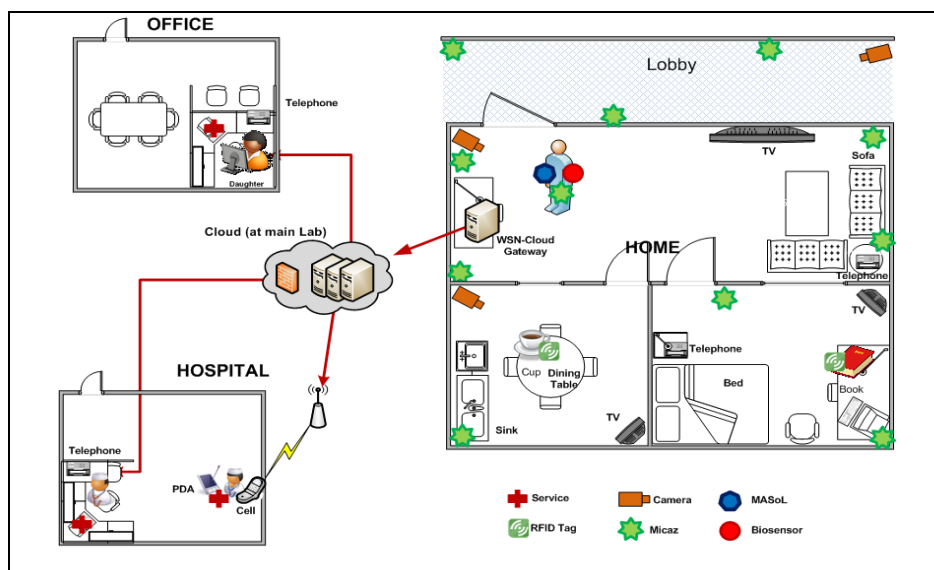


Fig. 6. Overall set-up in the patient's home.

To implement CAME with all its components, we used Jena2, Protégé, Protégé-OWL, Arq, and Pellet 3.4 (for inference). The outcome of CAME is partially dependent on the results of activity recognition modules that are responsible for detecting activities from the raw data collected by the sensors. We wrote SPARQL queries that are executed using the Jena2 API while using the functionality of the Arq API to obtain information about some specific activity and their consequent actions. **Fig. 7** is a query for obtaining information about some particular activity and their consequent actions. For this query, the activity is provided by the system or user and then its corresponding information is extracted.

```
"SELECT ?activityName ?hasConsequentAction ?type ?performedBy ?performerName ?time
?actionDes ?performedAt ?performedAtLoc ?hasType ?actionTime WHERE { <" +
strNS + strActivity + "><" + strNS + "hasName> ?activityName ." +
"<" + strNS + strActivity + "><" + strNS + "hasConsequentAction> ?hasConsequentAction ." +
"<" + strNS + strActivity + "><" + strNS + "hasType> ?type ." +
"<" + strNS + strActivity + "><" + strNS + "performedAtTime> ?time ." +
"OPTIONAL {<" + strNS + strActivity + "><" + strNS + "performedBy> ?performedBy} ." +
"OPTIONAL {?performedBy <" + strNS + "hasName> ?performerName} ." +
"?hasConsequentAction <" + strNS + "hasActionDescription> ?actionDes ." +
"?hasConsequentAction <" + strNS + "hasType> ?hasType ." +
"?hasConsequentAction <" + strNS + "hasTime> ?actionTime ." +
"OPTIONAL {?hasConsequentAction <" + strNS + "hasPerformedAt> ?performedAt} ." +
"OPTIONAL {?performedAt <" + strNS + "hasName> ?performedAtLoc}}";
```

**Fig. 7.** SPARQL query to extract all the corresponding information of an activity.

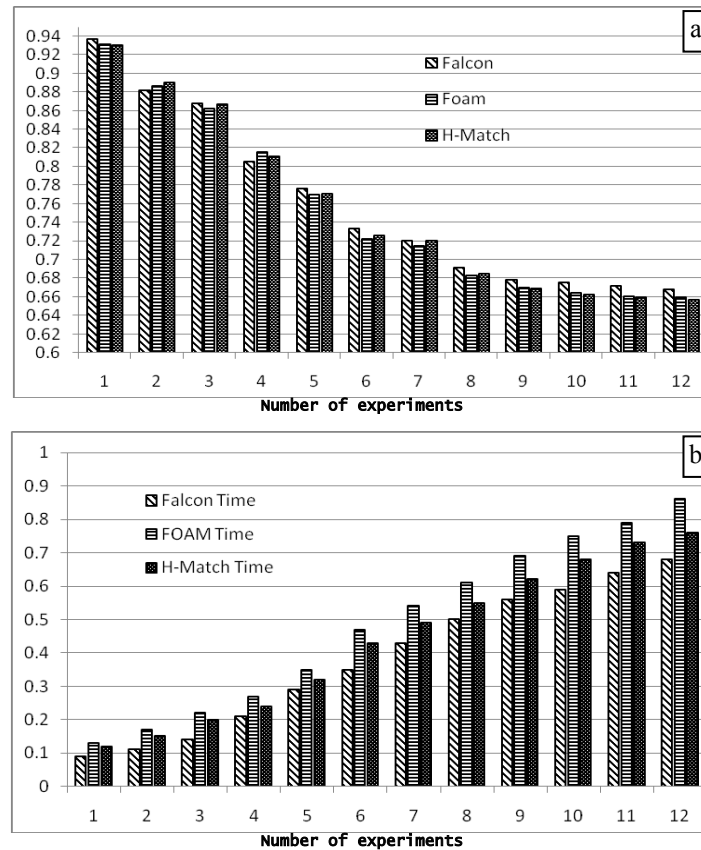
Our proposed algorithm is a two-phase algorithm, i.e., Match Making and Rule-Based Filtering. For the match-making process, different algorithms for ontology matching (e.g., Falcon [32], FOAM [33], H-Match [34], Lily [35], and Prompt [36]) were tested for their performance. In order to test these algorithms, we used standard data sets of *Human* and *Mouse* ontologies (available online at <http://oaei.ontologymatching.org/>). In our case, the number of activities in the KB increase as the everyday activities performed by the subjects are stored. To handle such cases and select appropriate matching algorithm, different incremental changes were introduced in the Human and Mouse ontologies and the algorithms were checked for all of these different versions. **Table 1** shows the matching time for all of these algorithms on the data sets. From these systems, we dropped Lily as it is not openly available and Prompt as it was only focusing on syntactic matching.

**Table 1.** Matching results for FOAM, Falcon, Lily, Prompt, and H-Match with respect to time. Units of time in this table is in minutes.

Human and Mouse Ontology	FOAM Time	Falcon Time	Lily Time	Prompt Time	H-Match Time
HuV1 and MoV1	204.13	5.04	38.01	1.53	8.43
HuV2 and MoV2	207.10	5.01	35.57	1.5	8.22
HuV3 and MoV3	211.14	4.47	35.17	1.48	8.23
HuV4 and MoV4	212.22	5.43	35.44	1.52	8.56

In our given situation, we need to find a match in the KB for a particular activity that has been performed by the subject. For this purpose, we tested FOAM, Falcon, and H-Match on our real collected data to determine the accuracy and execution time for these algorithms. **Fig. 8-(a)** and **Fig. 8-(b)** shows the results of Falcon, FOAM, and H-Match for accuracy and execution time, respectively. These algorithms have their own pros and cons, the discussion of which is beyond the scope of this paper. In our given customized situation, we found Falcon to be more suitable for the match making process, though there is little difference among the

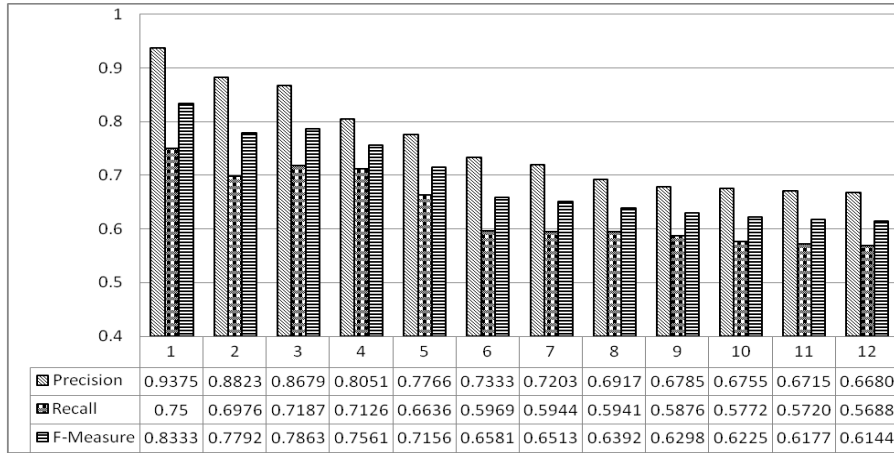
results. The results of these algorithms in terms of both accuracy and execution depend strongly upon the *threshold* value set for the match and the number of *iterations* for finding the match. We used Falcon for the purpose of match making newly detected activities against the already stored activities and related information in the KB.



**Fig. 8. (a)** The precision of the Falcon, FOAM and H-Match algorithms and **(b)** a comparison of the execution time for these algorithms.

We tested CAME using 12 different experiments with an increasing number of activities. As discussed above, the numbers of activities increase as the subject continues to perform different activities and these are stored in the KB. All activities in these 12 experiments are real-time activities detected by the sensors discussed above. In **Fig. 9**, the y-axis is the % of Precision, Recall, and F-Measure for the match making process while the x-axis represents the number of experiments. The graph in **Fig. 9** shows that the precision, recall, and f-measure decreases with increasing number of activities. However, with the increasing number of experiments, precision, recall, and f-measure are smoothed with an average of 0.759, 0.636, and 0.692, respectively. The video demonstration of our overall system deployment and operation is available online.<sup>1</sup> The result of the match making phase is not accurate enough to be used for decision making. Therefore, after match making, the results are filtered through the rules given in **Table 2** in order to generate appropriate suggestions and recommendations and to make decisions.

<sup>1</sup> <http://www.youtube.com/watch?v=FfRpsjD3brg>



**Fig. 9.** Precision, recall, and f-measure of CAME for match making against the number of performed experiments with an increasing number of activities.

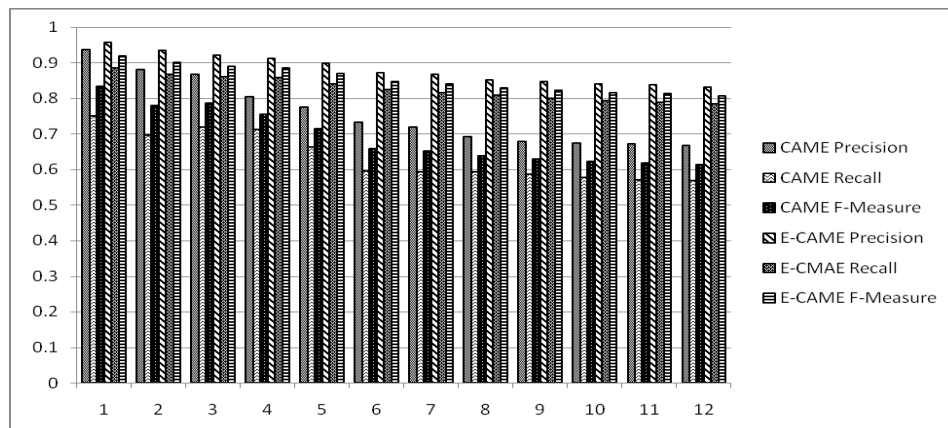
**Table 2.** Rules for recommendations, decision making, and higher level activity recognition.

<b>Rule1</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{eating}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1)$
<b>Rule2</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{taking medicine}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1)$
<b>Rule3</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{reading}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1)$
<b>Rule4</b> $\exists \text{Activity}(a1) \sqcap \neg \text{hasContents}(\text{taking medicine}) \sqcap \text{hasNextActivity}(a2) \sqcap \exists \text{Activity}(a2) \sqcap \text{hasContents}(\text{eating}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{Activity.Create}(a2) \sqcap \text{reminder}(\text{take medicine})$
<b>Rule5</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{reading}) \sqcap \text{hasNextActivity}(a2) \sqcap \exists \text{Activity}(a2) \sqcap \text{hasContents}(\text{TV On}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{Activity.Create}(a2) \sqcap \text{turnOff}(\text{TV})$
<b>Rule6</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{doing exercise}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{turnOn}(\text{music})$
<b>Rule7</b> $\text{Exercise} \sqsubseteq \forall \text{Activity} \sqcap \text{Activity.performedBy.Person} = 1 \text{ Person} \sqcap (\exists \text{Activity.hasContents}(\text{bending}) \sqcup \exists \text{Activity.hasContents}(\text{jacking}) \sqcup \exists \text{Activity.hasContents}(\text{jumping}) \sqcup \exists \text{Activity.hasContents}(\text{runing}) \sqcup \exists \text{Activity.hasContents}(\text{skipping}) \sqcup \exists \text{Activity.hasContents}(\text{siding}) \sqcup \exists \text{Activity.hasContents}(\text{walking}) \sqcup \exists \text{Activity.hasContents}(\text{waveing}) = \exists 2 \text{ Activity.distinctContents}$
<b>Rule8</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{unknown exercise}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{reminder}(\text{movements are wrong})$
<b>Rule9</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{entering kitchen}) \sqcup \exists \text{Activity}(a2) \sqcap \text{hasContents}(\text{entering bedroom}) \rightarrow \text{Activity.Create}(a1) \sqcup \text{Activity.Create}(a2) \sqcap \text{turnOn}(\text{lights})$
<b>Rule10</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{leaving kitchen}) \sqcup \exists \text{Activity}(a2) \sqcap \text{hasContents}(\text{leaving bedroom}) \rightarrow \text{Activity.Create}(a1) \sqcup \text{Activity.Create}(a2) \sqcap \text{turnOff}(\text{lights}) \sqcap \text{turnOff}(\text{TV})$
<b>Rule11</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{in the kitchen}) \sqcap \text{hasNextActivity}(\text{null}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{turnOn}(\text{TV})$
<b>Rule12</b> $\exists \text{Activity}(a1) \sqcap \text{hasContents}(\text{sit down}) \sqcap \text{hasNextActivity}(a2) \sqcap \exists \text{Activity}(a2) \sqcap \text{hasContents}(\text{looking at TV}) \rightarrow \text{Activity.Create}(a1) \sqcap \text{Activity.Create}(a2) \sqcap \text{turnOn}(\text{TV})$

In CAME development, we used A-Box inference that only involves instances. Thus we have extended CAME to use the integration of A-Box with T-Box [37]. A-Box inference is

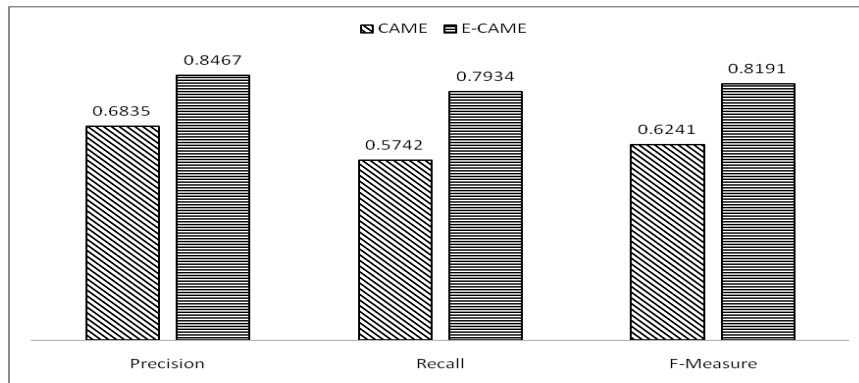
mainly used to verify the consistency at the instance level and to ensure the constraints and conditions of instance membership are satisfied in terms of a concept in particular and to an ontology in general. In contrast, T-Box focuses on the concepts and properties in an ontology. It depends on the axioms and roles defined to infer relationships using functional, transitive, inverse, and symmetric properties. Thus, A-Box focuses on the instances of ontology while T-Box handles the overall structure of the ontology and the constraints on the ontology structure.

Before applying A-Box we used T-Box to limit the number of instances by using the KB structure. Another main cause for the low precision of CAME is the unknown activities detected by the sensors. As we focus only on a set of 18 activities, any activity performed by the Alzheimer's patient not included in the set of 18 activities was reported as an "unknown activity." We have also modified CAME for unknown activities by implementing a filter to prevent the selection of unknown activities during the match making process, which resulted in better system precision. However, we still need to store these activities as they may reveal some interesting new activities for system enhancement. For instance, when taking a bath, there are always two unknown activities; one before and one after. After continuous pattern analysis it is determined that locking and unlocking the bathroom door are detected as unknown activities. Still, the results of the Extended CAME depend more on the sensors deployed to detect human activities in a timely manner.



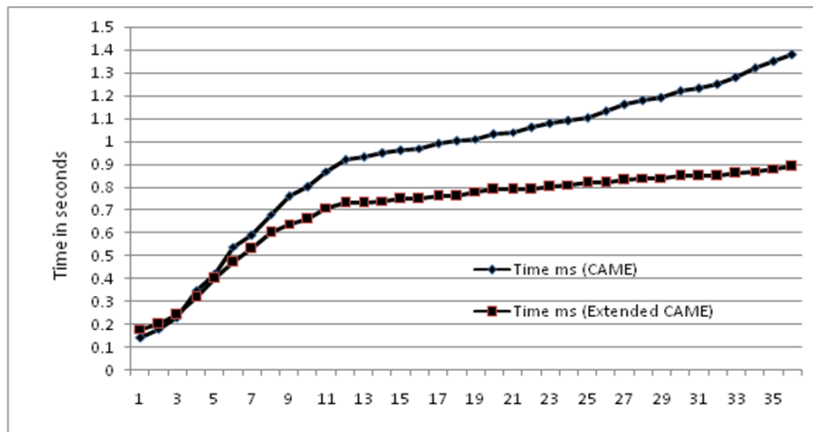
**Fig. 10.** Precision comparison between CAME and extended CAME for 12 different experiments with an increasing number of activities.

We tested CAME and Extended CAME using the same 12 experiments with an increasing number of activities. In **Fig. 10**, the y-axis is the % of precision, recall, and f-measure for the match making process while the x-axis represents the number of experiments. The graph in **Fig. 10** shows that the precision of CAME is less than that of the Extended CAME. The precision of both decrease as the number of activities increases; however, the Extended CAME still maintains a good precision rate. The average precision of CAME and Extended CAME for the 12 experiments is 0.7590 and 0.8810, respectively. To achieve more advanced results, we extended the number of experiments from 12 to 36 and used activities from the MIT Place Lab data set ([http://architecture.mit.edu/house\\_n/data/PlaceLab/PLIA1.htm](http://architecture.mit.edu/house_n/data/PlaceLab/PLIA1.htm)). The increase in experiments from 12 to 36 is syntactic but the activities used are those taken from MIT. CAME and Extended CAME are tested again for 36 different experiments. The average precision, recall, and f-measure for both systems were calculated and are shown in **Fig. 11**.



**Fig. 11.** Comparative analysis of CAME and Extended CAME in terms of precision, recall, and f-measure for a set of 36 different experiments with an increasing number of activities.

After the extensions were made to CAME, the response time of both CAME and Extended CAME was assessed. As shown in **Fig. 12**, the Extended CAME required more time in the initial tests (time is in seconds) to produce results as there are fewer activities and the filtering criteria takes more time. After some tests and increasing the number of activities, Extended CAME performed better than CAME. The greater the number of activities the better the response time of Extended CAME will be when compared to the traditional CAME.



**Fig. 12.** Response time comparison between CAME and extended CAME.

The results achieved by CAME and Extended CAME are highly dependent on the results of the Activity Recognition Engines of HARE. The increasing number of activities has two main effects on the overall workings of the system: 1) a decrease in system precision and 2) a delay in response time. We are currently experimenting with different techniques such as integrating A-Box with T-Box and a hybrid of forward and backward chaining to overcome the first limitation. We are incorporating different indexing techniques to optimize query response time and activity extraction from the KB to address the second limitation.

## 6. Conclusion

The Context-aware Activity Manipulation Engine (CAME) has been presented using a knowledge-driven approach to recognize Activities of Daily Livings (ADLs). The objective of

this system is to infer high-level activities from low level real-time ADLs detected by sensors and to facilitate the provision of better healthcare services. The nature and characteristics of ADLs were analyzed. Based on the analyses, ontology was used to model the ADLs (including activity, location, time, profile, and environmental information), domain knowledge, and expert knowledge. Using ontology with the knowledge engineering practice, a context model for personalized service provisioning and intelligent healthcare facilities has been developed. Ontological modeling of the context and using it for recommendation is the compelling feature of the proposed system. An integrated framework architecture has been developed in addition to the modeled knowledge in order to use the sensed activity information for generating reminders, alerts, and emergency situation analysis for decision making. To achieve better results and provide caregivers an interface for rich interaction, description logic rules have been incorporated. The description logic rules filter out the unnecessary information during decision making. The proposed system was more accurate and had shorter response time for a given situation. To view the detailed operation of the proposed system, please see the video demonstration.

We are planning to test the system on more activities with an extensive set of rules in future studies. We are also planning to work on and provide more services for different kinds of diseases such as stroke and Parkinson's disease. The system in its current state produces some conflicting results when there is more than one subject in the environment. We are trying to overcome this issue as there are some cases in which the appropriate system response is unclear. For example, two subjects are in same room and one is reading a book while the other wants to watch TV. We are developing different heuristics, rules, and scheduling to overcome these issues. Currently planned solutions are: 1) set priorities and levels for different subjects; 2) a subject who initiated activity first will be served in cases of conflict; 3) the location of the conflict will be considered for resolution, i.e., if its subject A's room then the system will respond in favor of subject A; and 4) the subject's schedule will play an important role in handling conflicts, i.e., if a subject's planned activity is relaxing or entertainment then that subject will be served in a common room location.

## References

- [1] A.M. Khattak, L.T. Vinh, D.V. Hung, P.T.H. Truc, L.X. Hung, D. Guan, Z. Pervez, M. Han, S.Y. Lee, Y.K. Lee, "Context-aware Human Activity Recognition and Decision Making," in *Proc. of 12th International Conference on e-Health Networking, Application Services (IEEE HealthCom 2010)*, Lyon, France, July 1-3, 2010. [Article \(CrossRef Link\)](#)
- [2] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computing Systems*, June 2009. [Article \(CrossRef Link\)](#)
- [3] L.X. Hung, P.T.H. Truc, L.T. Vinh, A.M. Khattak, and et al., "Secured WSN-integrated Cloud Computing for u-Life Care," in *Proc. of 7th IEEE Consumer Communications and Networking Conference (CCNC)*, USA, 2010. [Article \(CrossRef Link\)](#)
- [4] F. Wang, K. J. Turner. "An Ontology-Based Actuator Discovery and Invocation Framework in Home Care Systems," in *Proc. of 7th International Conference on Smart Homes and Health Telematics*, pp. 66-73, LNCS 5597, Springer, Berlin, June 2009. [Article \(CrossRef Link\)](#)
- [5] K. Henriksen, J. Indulska, "Modelling and Using Imperfect Context Information," in *Proc. of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, Washington DC, Mar. 14 - 17, 2004. [Article \(CrossRef Link\)](#)
- [6] A.M. Khattak, Z. Pervez, K.K. Ho, S.Y. Lee, Y.K. Lee, "Intelligent Manipulation of Human Activities using Cloud Computing for u-Life Care," in *Proc. of the 10th Annual International*



- Symposium on Applications and the Internet (SAINT 2010)*, Seoul, Korea, July 2010. [Article \(CrossRef Link\)](#)
- [7] T. Gruber, "A Translation Approach to Portable Ontology Specifications," *Knowledge Acquisition*, pp 199-220, 1993. [Article \(CrossRef Link\)](#)
  - [8] Microsoft HealthVault, <http://healthvault.com/>.
  - [9] Google Health, <https://www.google.com/health>.
  - [10] Unified Cloud Interface Standardization, <http://code.google.com/p/unifiedcloud/>.
  - [11] T. Gao, C. Pesto, L. Selavo, Y. Chen, J. Ko, J. Lim, A. Terzis, A. Watt, J. Jeng, B. Chen, K. Lorincz, M. Welsh, "Wireless Medical Sensor Networks in Emergency Response: Implementation and Pilot Results," in *Proc. of the 2008 IEEE International Conference on Technologies for Homeland Security*, May 2008. [Article \(CrossRef Link\)](#)
  - [12] M.E. Pollack, L.E. Brown, D. Colbry, C.E. McCarthy, C. Orosz, B. Peintner, S. Ramakrishnan, I. Tsamardinos, "Autominder: An Intelligent Cognitive Orthotic System for People with Memory Impairment," *Robotics and Autonomous Systems*, vol. 44, no. 3-4, pp. 273-282, 2003. [Article \(CrossRef Link\)](#)
  - [13] T. Sohn, K. Li, G. Lee, I. Smith, J. Scott, W. Griswold, "Place-its: A Study of Location-based Reminders on Mobile Phones," *LNCS*, vol. 3660, pp. 232-250, 2005. [Article \(CrossRef Link\)](#)
  - [14] N. Marmasse, C. Schmandt, "Location-Aware Information Delivery with Commotion," in *Proc. of the 2nd International Symposium on Handheld and Ubiquitous Computing*, Bristol, England, pp. 157-171, 2000. [Article \(CrossRef Link\)](#)
  - [15] K. Du, D. Zhang, X. Zhou, M. Mokhtari, M. Hariz, W. Qin. "HYCARE: A Hybrid Context-Aware Reminding Framework for Elders with Mild Dementia," in *Proc. of the ICOST*, 2008. [Article \(CrossRef Link\)](#)
  - [16] J. Boger, J. Hoey, P. Poupart, C. Boutilier, G. Fernie, A. Mihailidis, "A Planning System Based on Markov Decision Processes to Guide People with Dementia Through Activities of Daily Living," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 2, pp. 323-333, 2006. [Article \(CrossRef Link\)](#)
  - [17] P.T.H. Truc, T.S. Kim, S.Y. Lee, Y.K. Lee, "A Study on the Feasibility of Active Contours on CT Bone Segmentation," *Journal of Digital Imaging*, vol. 23, no. 6, pp. 793-805, 2009. [Article \(CrossRef Link\)](#)
  - [18] L.T. Vinh, L.X. Hung, N.Q. Hung, H.I. Kim, M. Han, Y.K. Lee, S.Y. Lee, "Semi Markov Conditional Random Fields for Accelerometer Based Activity Recognition," *Applied Intelligence*, vol. 35, no. 2, pp. 226-241 2010. [Article \(CrossRef Link\)](#)
  - [19] A.M. Khan, Y.K. Lee, S.Y. Lee, T.S. Kim, "A Triaxial Accelerometer-based Physical Activity Recognition via Augmented Features and a Hierarchical Recognizer," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 5, pp. 1166-1172, Sep. 2010. [Article \(CrossRef Link\)](#)
  - [20] V.H. Dang, V.D. Le, Y.K. Lee, S.Y. Lee, "Distributed Push-Pull Estimation for Node Localization in Wireless Sensor Networks," *Journal of Parallel and Distributed Computing*, July 2010. [Article \(CrossRef Link\)](#)
  - [21] M. Kass, et al., "Snakes: Active Contour Models," *Int. Journal on Computer Vision*, vol. 1, no. 4, pp. 321-33, 1988. [Article \(CrossRef Link\)](#)
  - [22] T. Chan, L. Vese, "Active Contours Without Edges," *IEEE Trans. Image Proc.*, vol. 10, pp-266-277, 2001. [Article \(CrossRef Link\)](#)
  - [23] P.T.H. Truc, S.-Y. Lee, T.-S. Kim, "A Density Distance Augmented Chan-Vese Active Contour for CT Bone Segmentation," in *Proc. of 30th Annual Int. Conf. of IEEE in Medicine and Biology Society*, 2008. [Article \(CrossRef Link\)](#)
  - [24] E. Choi, C. Lee, "Feature Extraction based on the Bhattacharyya Distance," *Pattern Recognition*, vol. 36, no. 8, pp. 1703-1709, Aug. 2003. [Article \(CrossRef Link\)](#)
  - [25] A. Hyvärinen, "New Approximations of Differential Entropy for Independent Component Analysis and Projection Pursuit," in *Proc. of Advances in Neural Information Processing System 10 (NIPS\*97)*, pp. 273-279, MIT Press, 1998.
  - [26] A. Hyvarinen, J. Karhunen, E. Oja, "Independent Component Analysis," *New York: Wiley*, 2001.

- [27] L. Kaufman, P.J. Rousseeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," *Wiley Series in Probability and Statistics*, New York, Nov. 1990.
- [28] S. Sarawagi, W. Cohen. "Semi-Markov Conditional Random Fields for Information Extraction," *Advances in Neural Information Processing Systems*, 2004. [Article \(CrossRef Link\)](#)
- [29] U. Ahmad, A. Gavrilov, S.Y. Lee, Y. K. Lee, "Self-scalable Fuzzy ArtMap for Received Signal Strength Based Location Systems," *Journal of Soft Computing*, vol.12, no.7, p. 699-713, Feb. 2008. [Article \(CrossRef Link\)](#)
- [30] Y. Chen, K. Kleisouris, X. Li, W. Trappe, R.P. Martin, "The Robustness of Localization Algorithms to Signal Strength Attacks: A Comparative Study," in *Proc. of IEEE International Conference on Distributed Computing in Sensor Systems*, San Francisco CA , 2006. [Article \(CrossRef Link\)](#)
- [31] P. Shvaiko, J. Euzenat, "Ten Challenges for Ontology Matching," in *Proc. of the 7th International Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, Aug. 2008. [Article \(CrossRef Link\)](#)
- [32] W. Hu, Y. Qu, "Falcon-AO: A Practical Ontology Matching System," *Journal of Web Semantics*, vol. 6, no. 3, pp. 237-239, 2008. [Article \(CrossRef Link\)](#)
- [33] M. Ehrig, Y. Sure, "Foam - Framework for Ontology Alignment and Mapping; Results of the Ontology Alignment Initiative," in *Proc. of the Workshop on Integrating Ontologies*, vol. 156, pp. 72-76, 2005. [Article \(CrossRef Link\)](#)
- [34] S. Castano, A. Ferrara, S. Montanelli. "Matching Ontologies in Open Networked Systems Techniques and Applications," *Journal on Data Semantics (JoDS)*, vol. 5, pp. 25-63, 2006. [Article \(CrossRef Link\)](#)
- [35] P. Wang, B. Xu, "Lily: Ontology Alignment Results for Oaei 2009," *Ontology Matching (OM)*, 2009. [Article \(CrossRef Link\)](#)
- [36] N. Noy, M. Musen. "The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping," *International Journal of Human-Computer Studies*, vol. 59, no. 6, pp. 983-1024, 2003. [Article \(CrossRef Link\)](#)
- [37] V. Haarslev, R. Moeller, "Racer: A core inference engine for the Semantic Web," in *Proc. of the 2nd International Workshop on Evaluation of Ontology-based Tools (EON-2003)*, Sanibel Island, FL, 2003. [Article \(CrossRef Link\)](#)



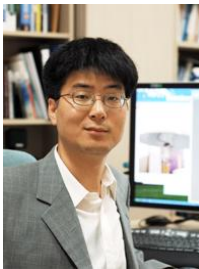
**Asad Masood Khattak** received his BS from the Institute of Computing and Information Technology, Gomal University in 2006. He received his MS in Information Technology from the School of Electrical Engineering and Computer Science, National University of Sciences and Technology in 2008. Since March 2009, he has been working on his PhD in the Department of Computer Engineering at Kyung Hee University, Korea. His research interests include data management, knowledge representation, semantic web, and ontology. He is a member of the IEEE.



**Zeeshan Pervez** is a PhD student at Kyung Hee University (Global Campus) South Korea in the Ubiquitous Computing Lab, and holds a Master's degree in Information Technology from the National University of Science and Technology (NUST), Rawalpindi, Pakistan. His interests include Cloud Computing Security, Intellectual property rights management and service oriented architecture. He is a member of the IEEE.



**Sungyoung Lee** received his BS from Korea University, Seoul, Korea. He earned his MS and PhD in Computer Science from Illinois Institute of Technology (IIT), Chicago, Illinois, USA in 1987 and 1991, respectively. He has been a professor in the Department of Computer Engineering, Kyung Hee University, Korea since 1993. He is a founding director of the Ubiquitous Computing Laboratory, and has been the director of the Neo-Medical Ubiquitous-Life Care Information Technology Research Center, Kyung Hee University since 2006. He is a member of ACM and IEEE.



**Young-Koo Lee** earned his BS, MS and PhD in Computer Science from the Korea Advanced Institute of Science and Technology, Korea. He is a professor in the Department of Computer Engineering at Kyung Hee University, Korea. His research interests include ubiquitous data management, data mining, and databases. He is a member of the IEEE, the IEEE Computer Society, and the ACM.