A Distributed Hierarchical Structure for Object Networks Supporting Human Activity Recognition

Venet Osmani[1], Sasitharan Balasubramaniam[1], Tao Gu[2]

[1]Telecommunications Software and Systems Group
Waterford Institute of Technology
Ireland
[2]Institute for Infocomm Research
Singapore 119613
{vosmani, sasib}@tssg.org , tgu@i2r.a-star.edu.sg

Abstract. Pervasive environments will witness heterogeneous smart embedded devices (e.g. sensors, actuators) integrated into user's living environment (e.g. smart homes and hospitals) and provide a multitude of information that can transparently support user's lifestyle. One promising application resulting from the management and exploitation of this information is the human activity recognition. In this paper we briefly describe our activity recognition architecture and focus on an important management component of this architecture using the concept of object networks. We explore how object networks can integrate various sensor networks and heterogeneous devices into a coherent network through embedded context and role profile and at the same time support distributed context reasoning. The paper also describes the mechanisms used to eliminate and refine context information that is deemed irrelevant due to user behaviour changes over time, by employing the idea of role fitness.

1 Introduction

The proliferation of wireless devices accompanied by multitude of multimedia applications and services has led to the requirement of responsive environments that can dynamically detect and adjust user activities. One current trend in pervasive systems to achieve dynamic user and application support is the ability to infer user's activities from various devices that supports user's lifestyles. Examples of such devices may include mobile devices (e.g. PDA, mobile phones) or embedded micro device networks (e.g. sensors, actuators). Activity recognition has huge potential in supporting users and pervasive applications by automatically deducing the current activity of a user from various objects in the environment coupled with monitoring user's history of interaction. Although in recent years, tremendous amount of focus has been applied towards evaluating context information to support activity recognition, many of these solutions have relied on centralised context gathering and reasoning techniques. For example in [1], authors rely on centralised architecture with a single level of abstraction of context information to process information regarding

user manipulated objects. Each object is instrumented by a RFID tag that is detected using a glove-mounted RFID reader. Another initiative in activity recognition based on low-level sensors has been recorded in [2], where authors have implemented a largely centralised model that has been deployed in a hospital environment. Guralnik and Haigh [3] follows a similar centralised model for observing human behaviour patterns, where the system only deals with one type of sensor (motion detection sensors) that have been used to gather information in a number of living environments. However, centralised context evaluation has a number of drawbacks which includes (i) performance constraints due to bottleneck effects, and (ii) all context rules and reasoning are centrally located and must cater for devices (e.g. sensors, PDAs, actuators) of various types in the environment.

Our activity recognition system counters these drawbacks by employing a decentralised architecture where reasoning mechanisms for activity recognition are performed collectively from different objects and sensors within the environment through local interactions. The technique for activity inference process is realised by the Activity Inference Engine (AIE) that functions in parallel with the Activity Map (AM) [4]. The AM is a repository of all the activities a user has been engaged in the past and is continually refined to support dynamic user behaviour. The activity inference is performed through an Object Network which provides a platform for processing and aggregating heterogeneous context information that is not only limited to hand held devices that users interact with, but also sensor networks surrounding the user's living environment. Through context information found embedded as profile roles in devices and role to role interaction between heterogeneous devices, a more accurate activity recognition process can be achieved. The rest of this paper is organised as follows: Section 2 describes the Object Network and the interaction process and collective reasoning from various objects within the network to deduce user's activity. Lastly section 3 presents the conclusion.

2   Object Networks

In order to deduce user's activity the AIE typically evaluates information from various objects being manipulated as well as information deduced from these objects describing the state of the environment. The idea of the object networks is a network of heterogeneous devices within an environment that have the ability to interact locally in a peer-peer fashion, while supporting distributed context reasoning. Object networks provide a supporting platform for the generation of the required context information to deduce user's activities using a hierarchical structure as shown in Fig 1. The object networks used to efficiently gather context information from sensors do not only support deduction of activities but also provide support for application adaptability for corresponding activities. An important requirement for context information generated from the object network is that it should be at a *high level of abstraction*. Generally deducing high-level context information is challenging since it requires combination of multiple and heterogeneous context sources on top of a processing logic to deduce such information as described by many authors [5] [6] [7].

The AIE is typically contained within the device, directly supporting the application of the user and is usually selected based on the object (leader object) with the highest capability in the environment (e.g. PDA). The selection is based on an optimisation election process. As shown in Fig. 1, the various objects also house an object profile. The object's profile plays an important role in the discovery, election as well as interaction with peer objects since it describes object's capabilities and *roles* the object can be engaged in.

2.1   Object Role(s)

As we briefly mentioned above, in addition to other properties such as identity and communication capabilities, an object will also assume a specific role. In the most simplistic case a role will specify the type of context information that an object is designed to provide. For example a temperature sensor plays the role of 'providing temperature' since this is the object's primary designation. However objects in the environment that the user interacts with are typically much more complex to be described by a single role. For instance a PDA may fulfil multiple roles, from a simple text editor, up to a web server [8].

The role assigned to an object will remain in effect for the duration of the lifetime of the object, unless object's functionality is changed. Particular objects fulfil their role by utilising information generated within the object, for instance information acquired from onboard sensors, while other objects may in addition require information from external sources. As shown in Fig. 1 we collectively deduce the user's activity from the collection of role interactions between the various objects forming a hierarchical structure.

There are two types of object roles that support the AIE of the leader object which includes the *self-sufficient role* and the *conditional role*. Self-sufficient role objects are typically designed for a specific task and provide only a single layer abstraction of context information (e.g. temperature sensor gives only temperature information).

However, a single level of abstraction of context information is typically insufficient, except in the most simplistic instances. Generally multiple levels of abstraction are desired that involves combining context information from multiple and diverse sources. Thus the objects adhering to self-sufficient role are not particularly apt for this task. An object role that cannot be fulfilled unless the precondition of acquiring external information generated outside the object holds true is known as the conditional role. Certainly both the self-sufficient role and the conditional object role play a major part in increasing the level of abstraction of context information; however the conditional role has a higher influence on this process.

The description of each conditional role within an object semantically defines the type of context information that the object requires in order to fulfil the role in question. In addition, the role description also specifies a set of rules that process the required context information and provide the basis to increase the level of abstraction of this information. For each type of context information required to fulfil a particular role, the object sends a query to the object network semantically describing the type of context information it requires. As soon as the query is propagated through the network, objects providing the requested type of information respond and a

'*depends_on*' relationship is established between the requesting object's role and the respondent object's role. Generic representation of this relationship between the role of the requesting object for context information O_a and the role of the context information provider object O_b can be described in terms of Z specification as follows (assuming each object has one role):

$$((O_a, r_a) \wedge (r_a \rightarrow T_r)) \Leftrightarrow \exists (O_b, r_b) \bullet (r_b \rightarrow T_p \wedge T_r \equiv T_p) \wedge (O_a \neq O_b)$$

Therefore a dependency relationship between two objects can be established if and only if exists an object O_b engaged in a role r_b providing information of type T_p which is identical to the type of information T_r requested by a role r_a within the object O_a such that the roles in question are housed in different objects. The last clause of the equation is a 'sanity check' that guards against an object establishing a dependency relationship with itself.
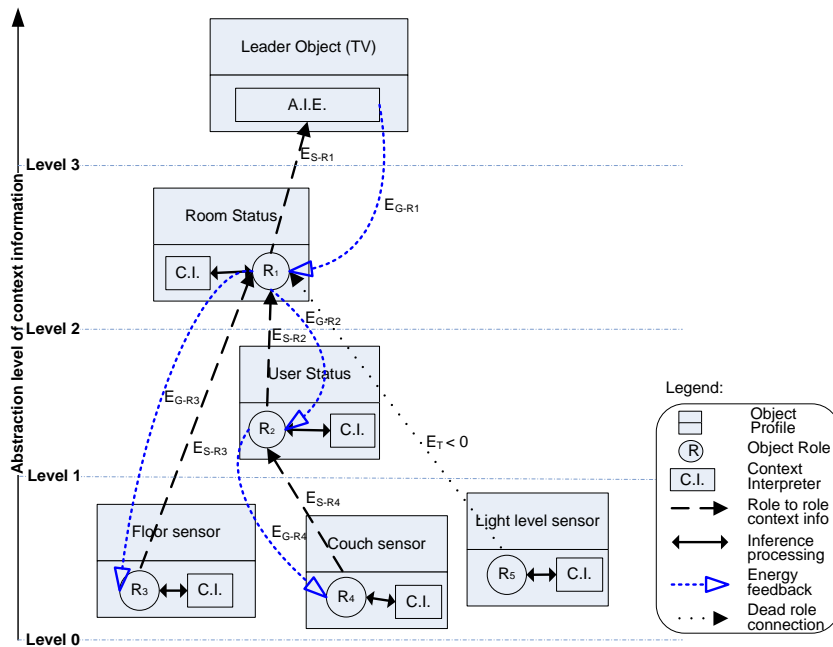

Fig. 1. Objects' role to role interaction

The fact that an object engaged in a conditional role requires information from peer objects that may be engaged in their respective conditional roles creates the necessity to establish a hierarchical context information processing structure. The conditional role will assume at least two levels of abstraction of context information. In instances when the information provider objects specified in the conditional role are further engaged in their respective conditional roles, the context processing and in effect the abstraction level of context information is further augmented with multiple levels of abstraction of context information. In order to realise the hierarchical context processing structure we establish sub-networks within the overall object network. The

sub-networks are smaller entities with a structure similar to the overall object network, containing a *central* object (similar to the leader object) that requests information from the *edge* objects forming the sub-network. As a consequence the object network shifts away from a two level architecture, creating multiple tiers whereby each tier has the effect of increasing the level of abstraction of context information.

2.2 Role fitness for activity recognition refinement

User behaviour is considered dynamic and results in activity characteristics that continually change to support user experience. In order to refine the activity inference process to suit the changing user behaviour we employ the *role fitness* concept. The role fitness is based on the idea that each role is assigned a set amount of energy ($E_T=E_G-E_S$) used to keep the role 'alive', which is inspired from the work of Nakano and Suda [9]. Whenever an object engages a particular role, the role's energy constantly decreases as a result of energy spending (E_S) from producing information pertaining to that role. However, at the same time a role will gain energy (E_G) whenever the role generates relevant information that is utilised by other objects in the object network. The object's role utilising the information will signal back to the object's information generator role that its information has been utilised, as such the amount of energy increases. In the event that a context information is not evaluated based on feedback response, the role will eventually die off due to excessive energy spending and no energy gain ($E_T <=0$).

Therefore in effect this mechanism enables discrimination of the non relevant context information while relevant context information is actively utilised, which reduces the unnecessary context evaluation load at the AIE. Fig.1. depicts a simple scenario of our distributed hierarchical activity recognition systems. After the initial election process the TV is chosen as the leader object. Based on the activity map, the next activity to be deduced is either "watching television" in the living room or "chatting on PC" in the bed room. Based on the object network for each activity, only the object network from "watching a tv" receives events and context information from peer objects. As shown in Fig. 1, this includes three objects (floor sensor, light level sensor, and couch sensor). These sensor roles have established *depends on* relationship with the Room Status object, which in turn has *depends on* relationship with the TV. The three sensors each get the feedback from the Room Status object increasing the energy levels, while the Room Status object gets feedback from the leader object (TV). However user habits have changed and now the user prefers to watch the sports programme with the lights off. Such event is detected by the room status object, resulting in the information about the light levels becoming irrelevant ($E_T <=0$). The light sensor's ratio between the gained energy and spent energy is increasing in favour of the latter, causing the light level sensor role to slowly deplete and eventually die, as shown in Fig. 1.

3   Conclusion

In this paper we have demonstrated mechanisms to manage objects within an environment in order to deduce the activity of a user. We have combined several ideas to achieve the required functionality which includes embedding roles into various devices and supporting role to role interaction to allow high-level context abstraction and distributed context reasoning. While object roles provide the necessary information to enable the activity inference process, the concept of role fitness augments this process to make it dynamic and adjustable to changing user behaviour by effectively eliminating roles that are deemed irrelevant to deduce user activity. We are in the process of implementing the architecture and deploying this into a simulated environment to evaluate the overall design for our current work on activity recognition supporting health-care [4].

References

[1] Matthai Philipose, Kenneth P.Fishkin, Mike Perkowitz, Donald J. Patterson, Dieter Fox, Henry Kautz, and Dirk Hähnel, "Inferring Activities from Interactions with Objects", IEEE Pervasive Computing, vol. 03, no. 4, pp. 50-57, October-December, 2004, Available: http://seattleweb.intel-research.net/projects/activity/pics/philipose_pervasive_adl.pdf.

[2] Jakob Bardram and Henrik Bærbak Christensen, "Open Issues in Activity-Based and Task-Level Computing", First International Workshop on Computer Support for Human Tasks and Activities, CfPC PB-2004-60, 2004, Available: www.pervasive.dk/publications/files/open_issues_bardram.pdf.

[3] Valerie Guralnik and Karen Zita Haigh, "Learning Models of Human Behaviour with Sequential Patterns", Proceedings of the AAAI-02 workshop 'Automation as Caregiver', pp. 24-30, 2002, Available: http://www.cs.cmu.edu/~khaigh/papers/AAAIWS02-VGuralnik.pdf.

[4] Venet Osmani and Sasitharan Balasubramaniam, "Context Management Support for Activity Recognition in Health-Care", Pervasive 2006 Workshop Proceedings (ISBN 978-3-00-018411-6), pp.453-465 T. Strang, V. Cahill, & A. Quigley, eds in 3rd International Workshop on Tangible Space Intiative, 2006, Available: www.tsi2006.org.

[5] Anind K. Dey, "Providing Architectural Support for Building Context-Aware Applications", PhD Thesis, Georgia Institute of Technology, 2000, Available: http://www.cc.gatech.edu/fce/ctk/pubs/dey-thesis.pdf.

[6] Harry Chen, Tim Finin, and Anupam Joshi, "An Intelligent Broker for Context-Aware Systems", Adjunct Proceedings of Ubicomp 2003, Seattle, Washington, USA, 2003, Available: http://www.csee.umbc.edu/%7Ehchen4/paper/hc-ubicomp03-poster.pdf.

[7] Tao Gu, Xiao Hang Wang, Hung Keng Pung, and Da Qing Zhang, "An Ontology-based Context Model in Intelligent Environments", In Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference, pp. 270-275, 2004, Available: http://www.comp.nus.edu.sg/~gutao/gutao_NUS/CNDS2004_gutao.PDF.

[8] Newmad Technologies, "Newmad Technologies AB", 2006, Available: http://www.newmad.se/picowebserver.html.

[9] T. Nakano and T. Suda, "Self-organizing network services with evolutionary adaptation", IEEE Trans Neural Netw. 2005 Sep;16(5):1269-78, 2005, Available: http://www.cecs.uci.edu/journal_articles/ieee_suda.pdf.