Complex Activity Recognition Using Context-Driven Activity Theory and Activity Signatures

SAGUNA, Monash University and Luleå University of Technology
ARKADY ZASLAVSKY, CSIRO and Luleå University of Technology
DIPANJAN CHAKRABORTY, IBM Research

In pervasive and ubiquitous computing systems, human activity recognition has immense potential in a large number of application domains. Current activity recognition techniques (i) do not handle variations in sequence, concurrency and interleaving of complex activities; (ii) do not incorporate context; and (iii) require large amounts of training data. There is a lack of a unifying theoretical framework which exploits both domain knowledge and data-driven observations to infer complex activities. In this article, we propose, develop and validate a novel Context-Driven Activity Theory (CDAT) for recognizing complex activities. We develop a mechanism using probabilistic and Markov chain analysis to discover complex activity signatures and generate complex activity definitions. We also develop a Complex Activity Recognition (CAR) algorithm. It achieves an overall accuracy of 95.73% using extensive experimentation with real-life test data. CDAT utilizes context and links complex activities to situations, which reduces inference time by 32.5% and also reduces training data by 66%.

Categories and Subject Descriptors: H.1.2 [**User/Machine Systems**]

General Terms: Algorithm, Design, Experimentation, Measurement, Performance

Additional Key Words and Phrases: Activity recognition, complex activity, context-driven activity theory, context-awareness, concurrent activities, interleaved activities, prototype, test bed, experimentation, evaluation

ACM Reference Format:
Saguna, Arkady Zaslavsky, and Dipanjan Chakraborty. 2013. Complex activity recognition using context-driven activity theory and activity signatures. ACM Trans. Comput.-Hum. Interact. 20, 6, Article 32 (December 2013), 34 pages.
DOI: http://dx.doi.org/10.1145/2490832

1. INTRODUCTION

In recent years, the wide scale availability of low-cost sensing and mobile technology has led to an increased interest in the field of human activity recognition. The recognition of human activities has vast potential in application areas such as healthcare, aged care, emergencies such as natural disasters, comfort applications in smart homes, and energy-efficient urban spaces [Tapia et al. 2004; Philipose et al. 2004; Choudhury et al. 2008]. It can benefit the field of personal informatics by helping humans to better understand and analyse their behavior [Li et al. 2012]. Activities of Daily Living (ADLs) such as house work, office work, cooking, eating, shopping, exercising and grooming are highly complex and humans tend to multitask these with ease. Each ADL has

Author's addresses: Saguna, Monash University, Australia and Luleå University of Technology, Luleå, Sweden; A. Zaslavsky, CSIRO, Acton, ACT, Australia and Luleå University of Technology, Luleå, Sweden; D. Chakraborty, IBM Research Lab, New Delhi, India.

DOI: http://dx.doi.org/10.1145/2490832

more than one subactivity. For example, eating breakfast may involve sitting on chair, picking a knife, picking a fork, and so on. We call these subactivities *atomic activities* and define them as those unit-level activities that cannot be broken down further given application semantics. For example, the placement of a certain sensor can only be used to infer certain types of gestures like movement of fingers or a hand [Tu et al. 2012] or atomic activities like sitting or walking [Bao and Intille 2004]. This can also be dependent on the type of sensing technology used. These atomic activities belonging to different ADLs interleave or occur concurrently and are performed as operations that are routinized [Kaptelinin et al. 1999]. Recognition of such complex activities can help build smarter environments where, for example, a larger section of the ageing population that prefers to live independently in their own homes can be assisted.

Sensing technology and interaction of humans with computers is fundamental to the many areas of human computer interaction such as pervasive, ubiquitous, and distributed computing [Zhai and Bellotti 2005]. Human activity recognition using sensors in pervasive, distributed, and ubiquitous computing systems requires access to sensors, computational resources, and reasoning mechanisms to infer the activities [Abowd and Mynatt 2000]. Human users in their daily lives perform a large number of *complex activities,* which consist of a number of atomic (unit level) activities. Complex activities can be concurrent and interleaved and can have variations in sequence. Let us consider an example where a user John performs a complex activity such as " cooking an omelet," which involves a number of atomic activities such as "walking into kitchen," "standing close to the stove," " picking a pan," " turning the stove on," "opening the fridge," "taking the eggs out," "taking the bread out," "whisking the eggs," "frying the eggs," "chopping the vegetables," and "adding the vegetables." John can perform these activities in different ways each time, or he can perform it in an interleaved or concurrent manner with another complex activity such as "preparing tea." These atomic activities can also differ depending on "who is cooking the omelet" and "what type of omelet John is cooking." Similarly, John can perform a number of complex activities at home, in the office, and outdoors.

Different atomic activities can be categorized with respect to the sensing technology that is used to infer complex activities. For example, body motion activities can be inferred using accelerometer or gyroscope sensors worn by John on his body. An accelerometer worn on the arm would imply arm movement. An accelerometer worn on the waist or upper leg can be used to infer overall body motion such as walking, running, or sitting. Object interaction such as picking objects and touching objects can be inferred using RFID sensing technology in John's environment. Other modes of sensing computer or mobile phone related activities can be used based on virtual sensors present on John's devices. Virtual sensors represent software that is used to collect information about any activity while using the device (e.g., using a document writing software on a laptop). Similarly, a wide range of other types of atomic activities can also be recognized that involve other sensing technologies both wearable and those present in the user's environment.

Various techniques have successfully been applied to infer human activities. These include machine learning techniques [Tapia et al. 2004; Kaptelinin et al. 1999] and data mining techniques combined with use of semantics, string matching algorithms and human activity language development [Yang et al. 2006]. The most widely studied approaches are mainly data driven and use machine learning techniques for classification of complex activities after gathering and labelling large amounts of training data based on a specific sensor or set of sensors. We identify the key problems with existing techniques and present them here. The problem with existing techniques is that they work well within a particular domain and situation in which they are initially set in. For example, once removed out of the current domain into a new environment and infrastructure, the activity recognition techniques need to be trained again using new data [Kasteren et al. 2010; Rashidi et al. 2011]. These techniques suffer from problems such as large amounts of data collection and annotation for training of models [Tapia et al. 2004; Albinali et al. 2007; Davies et al. 2008]. Further, the changes in sequence of ADLs affect activity recognition accuracy of trained models adversely. Concurrent and interleaved ADLs are not accurately recognized, as it is time-consuming to train activity models for the different possible ways in which complex activities are interleaved or performed concurrently [Tapia et al. 2004; Albinali et al. 2007; Davies et al. 2008].

The field of activity recognition has a number of research challenges that need to be addressed by the research community. We consider the following challenges in this article:

—As mentioned previously, complex activities can have a different sequence each time they are performed. They can also be interleaved with other complex activities or more than one activity can be performed concurrently. It is required that variations in sequence, interleaving, and concurrency of complex activities should be handled during activity recognition [Davies et al. 2008].
—The use of context (e.g., location of user, velocity of user, light on/off and environmental temperature) in activity recognition can facilitate and improve the recognition of different activities [Abowd and Mynatt 2000; Choudhury et al. 2008; Ye et al. 2009]. This can be in the form of contextual cues or situations inferred from available context. *Context* is defined as any information which helps to describe a situation or scenario of a person or object [Abowd and Mynatt 2000; Dey 2000].
—Further, there arises a need to assimilate these atomic activities and context which are inferred from different sensing technology to recognize complex activities performed by the user. In doing so, we need to minimize the amount of training data required as well as the process of its annotation.
—A unifying theoretical framework to describe and define the information pertaining to complex activities based on both domain knowledge and experimentation is required. The definitions should include the associations between atomic activities, context, and complex activities. The situations associated with a complex activity and the signatures of each complex activity should also be present in the definitions to facilitate the activity recognition process.

In this article, we propose, develop and validate our Context-Driven Activity Theory (CDAT) based on extensive real-life experimentation. CDAT includes atomic and complex activity definitions along with other activity related context essential for complex activity recognition. Our approach towards complex activity recognition is shown in Figure 1. We claim the following contributions:

(1) We use our novel CDAT [Saguna et al. 2011] to build complex activity definitions and develop a mechanism which combines domain knowledge and activity data collected from real-life experimentation;
   (a) We discover complex activity signatures for different users using Markov chains as well as associations between atomic activities, context, and complex activities using probabilistic analysis, and
   (b) We use the updated CDAT definitions, activity signatures, and recomputed weights to infer complex activities using our complex activity recognition algorithm. We extend our Situation And Context-Aware Activity Recognition (SACAAR) system [Saguna et al. 2011] to recognize complex activities that are concurrent and interleaved. We achieve an average accuracy of 95.73% for complex activity recognition while maintaining computational efficiency, and it validates our approach.

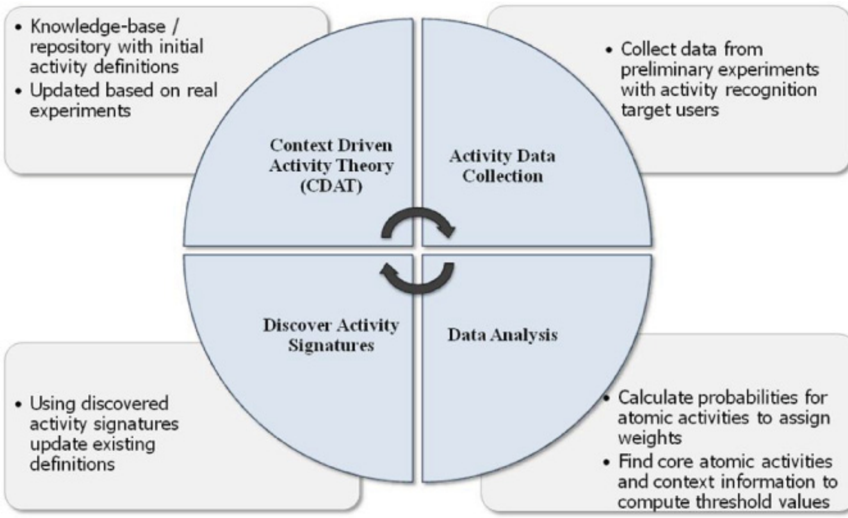

Fig. 1. CDAT based on domain knowledge and activity data.

(2) Our system reduces inference time by 32.5% and the amount of training data by 66%.

This article is organized as follows: In Section 2, we present the related work. Section 3 presents our proposed context-driven activity theory and the details relating to situation-aware activity recognition. Section 4 presents the details of the probabilistic analysis for weights assignment and the use of Markov chains to discover activity signatures. Section 5 presents the architecture of our approach and the complex activity recognition algorithm. Section 6 gives the test bed and prototype implementation. Section 7 presents the experimentation and results validation. Section 6 presents discussion, comparison with related work, and limitations, followed by a conclusion and discussion of future work in Section 9.

2. RELATED WORK

Activity recognition research includes the use of a variety of sensing technology and can be applied to different application settings. Sensors are broadly categorized based on deployment as wearable or on-body, environmental, and mobile sensors [Yang et al. 2006]. Wearable sensors include inertial sensors like accelerometers and gyroscopes, magnetic sensors, microphones, physiological sensors like electrocardiography (ECG) and electrooculography (EOG). These help in detecting a user's posture, motion, and other activities based on sound and physical states. Environmental sensors are those that are deployed in the user's environment and provide information about the user's surroundings or movements. These include video cameras, infrared sensors, reed switches, magnetic switches, force-sensitive resistors, and RFID tags that detect users' mobility in the area covered by them. There is a wide array of research that uses one or more of these sensors to detect human activities. Research in activity recognition is focused on a number of settings like industry and workshop [Lukowicz et al. 2004; Stiefmeier 2008], defense [Jovanov et al. 2003], health care and aged care [Tentori and Favela 2008; Homed et al. 2008; Amft and Troster 2008], comfort applications in smart homes [Meyer and Rakotonirainy 2003; Kasteren et al. 2008], emergency or natural disaster situations [Storf et al. 2009], and energy efficiency in urban spaces [GreenerBuildings 2010].

Researchers have applied both data-driven and knowledge-driven techniques to recognize activities of human users. A number of techniques such as Bayesian networks [Albinali et al. 2007], hidden Markov models (HMMs) [Deng and Tsui 2000; Rashidi et al. 2011], artificial neural networks [Van Laerhoven and Cakmakci 2000; Van Laerhoven 2001], context-free grammars [Ryoo and Aggarwal 2006], eigenspace algorithms [Huynh and Schiele 2006], string matching algorithms [Stiefmeier et al. 2007], conditional random fields [Liao et al. 2007], human activity language [Guerra-Filho and Aloimonos 2007], suffix trees [Hamid et al. 2007], minimum description length [Minnen et al. 2006], and ontological reasoning [Riboni and Bettini 2009; Chen et al. 2012] have been applied in the activity recognition field. We look at activity recognition as a hierarchical problem. In Bao [2003], an analogy of activity recognition with speech and text recognition is presented. Activities are recognized at word, sentence, and paragraph level keeping with the analogy of text or speech recognition [Bao 2003]. We refer to word-level (low-level) activities as atomic activities and paragraph-/sentence-level (high-level) as complex activities. As mentioned previously, atomic activities are unit-level activities that cannot be broken down further given application semantics (e.g., standing, sitting, walking, picking an object).

Existing research largely focuses on atomic activity recognition [Choudhury et al. 2008; Bao 2003] or complex activities constituting of simple sequential atomic activities [Tapia et al. 2004; Philipose et al. 2004; Choudhury et al. 2008]. Most of these works utilize one or more sensing technology such as accelerometer, RFID reader and tags, microphone, and so on, and infer simple sequential activities. For example, accelerometer-based complex activity recognition is restricted in terms of how widely it can be deployed, but it is well suited for recognizing atomic activities. The number of body movements for a complex activity can vary widely, and thus it becomes difficult to model such movements to achieve high levels of accuracies, as the complexity of activities increases in terms of its constituent atomic activities. The same is true for taking two sensing technologies together, such as accelerometer and RFID or accelerometer and microphone. The features are hard to model and discriminate when attempting to recognize complex activities without separating atomic activity inference from complex activity inference [Lee and Cho 2011]. There is a need to build complex activity recognition techniques that are not completely data driven but also involve domain knowledge. In Table I, we compare our approach with existing research in terms of a number of criteria, such as types of sensors used, the types of activities recognized, and the accuracy that was achieved, and we discuss the pros and cons of each.

Probabilistic models such as naive Bayes, decision trees, HMM, and dynamic Bayesian networks (DBN) are used for activity recognition in specific areas for complex activities. These models work well in a workshop or industry setting [Stiefmeier 2008; Ward et al. 2006; Stiefmeier et al. 2008] where activities are performed in the same sequence each time, but accuracy levels fall considerably when variations occur or activities are interleaved or concurrent in nature [Tapia et al. 2004; Davies et al. 2008]. Data mining techniques also show promise for complex activity recognition in daily life [Tao et al. 2009] with recognition of concurrent and interleaved activities, but there are issues that need to be addressed relating to window sizes and temporalities or variation in durations of activities that adversely affect the system. Conditional random fields used in Wu et al. [2007] and Hu and Yang [2008] also lead to interesting results but are susceptible to inaccuracies with complex activity recognition in daily life, as they are unable to handle changes in activities. They require the interleaving or concurrent activity data to be available during the training of the model. This leads to large amount of training data, as ADLs are interleaved and concurrently performed in a large variety of ways.

Temporal analysis is used in Brdiczka et al. [2010] for recognizing recurrent tasks. The T-pattern analysis uses temporal distances between events to infer high level

Table I. Taxonomy of Activity Recognition Approaches

Research Project	Sensor(s) Used	Activities Recognized (Atomic/ Complex)	Con-current/ Interleaved Activities	Inferencing Technique(s)	Accuracy Achieved	Use of Other Context	Other Pros/Cons
[Oliver et al. 2002; Brdiczka et al. 2007]	Microphone, video and virtual sensors for device activity	Atomic activities only	No	Layered HMM and Situation inference using Markov Models	99.7% and 88.58%	No	Limited to confined smart home environments and subject to privacy concerns
Mobile Sensing Platform (MSP) used for UbiFit Garden and UbiGreen [Consolvo et al. 2008; Choudhury et al. 2008]	Single wearable sensing device built with a number of sensors	Yes, but basic word level activities, like running, walking, watching TV, mainly focused on encouraging physical activity and saving on fossil fuel like in cars	No	Machine learning algorithms	93.8% for v2 supervised and 87.4% for v3 with 40% training data labeled	No	Non obtrusive small, single wearable device but can infer only limited number of activities
[Wang et al. 2007; Stikic et al. 2008]	Mainly RFID in both with the addition of accelerometer	Mostly atomic activities with couple of complex activities	No	In [Wang et al. 2007], Layered Model using DBN, HMM and Virtual Boosting and in [Stikic et al. 2008] Naive Bayes, HMM and Joint Boosting	Recall values between 85% and 90% for [Wang et al. 2007] and recall values lie between 72% and 64% for [Stikic et al. 2008]	No	Effective for recognizing atomic activities in ADLs but only use RFID-tagged objects

Research Project	Sensor(s) Used	Activities Recognized (Atomic/ Complex)	Con-current/ Interleaved Activities	Inferencing Technique(s)	Accuracy Achieved	Use of Other Context	Other Pros/Cons
[Riboni and Bettini 2009, 2011]	Sunspot based sensors	Results provided for 10 atomic activities	No	Statistical machine learning tools combined with ontological reasoning	89.20% and 93.44%	Yes, but only location informa-tion	Infers only preliminary atomic activities but combines ontology with statistical reasoning. The ontology presented can provide excellent modelling of context information though it is not used extensively in this work.
[Chen et al. 2012]	Object interaction sensors (tilt, pressure, and contact sensors)	8 complex activities of 3 types	No	Ontological constructs and reasoning	94.44% for coarse-grained activity recognition	No	Knowledge-based approach to complex activity recognition where all inferred activity specifications are present in the ontological constructs. It does not combine data-driven and knowledge-based approaches.
[Tao et al. 2009]	accelero-meter, temperature, humidity, light, and RFID sensors	17 atomic activities and 8 complex activities	Yes	Data mining specifically finding emerging patterns using frequent item set mining	90.96% (sequential), 87.89% (interleaved) and 78.58% (concurrent)	No	Achieves good accuracies but has issues with temporal window lengths, which affects accuracy adversely

Continued

Table I. Continued

Research Project	Sensor(s) Used	Activities Recognized (Atomic/ Complex)	Con-current/ Interleaved Activities	Inferencing Technique(s)	Accuracy Achieved	Use of Other Context	Other Pros/Cons
[Rashidi et al. 2011]	Smart home environment with a multitude of sensors such as motion, temperature, and contact sensors	8 complex activities, such as prepare meal, converse on phone, watch DVD, water plants, dust and sweep, select an outfit, and write a birthday card, with fixed description of steps	Yes	Data mining method called discontinuous varied order mining with voted HMM and clustering	73.8% to 77.3% with clustering	No	Requires data to create clusters of activities with human input to identify the clusters, also with multiple users in the smart home it will be difficult to distinguish between data of different users
[Helaoui et al. 2011a, 2011b]	RFID only	2 atomic activities and 9 complex activities	Yes	Markov logic networks	93% to 95%	No	The technique is only tested using RFID-based sensor data alone and the hard and soft formulae are preliminary, though they are expressive and flexible in terms of incorporating background knowledge. The system performs well with interleaved activities, but there is no use of context demonstrated in the experiments.

PC-based tasks. Temporal models such as Brdiczka et al. [2010] require collection and labeling of large amounts of training data and do not incorporate the use of background knowledge. The technique does not consider the use of context, and Brdiczka et al. highlight that accuracies of the T-pattern algorithm can be improved with less fine-grained distinction of tasks when applied to recognize the PC-based tasks. Such a temporal technique can be incorporated within our proposed approach of discovering complex activity signatures. Chen et al. [2012] use ontological constructs and reasoning to recognize complex activities but do not attempt to recognize concurrent and interleaved activities. A sliding-time window technique is also used for real-time continuous recognition of activities, but it lacks combining data-driven approaches with domain knowledge.

The notion of activity and context used in this article originate from the overall field of computer science and in particular from the field of ubiquitous computing and human-computer interaction. But the theoretical framing of complex activities is also relevant in the field of psychology and artificial intelligence [Abelson 1981; Schank and Abelson 1975] where scripts, events, and actions are used to describe everyday situations. These much earlier works focus on the much higher levels of activities that are related to the cognitive descriptions of different situations in everyday life, while the work in this article is more focused on using sensing technology and inferring the complex activities performed by users. As mentioned previously, we divide the problem of activity recognition into low-level and high-level activity recognition. We focus on combining both knowledge- and data-driven techniques to recognize the daily activities of users. Existing approaches do not offer a generic theoretical framework to activity recognition that can be widely used across application domains where activity related context can be shared easily. There is a need to have a standardized mechanism to define activities and types of sensors used for a particular activity. Such knowledge needs to be shared and reused so that there is efficient use of time and redundant tasks that have already been performed in one application domain are not repeated. Sensor deployments should be standardized so that systems can share sensor data. Recent works [Kasteren et al. 2010; Rashidi and Cook 2011] attempt to transfer activity knowledge from one house to another using data mining and HMM. They achieve reasonable accuracies but do not handle sequential, concurrent, and interleaved complex activities while performing transfer learning, which can be achieved using our approach. The works in Rashidi and Cook [2011] are only attempting to cluster sensors and activities without dealing with their complex nature from one domain to another. Thus, we argue for a theory for complex activity recognition that can represent the complex nature of complex activities. It is important that activity recognition be carried out at two levels: atomic and complex. Research has shown how various techniques can be deployed to infer atomic activities with ease. Variations that are inherent due to different humans performing the activities differently does not affect atomic activity recognition process but instead affects complex activity recognition process where the temporal lengths of these activities increases significantly. Algorithms built for activity recognition should be able to handle such issues. This can be better dealt by dividing the activity recognition problem into the atomic and complex activity levels.

There is a need to use context to infer complex activities, as this can help in achieving better accuracy [Choudhury et al. 2008; Hong et al. 2009; Saguna et al. 2011]. Knowledge-based approaches [Riboni and Bettini 2011; Chen et al. 2012] can be used to incorporate context within activity recognition, but current works have shown limited or no use of context. Hong et al. [2009] have used the Dempster-Shafer theory to incorporate contextual information from uncertain sensor data for activity recognition. In Helaoui et al. [2011b], Markov logic networks are used for complex activity recognition where hard and soft formulae are defined for recognizing activities. Such techniques can allow for expressive and flexible integration of background knowledge and context, but this has not been demonstrated. Our approach is fundamentally different from Hong et al. [2009] and Helaoui et al. [2011b], as we use a context-driven activity theory for recognizing complex activities and discover complex activity signatures. We incorporate situation recognition in the process of activity recognition. Ye et al. [2007, 2009] have used situation lattices to detect activities where they call " activities" as " situations" and have highlighted the importance of context and situations in activity recognition. Although their work has shown promising results for complex activities, they also state in Ye et al. [2009] that scalability can be an issue with situation lattices. We argue for the need to incorporate situations in our work based on Ye et al. [2007, 2009] and Padovitz et al. [2008] to detect complex activities. We differ from Ye et al. [2009], as we first detect spatio-temporal situations and then detect complex activities relating to those situations. In our previous work [Saguna et al. 2011], we performed complex activity recognition using our Situation- And Context-Aware Activity Recognition (SACAAR) system and Context-Driven Activity Theory (CDAT). In this article, we extend on CDAT and SACAAR by combining domain knowledge with experimental data to discover complex activity signatures. We show how detecting an occurring situation can facilitate the process of activity recognition. This can help to reduce the number of activities being detected. In the next section, we present our context-driven activity theory, its detailed formalisms, and our Situation- And Context-Aware Activity Recognition (SACAAR) system architecture with the complex activity recognition algorithm.

3. CONTEXT-DRIVEN ACTIVITY THEORY FOR COMPLEX ACTIVITY RECOGNITION

In this section, we propose and discuss a Context-Driven Activity Theory (CDAT), where activity could be complex or atomic. These activities can be linked to different situations. We employ the use of context at three levels within our CDAT. Context is used to infer atomic activities, complex activities, and situations. We will discuss this in detail in the following subsections. We further develop a context and atomic activity reasoning approach to infer and reason about complex activities.

3.1. Atomic Activity and Complex Activity Definitions

In this section, we give the definitions of the main components of our CDAT with some running examples.

Definition 3.1 *(Atomic activity).* Atomic activity A is defined as a unit-level activity which cannot be broken down further given application semantics and is observed from a set of sensors, $\Sigma S = s_1, \ldots, s_n$, where n ≥ 1, where the level of granularity can vary based on sensor placement scenarios.

For example, Case 1: A_1 = "*user is walking*" inferred by sensor s_1, which is an accelerometer placed on the user's waist. Case 2: A_1 = "*user is walking*" inferred by sensors s_1, s_2, s_3, and s_4, where s_1, s_2, and s_3 are three accelerometers placed on different parts of the body, and s_4 is a GPS sensor placed on the user. The accelerometers provide individual body part movements and additional context data originating from the GPS sensor provides velocity information, which can be together used to infer that user is walking.

Next, we give the definition of a context attribute [Padovitz et al. 2008], which can be used to infer complex activities from sensors.

Definition 3.2 *(Context attribute).* A context attribute is defined as any type of data at time t that is used to infer an activity or a situation(s). It is represented as C_i^t. Context attributes are related to physical sensors such as GPS to collect location coordinates or

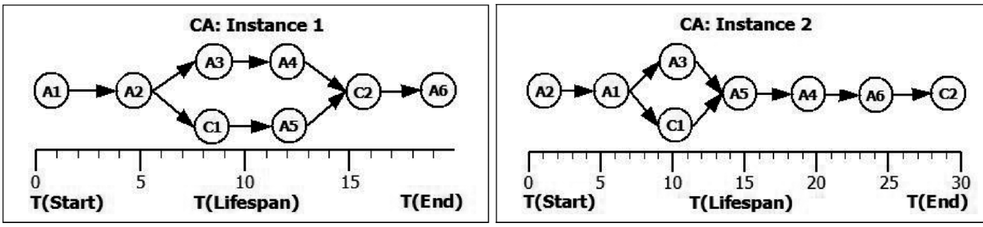

Fig. 2. An example complex activity which can be performed in two different ways: Instance 1 and Instance 2.

any other virtual or physical sensors. For example, location, time, and temperature are retrieved from sensors present on the user or his/her environment. Similarly, device activity and phone activity is retrieved from virtual sensors on the user's devices. For the previous examples, "*user is walking on a treadmill*" is inferred using additional context of detecting proximity to a treadmill or the location as gym. "*User is walking in the park*" can be inferred by user location as " in park." In the rest of the article, we refer to a context attribute simply as " context."

Definition 3.3 *(Complex activity).* A complex activity, CA, is a tuple CA = (γA, ρC, αA_S, αC_S, βA_E, βC_E, T_S, T_E, T_L), where γA is the subset of atomic activities from the complete set of atomic activities, $\Sigma A = \{A_1, \ldots, A_n\}$, n $\geq$ 1, and ρC is the subset of context from the complete set of context, $\Sigma C = \{C_1, \ldots, C_n\}$, n $\geq$ 1, which occur within the complex activity CA. Further, CoreγA and CoreρC are subsets of γA and ρC, which must be observed for a complex activity CA to occur. (αA_S, βA_E) $\subset \Sigma$A are atomic activities that are observed at the beginning or end of a complex activity, respectively. Similarly, (αC_S, βC_E) $\subset \Sigma$C are context and are observed at the beginning and end of a complex activity. Figure 2 shows two instances of the same complex activity as a state transition diagram. The different nodes represent context and atomic activities. A complex activity can vary each time a user performs it and can have a different starting and ending. T_S and T_E denote the starting time and ending time of a complex activity on a timeline. $T_L = |T_E - T_S|$ is the lifespan of a complex activity. As shown in Figure 2, a complex activity can be performed in different ways and can also have varying lifespans. This is depicted as $T_{Lmin} < T_L < T_{Lmax}$, where ($T_{Lmin}$, T_{Lmax}) gives the time range for the lifespan of a complex activity.

For example, Case 1: CA_1 = "*user is walking to bus stop*" is inferred by atomic activity, A_1 = "*user is walking,*" and context activity, C_2 = "*user direction towards bus stop from home*,"

Case 2: CA_2 = "*user is working on presentation at home*" is inferred by atomic activities A_2 = "*user is sitting*," A_3 = "*user is using study desk*," A_4 = "*user is detected near laptop*," and A_5 = "*user is typing in Microsoft Powerpoint application file*," and context activities C_3 = "*user location is home*" and C_4 = "*study desk light is on*." We further look at an example of concurrent and interleaved activities.

Case 3: A user performs the following activities concurrently and interleaved in time, which can be represented as ($CA_3|CA_4|CA_5|CA_6$) for complex activities, CA_3 = "*user is writing a document on his/her laptop in room at office*," CA_4 = "*user is browsing the Internet for research articles*," CA_5 = "*user is drinking coffee*," and CA_6 = "*user is chatting on IM with friend*." We can further perform operations on complex activities such as $\cup$ to create complex activities such as ($CA_3 \cup CA_4 \cup CA_5$).

3.2. Situation-Aware Activity Recognition

In Figure 3(a), we show how activities and situations are related in the activity analysis domain. First, context and sensors directly related to inferencing atomic activities

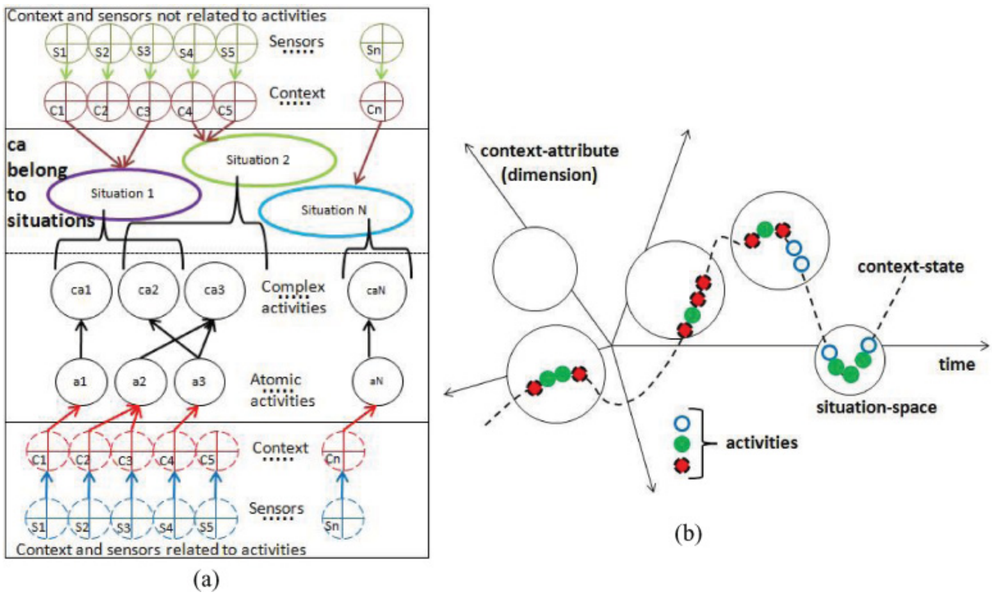

Fig. 3. (a): Sensors (s), context (c), atomic activities (a), complex activities (ca), situations and their relationships; (b): Situation subspaces and activities.

are used; for example, one or more accelerometers are used for atomic activities like sitting and walking. Also, certain context such as "speed of user" may further validate whether the user is walking. Further, a number of atomic activities together form a complex activity. *Situations* are "a set of circumstances in which an entity may find itself" where an entity can be a human or object. Human users in a day are part of a number of situations. All activities that are performed by them are part of these situations. We are referring to situations that are inferred by context, not sourced by activity-related sensor data but other context, for example, location, time, temperature, weather conditions, and light on/off. By inferring situations, we can limit the number of activities to be recognized. In a situation hierarchy, situations and subsituations exist, and it is possible to identify situations with a limited set of sensor and context. For example, a simple high-level situation is "user is in office room" detected by user's location sensor. This can help in activity inference because the user will perform only those activities that are linked to office room and not the kitchen. Thus, spatiotemporal situations by themselves can play a significant role in the activity recognition process.

Figure 3(b) shows how a trajectory can traverse through different situations and the activities that occur within them. The bubbles in the diagram represent different situation subspaces. The different axes represent context attributes (e.g., location, time, weather). The traversed path is representative of certain context states, which are the meeting points of these different context attributes. The small colored circles are merely activities that occur within situations on this path. We show how activities can occur in different situations and are linked to them. Thus, linking situations to activities is important in for activity recognition.

For the definition of a "Situation," we extend on the definition provided in the Context Spaces Theory [Padovitz et al. 2004, 2008].

Definition 3.4 *(Situation space).* A situation space represents a real-life situation. It is a tuple of regions of attribute values corresponding to a situation and denoted

Table II. Complex Activity Examples

CA_k ($\omega^T_{CA_k}$)	γA ($w^{A_i}_{CA_k}$)	ρC ($w^{C_i}_{CA_k}$)	Core γA and ρC	A_S, C_S	A_E, C_E	T_S, T_E, T_L (minutes)	$T_L$range (minutes)
Cooking omelette for breakfast in kitchen (0.59)	A_3: walking (0.10), A_2: standing (0.10), A_5: fridge (0.05), A_{18}: eggs (0.10), A_{21}: frypan (0.10), A_6: vegetable drawer (0.07), A_{17}: slicer (0.10), A_{11}: salt (0.10), A_{23}: whisker (0.08), A_{10}: knife (0.10), A_9: plate: (0.10)	C_2: kitchen (0.19), C_7: kitchen light on (0.12), C_{14}: stove on (0.19), $\neg C_{14}$: stove on (0.19), $\neg C_7$: kitchen light on (0.12), $\neg C_2$: kitchen (0.19)	A_3, A_2, A_{18}, A_{21}, A_{11}, A_{23}, A_9 and C_2, C_7, C_{14}	C_2, A_{18}, A_5, A_{21}, A_{23}	$\neg C_7$, A_9, $\neg C_{14}$	07:06, 07:22, 16	10-20
Preparing coffee in office kitchen (0.65)	A_3: walking (0.25), A_2: standing (0.25), A_{32}: coffee mug (0.25), A_{33}: coffee machine (0.25)	C_2: kitchen (0.3), C_7: kitchen light on (0.3), $\neg C_7$: kitchen light on (0.13), $\neg C_2$: kitchen (0.26)	A_3, A_2, A_{32}, A_{33}, C_2	A_{32}, C_2	$\neg A_{33}$, $\neg C_2$	18:16, 18:22, 6	5-10
Working on a doc in office room (0.90)	A_1: sitting (0.2), A_{47}: using office desk (0.2), A_{36}: using laptop/PC (0.2), A_{48}: typing or using document application (0.2)	C_{13}: in office room (1.0)	A_1, A_{47}, A_{36}, A_{48}, C_{13}	A_{48}	$\neg A_{48}$	10:10, 11:30, 80	10-60

by $S_j = (C^j_1, C^j_2, \ldots, C^j_n)$(consisting of n acceptable regions for these attributes). An acceptable region C^j_i is defined as a set of elements V that satisfies a predicate P, that is, $C^j_1 = V|P(V)$. Let the type of a region of acceptable values C^j_1 (i.e., the context attribute it is defined for) be represented with the symbol $[C^j_1]$, that is, $[C^j_1] = c_1$.

In Ye et al. [2007] and Padovitz et al. [2004, 2008], situations are reasoned from context and exist within a hierarchy. Ye et al. [2007] consider situation lattices for activity recognition. Padovitz et al. [2004, 2008] proposed the Context Spaces Model (CSM), where the concept of situation spaces and subspaces is used for reasoning real-life situations. We consider the CSM to represent and model context for inferring situations in the activity recognition domain. We consider those situations that are formed from context that is not sourced or related to the user's activity inference process. For this research, we have primarily focused on spatiotemporal situations, for example, "*user in home at morning*," "*user in home at night*," "*user in office at morning*," and so on.

3.3. Context and Atomic Activity Reasoning to Infer Complex Activities

To infer complex activities, we propose reasoning about context and atomic activities. Each complex activity has a set of atomic activities, γA, and a set of context, ρC, as mentioned in Definition 2. Table II gives some examples to explain our reasoning approach. Each atomic activity, $A_{i,}$ and context attributes, C_i, is assigned a particular weight, $w^{A_i}_{CA_k}$ and $w^{C_i}_{CA_k}$, respectively, corresponding to its importance in relation to the occurrence of a complex activity CA_k. We assigned importance based on expert opinion using a cookbook for the cooking activities. If expert knowledge is not available, it is possible to assign equal weights initially. For our experiments, we used equal weights for some complex activities. The sum of all the weights ω_{CA_k} for each CA_k is 1. If A_i or C_i do not occur for a particular CA_k, then $w^{A_i}_{CA_k} = 0$ and $w^{C_i}_{CA_k} = 0$. The sum of all the weights ω_{CA_k} for all occurring atomic activities and context needs to be above a threshold $\omega^T_{CA_k}$ in order for CA_k to occur successfully. If the sum of weights ω_{CA_k} is less than the threshold $\omega^T_{CA_k}$, then (1) it implies that the activity was started but abandoned in between, and (2) it implies that the core set of atomic activities and context for that particular CA_k did not occur. Thus,

$$\omega_{CA_k} = \frac{\sum_{i=1}^{N} w^{A_i}_{CA_k} + \sum_{i=1}^{N} w^{C_i}_{CA_k}}{2} \tag{1}$$

where, $0 \leq \omega_{CA_k} \leq 1$,
and if $w^{A_i}_{CA_k} = 0$ and $w^{C_i}_{CA_k} = 0$, if A_i and C_i do not occur for CA_k and

$$\omega_{CA_k} \geq \omega^T_{CA_k} \tag{2}$$

for any CA_k to have occurred successfully. We further demonstrate the use of weights, which helps in complex activity recognition by checking the occurrence of key atomic activities using the following examples from Table II.

Example 1: CA_1 = "*Cooking omelette for breakfast in kitchen*" is a complex activity, as shown in Table II. We use our CDAT along with our reasoning approach to define and infer CA_1. We define $CA_1 = (\gamma\text{A}, \rho\text{C}, \alpha A_S, \alpha C_S, \beta A_E, \beta C_E, T_S, T_E, T_L)$ in a straightforward way by using the type of atomic activities and context attributes available. The assignment of weights is based on the importance of each A_i and C_i for a corresponding CA_1. Initially, we assigned equal weights for A_i in CA_1. However, weights shown in Table II are the recomputed weights after probability analysis (explained further in Section 7.2). A number of A_i are assigned the highest and equal weight based on their occurrence. These are A_3, A_2, A_{18}, A_{21}, A_{17}, A_{11}, A_{10}, and A_9. These are followed by other constituent A_i for CA_1 having lower weights. Similarly, for the context attributes, C_i in CA_k, Table II shows the recomputed weights. In this case, ω_{CA_k} should be greater than the threshold $\omega^T_{CA_k} = 0.59$ for CA_1 to have successfully occurred. Also, temporal context is part of the complete set of context, ΣC. $\neg$A and $\neg$C represents nonoccurrence of atomic activity and context.

3.3.1. Complex Activity Weight Thresholds and Handling False Positives. Initially, as shown in Table II, each complex activity has a core set of atomic activities CoreγA and a core set of context attributes CoreρC. The sum of these core attributes is taken as the initial threshold values. CoreγA and CoreρC are used in handling false positives as well. If the sum of atomic activities and context attributes exceeds the threshold for a particular complex activity such as CA_1 even when CA_1 has not been completed or performed, it is possible to detect this false positive by checking if each element within CoreγA and CoreρC has occurred. If one or more elements from this core set are missing, it can be observed that CA_1 has not occurred yet.

Similarly, for complex activities CA_2 = "*Drinking coffee in lounge*" and CA_3 = "*Working on a document in office*" can be reasoned and inferred, as shown in Table II. We highlight that uncertain context or sensor information can also be dealt with by taking the probabilities of occurrence of atomic activities and multiplying with their assigned weights in Equation (1).

4. DISCOVERING ACTIVITY SIGNATURES AND GENERATING ACTIVITY DEFINITIONS USING PROBABILISTIC ANALYSIS

Complex activity definitions are created by finding the associations between each atomic activity and its occurrence within the occurrence of its parent complex activity between pairs of atomic activities belonging to the same complex activity as well as those belonging to two different complex activities occurring together. The data collected from users using the preliminary domain knowledge based on our CDAT approach is analyzed further for creating activity definitions.

4.1. Associations between Atomic and Complex Activities for Different Users

Associations between atomic and complex activities involves the calculation of individual probabilities of start, end, and other atomic activities for a complex activity. We determine the individual probabilities of each atomic activity, $A_i \in \gamma\ A_K$, where $\gamma\ A_K$ is the set of atomic activities for complex activity CA_K. The same procedure is also used to compute the individual probabilities for the start and end atomic activities (αA_S, βA_E) $\in \Sigma$A for complex activity CA_K. As mentioned previously in Section 3.1, we also use context as part of our CDAT. Thus, we determine the individual probabilities of each context attribute, $C_i \in \rho\ C_K$, where $\rho\ C_K$ is the set of context attributes for complex activity CA_K along with the individual probabilities for the start and end context attributes (αC_S, βC_E) $\in \Sigma$C for complex activity CA_K.

Thus, for example, for our complex activity CA_1 = "*cooking omelette*" from Table II, we calculate for each A_S and A_E for a complex activity CA: Pr(A_i, t) performed by a user n times as:

—Probability of A_S, $\Pr(A_i)\ \forall A_S$ in $\alpha A_S = \frac{total\ occurrence\ of\ A_i\ as\ A_S}{n}$
—Probability of A_E, $\Pr(A_i)\ \forall A_E$ in $\beta A_E. = \frac{total\ occurrence\ of\ A_i\ as\ A_E}{n}$

For example, taking CA_1 = "*cooking omelette*," we have A_{18}, A_5, A_{21}, A_{23} as the initial start atomic activities and A_4 as the initial end atomic activity in the CDAT. We calculate the probabilities as follows for the start and end atomic activities:

—Probability of A_S, $\Pr(A_{18}) = 0.60$
—Probability of A_S, $\Pr(A_5) = 0.05$
—Probability of A_S, $\Pr(A_{21}) = 0.30$
—Probability of A_S, $\Pr(A_{23}) = 0.05$

Similarly, for CA_1, we have:

—Probability of A_E, $\Pr(A_9) = 1.00$

We then find the A_S and A_E as follows:

—$A_S = \max \Pr(A_i)$, where $A_i \in \gamma A_K$, and $A_E = \max \Pr(A_i)$, where $A_i \in \gamma A_K$, which for CA_1 is A_{18} and A_9.

Next, we calculate the probabilities for the C_S and C_E for CA_1 as CA: P(C_i, t), performed by a user n times as:

—Probability of C_S, $\Pr(C_i)$ for all start $C_S = \frac{total\ occurrence\ of\ C_i\ as\ C_S}{n}$
—Probability of C_E, $\Pr(C_i)$ for all end $C_E = \frac{total\ occurrence\ of\ C_i\ as\ C_E}{n}$

We then have for CA_1 the C_S and C_E probabilities as:

—Probability of C_S, $\Pr(C_2) = 1.00$
—Probability of C_E, $\Pr(\neg C_7) = 0.30$
—Probability of C_E, $\Pr(\neg C_{14}) = 0.70$

We then find the C_S and C_E as follows:

—$C_S = \max \Pr(C_i)$, where $C_i \in \rho C_K$, and $C_E = \max \Pr(C_i)$, where $C_i \in \rho C_K$, which for CA_1 are C_2 and $\neg C_{14}$

The A_S and A_E as well as C_S and C_E can now be ranked from the most probable to the least probable for each complex activity.

Next, we take all the atomic activities and context attributes that lie between A_S, A_E and C_S, C_E. For example, in the case of complex activity CA_1 = "*cooking omelette*," we calculate for every atomic activity A_i and every context attribute C_i: $\Pr(A_i, \mathrm{t})$ and $\Pr(C_i, \mathrm{t})$ as:

—Probability of atomic activity, $\Pr(A_i) = \frac{total\ occurrence\ of\ A_i}{n}$, where n is the sum of occurrences of all atomic activities
—Probability of context attribute, $\Pr(C_i) = \frac{total\ occurrence\ of\ C_i}{n}$, where n is the sum of occurrences of all context attributes

The resulting values of probabilities for each atomic activity are checked against a threshold for relevance. The atomic activities whose values are equal to or higher than the required threshold are used for creating the activity definition for the respective complex activity. This results in removing those atomic activities that are of least importance to the occurrence of a complex activity, thus reducing the number of atomic activities to look for in the complex activity recognition process. It is also used to update the weights initially assigned to each atomic activity and context attribute.

4.2. Associations between Different Atomic Activities and Context Attributes within a Complex Activity

Associations between atomic activities involves the calculation of conditional probabilities and transition probabilities (p_{ij}) for different pairs of atomic activities within each complex activity. We used Markov chains for discovering these associations between pairs of atomic activities for a complex activity. For each atomic activity $A_i \in \gamma A_K$ where γA_K is the set of atomic activities for complex activity CA_K, we calculate for each pair of atomic activity (A_i, A_{i+1}) in complex activity, CA: $\mathrm{P}(A_i \mid A_{i+1}, \mathrm{t})$ (for dependency), for example, in the case of CA_1:

—$\Pr(A_1 \mid A_2) = \Pr(A_1 \text{ and } A_2) / \Pr(A_1) = 0.80$

We find all such conditional and transition probabilities. This is crucial in discovering the activity signatures of each complex activity for different users. Similarly, the associations between the context attributes of a complex activity are also computed using the conditional and transition probabilities. The details for discovering the activity signatures are given in the following section.

4.3. Discovering Complex Activity Signatures of Users

Based on the probability calculations in the Section 4.2, we build complex activity signatures for each complex activity corresponding to individual users. For example, the complex activity signature for CA = "" is $A_3 \to A_2 \to A_5 \to A_{18} \to A_3 \to A_2 \to A_{21} \to A_3 \to A_2 \to A_6 \to A_3 \to A_2 \to A_{17} \to A_{23} \to A_{23} \to A_{10} \to A_{21} \to A_9$. We use Markov chains to discover activity signatures by calculating the path probabilities for each complex activity. There can be different paths of atomic activities within a complex activity. A Markov chain takes a path in time through its state space. A sequence $\{x_1, x_2, \ldots x_t\}$ of states at successive times is referred to as a path [Ocone 2010; Meyn and Tweedie 1993; Pardoux 2008]. The probability $\Pr((X_1, X_2, \ldots X_t) = (x_1, x_2, \ldots x_t))$ of a path is the joint probability of $(X_1, X_2, \ldots X_t)$. But, as shown by other authors [Ocone

2010; Meyn and Tweedie 1993; Pardoux 2008], for a Markov chain $\{X_t\}_{t\geq 1}$ and for any path $\{x_1, x_2, \ldots x_t\}$,

$$Pr((X_1,\ X_2,\ \ldots X_t) = (x_1,\ x_2,\ \ldots x_t) \mid X_1 = x_1) = p_{x_1x_2}p_{x_2x_3}\cdots\ p_{x_{t-1}x_t}. \tag{3}$$

Therefore, it can be stated that the conditional probability of a path that is conditioned on the first value can be computed as the product of the transition probabilities (p_{ij}) between successive states of the path. Thus, the probability of a path, $\{x_1, x_2, \ldots x_t\}$ is

$$Pr((X_1,\ X_2,\ \ldots X_t) = (x_1,\ x_2,\ \ldots x_t)) = Pr(X_1 = x_1)p_{x_1x_2}p_{x_2x_3}\cdots\ p_{x_{t-1}x_t}. \tag{4}$$

The path with the highest probability is considered the most relevant complex activity signature for the user. This is used to detect complex activities of a user. The paths with lower probabilities can also be stored as they provide information about the user's activity variation in performing the same activity, and it can be associated to different contextual attributes with further study into the user's behavior and environment.

Similarly, the context attribute information is taken into consideration and the complex activity signature using context attributes are also discovered as, for example, in case of $CA_1 \Rightarrow \{C_3 \rightarrow C_4 \rightarrow \neg\, C_4 \rightarrow \neg\, C_3\}$.

4.4. Concurrent and Interleaved Complex Activities

For concurrent activities, the atomic activities that occur together in time are considered. The associations between atomic activities belonging to different complex activities are discovered via joint probability calculation of atomic activities. We calculate the joint probability for different pairs of atomic activities, $(A_i^{CA_1},\ A_j^{CA_2})$ in complex activities, CA_1 and CA_2: P($A_i^{CA_1}$, $A_j^{CA_2}$, t) when occurring together in time,

$$Pr\left(A_i^{CA_1},\ A_j^{CA_2},\ t\right) = Pr\left(A_i^{CA_1} \mid A_j^{CA_2}\right) * P\left(A_j^{CA_2}\right). \tag{5}$$

The use of conditional probability allows the system to, for example, know when CA_1 starts before CA_2 in time and then continue to occur simultaneously until they end and vice versa. If such information is not required by the system, then co-occurrence probability can be directly computed.

Similarly, for interleaving activities, associations between the atomic activities belonging to two different complex activities occurring consecutively at $t_i,\ t_{i+n}$ are discovered via joint probability calculation as shown in the following, where $A_i^{CA_1}$ and $A_j^{CA_2}$ are two atomic activities occurring at $t_i,\ t_{i+n}$, respectively:

$$Pr\left(A_i^{CA_1},\ A_j^{CA_2},\ (t_i,\ t_{i+n})\right) = Pr\left(A_i^{CA_1} \mid A_j^{CA_2}\right) * Pr\left(A_j^{CA_2}\right). \tag{6}$$

5. SACAAR ARCHITECTURE

The proposed SACAAR system architecture consists of three layers, as shown in Figure 4. SACAAR can be deployed in different application domains that require complex activity recognition. The sensory layer of the system can handle different types of sensor data, which is required to infer atomic activities, for example, accelerometers to infer body motion such as sitting, walking, standing, jogging; RFID readers and tags for object interaction such as picking a cup, picking a pan, touch sensors for opening a cupboard, opening a door; and physiological sensors for mood and stress levels. Virtual sensors can include modules of code that gather device activity, browser activity,

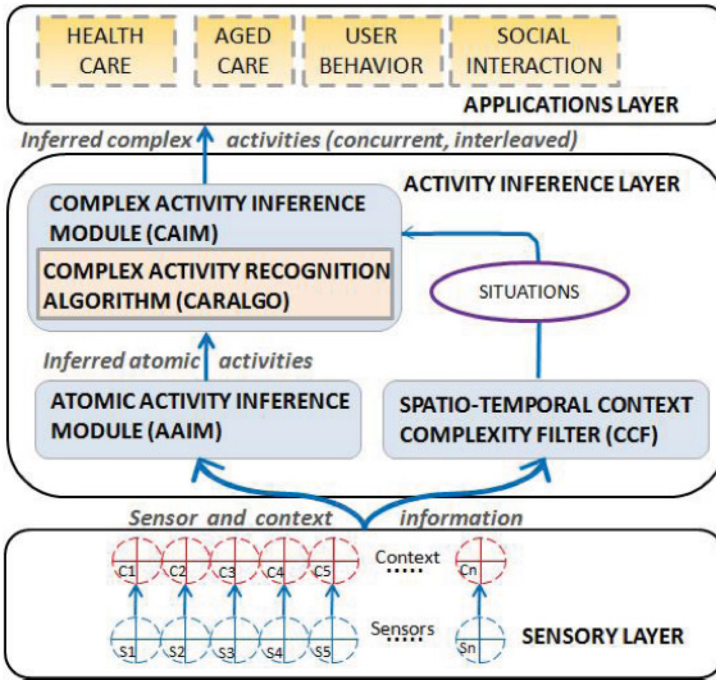

Fig. 4. SACAAR architecture.

and music player activity from a user's devices such as computers, laptops, and smart phones. We infer three types of context: First, context that helps in improving activity inference; for example, if sensor 1 (accelerometer) returns atomic activity walking, the confidence in this atomic activity can be increased by context from sensor 2 (GPS), which provides the speed of the user. Second, context that helps in inferring complex activities; for example, in the aforementioned case, sensor 2 (GPS) can also give the direction of the user between two points, A and B. This helps in inferring complex activity, such as the user walking toward point B. Last, context can be used to infer situations. The activity inference layer is used to infer activities and it consists of the atomic activity inference module, the complex activity inference module, and the spatiotemporal context complexity filter.

In the Atomic Activity Inference Module (AAIM), atomic activities are inferred. Body motion–related atomic activities, such as sitting, standing, and walking, are inferred using decision trees based on Bao [2003]. A weighted voting sliding window mechanism similar to Stikic et al. [2008] is used to infer RFID object interaction. A similar sliding window weighted voting mechanism is also applied to device and browser activity. Other context, such as speed and direction, are used to ascertain activities as well. Context from the sensory layer is used by spatiotemporal Context Complexity Filter (CCF) for inferencing situations, and the Complex Activity Inference Module (CAIM) is used to fuse together different sources of atomic activities and combine them to infer complex activities using the complex activity recognition algorithm (CARALGO).

5.1. Complex Activity Recognition Algorithm

The CAIM is used to fuse together different sources of atomic activities and combine them to infer complex activities with the help of the situation and activity linkages provided by the CCF. The complex activity recognition algorithm (CARALGO) is then used for inference of complex activities that can be concurrent and interleaved and is shown as Algorithm 1. CARALGO works by taking atomic activities, context, and situations as input and recognizes the complex activities. The algorithm finds the start atomic activity and then sets a time window of the size of the lifespan T_L for each matched $A_S \vee C_S$ belonging to a CA_k. CARALGO looks for matching γA, ρC and $A_E \vee C_E$ within the time window for each CA_k. It computes ω_{CA_k} using Equation (1) and then checks against $\omega^T_{CA_k}$, as shown in Equation (2). If the condition is matched, CA_k is inferred successfully. All time windows run in parallel and all incoming A_i and C_i for each CA_k are added to them until a successful match is found. The weights are added at runtime after each addition. The initial weights are assigned using domain knowledge. This is followed up by updating the weights once initial data is collected and analyzed and probabilities of all atomic activities are calculated. This removes discrepancies that may arise from domain knowledge-based weight assignment.

ALGORITHM 1: Complex Activity Recognition

Input: A_i, C_i, S_i.
Output: CA_k.
Initialization:
findStartAtomicActivity(A_i, C_i);
check for current situation S_i;
findComplexActivitiesList(S_i)
foreach (CA_k) **do**
    if $A_i == A_S$ **then**
        add($CA_{list} \longleftarrow CA_i = (\gamma A, \rho C, A_S, A_E, C_S, C_E, T_L)$)
    end
end
return CA_{list} ;
findComplexActivity(A_i, C_i)
foreach ($CA_{list} \longleftarrow CA_k$) **do**
    while $timecounter < T^{CA_k}_{L_{max}}$ **do**
        if ($A_i ==$ *element in* γA_i **then**
            add $A_i \longrightarrow \gamma A_i$ **and recalculate** $w^{A_i}_{\mathbf{CA}_k}$
        end
        if ($C_i ==$ *element in* ρC_i **then**
            add $C_i \longrightarrow \rho C_i$ **and recalculate** $w^{C_i}_{\mathbf{CA}_k}$
        end
    end
    if (($A_E, C_E\, foundforCA_i$) **and** ($\rho C_i and \gamma A_i$ **are complete and** $\omega_{\mathbf{CA}_k} \geq \omega^T_{\mathbf{CA}_k}$)) **then**
        $foundCA_k$
    end
    return CA_k ;
end

Context from the sensory layer is used by CCF for inferencing situations. This information is used in two ways by our SACAAR system architecture. First, context is used to infer situations. Activities always belong to some situation, and linking them can help in inferencing of activities as well as enhancing the richness of the inferred complex activity. These linkages can be created at runtime and new activities can be dynamically added to situations in SACAAR system. We initially created situations and link our predefined complex activities to these situations during the setup phase. We then provided a mechanism to add activities to new situations at runtime, if required.

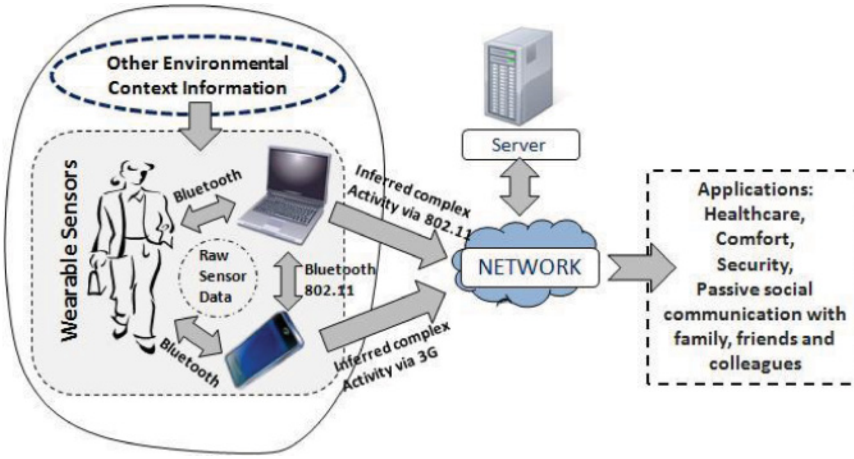

Fig. 5. SACAAR test bed and prototype.

Second, when the links between activities and situations are established, it can help to reduce the number of complex activities to look for during the inference process by inferring the situations first. Such situation inference can help in activity recognition by reducing battery consumption of certain sensors which are being polled to infer atomic activities which do not belong to the current situation. In the next section, we show how probabilistic analysis and Markov chains are applied to create CDAT definitions and create complex activity signatures.

6. TEST BED AND PROTOTYPE IMPLEMENTATION

To validate our proposed approach, we built a test bed and a prototype, as shown in Figure 5. Our test bed comprises of several sensors. For example, we place either Mulle v3 sensor [Mulle-v3 2010] (accelerometer) or the Android phone with its inbuilt accelerometer on the user's waist to detect body motion using decision trees. We chose the waist of the user as the most appropriate position [Bao 2003]. A Bluetooth RFID reader was used. It was placed on the user's wrist to detect LF passive RFID tagged objects within a distance of 1 to 5 cm. Each tag is labeled in terms of an activity, and the RFID readings are inferred as atomic activities. The device and browser activity are inferred using a C++ TimeTracker tool [Bennett 2007] available as open source on the Sourceforge website. We extended this tool to incorporate it in our test bed. The user's activity on the mobile phone is inferred using a Java-based code built by us specific to the Android platform [Android 2010]. Location information is collected using GPS, Wi-Fi positioning, and RFID tags. We use the inbuilt GPS on an Android phone to gather the location, speed and weather information. The prototype consists of 7,200 lines of Java and C++ code including different platforms and software languages taken together. The communication between different devices was done using Bluetooth, WLAN, and 3G technology. We also build a web-based infrastructure where, if required, atomic activity, context attribute, and inferred complex activity information can be viewed either by the user or by those involved in deploying activity recognition systems.

7. EXPERIMENTATION AND RESULTS VALIDATION

We performed extensive experimentation to validate our CDAT and SACAAR. In particular, this section presents results relating to (1) complex activity recognition with CDAT based on domain knowledge, (2) the activity data gathered with its detailed probabilistic analysis for weight computations and discovery of complex activity signatures, and (3) complex activity recognition using recomputed weights and complex activity signatures with our updated CDAT.

7.1. Experiment 1: Recognition of Activities Using CDAT

We use our SACAAR test bed (as shown in Figure 5) for testing and validating our proposed approach to recognize concurrent and interleaved complex activities. For results validation, we initially consider two subjects for the duration of 21 days, with an average of 8 hours daily. The experiments were performed from 8:00 am to 12:00 pm and from 2:30 pm to 9:30 pm. We identified 16 complex activities (listed in Table V in the appendix) and used our CDAT to define them. These definitions were stored in the SACAAR system. We gave our subjects an Android phone to record the activities manually, which involved adding a count for each occurrence of a complex activity in the corresponding hour. Users were asked to keep the record simply for establishing the ground truth, which enabled us to measure the accuracy of our algorithm. This information was not used for any annotation or training purpose within our CDAT and SACAAR. We show the accuracy of our algorithm for both subjects combined together in Figure 8 as "Before." Our algorithm performed with an overall accuracy of 88.5% using domain knowledge-based CDAT definitions. We also used Decision Trees (DT) (J48) and Naive Bayes (NB) to detect complex activities. The results presented in Figure 8 show that our algorithm significantly outperformed both techniques. DTs and NBs have low accuracies, as they are unable to handle the variations in sequence and the concurrent/interleaved nature of the complex activities.

7.2. Probabilistic Activity Data Analysis

We further analysed the data collected during the initial experimentation and used it to automatically compute the probabilities of atomic activities, context, transition probabilities for atomic activities and context and joint probabilities for concurrent and interleaved activities. The knowledge gained by the probabilistic analysis is used to update the definitions in CDAT that were previously set by domain knowledge. Using the updated definitions and recomputed weights, we performed another set of extensive experiments. The results from this experimentation are presented in Section 7.3.

7.2.1. Probability Calculations. The probabilities for atomic activities ($\Pr(A_i)$) and context attributes ($\Pr(C_i)$) for eight complex activities (CA_k) are shown in Figure 6.[1] The remaining eight complex activities are not shown, as their constituent atomic activities and context attributes had equal or close to equal probabilities. Figure 6 shows that for highly complex activities, there is greater variation in probabilities of constituent atomic activities and context attributes. For example, in "*Making Pizza,*" CA_3 complex activity, certain atomic activities that are basic to the making of a pizza have higher probabilities. Probabilities of other atomic activities are lower, as they are dependent on the user's choice of ingredients on different days and his/her dietary preferences. Similarly, for other kitchen activities, there exists greater variation in probabilities of constituent atomic activities and context attributes. Those activities that are fairly less complex, such as "*Preparing Coffee,*" CA_6, have constant probabilities of its constituents, as the scope for variation is less and all basic steps are always followed. In such cases, there may only be variation in the order of steps. Figure 7 shows the recomputed weights based on the probabilities shown in Figure 6. The recomputed weights follow similar distribution, as shown in Figure 6 after normalization, and achieve a sum of 1.

[1]Refer to tables V, VI and VII for atomic and complex activity as well as context attribute notations.

Fig. 6. Atomic activity and context attribute probabilities for eight complex activities from Table V.

7.2.2. Discovering Complex Activity Signatures. Table III shows the complex activity signatures. These are based on the transition probabilities between atomic activities for each complex activity discovered using Markov chains, as elicited in Section 4. The most traversed path is selected after calculating the transition matrices (not shown here due to lack of space) for each complex activity. Table III shows all the complex activity signatures discovered from the first round of experimentation for all 16 complex activities. We divided the signatures into two separate parts based on atomic activity transitions and context attribute transitions, as shown in columns 2 and 3 of Table III. The signatures shown in the table are those with

Fig. 7. Atomic activity and context attribute weights for eight complex activities from Table V. These share similar distribution, as shown in Figure 6. Similarly, weights were computed for all other complex activities.

the highest path probability, which was calculated using Equation (4). The path probabilities for both atomic activities and context attributes are shown in column 4. Most complex activities had more than one signature but only one significant signature. They were also stored for the inference process in further experimentation (Experiment 2). The most variation was observed in *"Getting Ready,"* CA_4; *"Searching the Internet,"* CA_{14}; *"Working on Document,"* CA_{13}; and *"Working on Presentation,"* CA_{12}. This was caused due to the high interleaving and concurrent nature of these activities with other complex activities as well as the high variation in which the

Table III. Complex Activity Signatures for User 1

Complex Activity CA_k	Complex Activity Signature with Atomic Activities γA_i	Complex Activity Signature with Context ρC_i	Path Probability $(\gamma A_i)\,(\rho C_i)$
Making Sandwich CA_1	$A_3 \to A_2 \to A_{20} \to A_3 \to A_2 \to A_5 \to A_{19} \to A_6 \to A_3 \to A_2 \to A_7 \to A_{17} \to A_{11} \to A_{12} \to A_9$	$C_2 \to C_7 \to \neg C_7 \to \neg C_2$	(0.85) (0.87)
Making Omelette CA_2	$A_3 \to A_2 \to A_5 \to A_{18} \to A_3 \to A_2 \to A_{21} \to A_3 \to A_2 \to A_6 \to A_3 \to A_2 \to A_{17} \to A_{23} \to A_{23} \to A_{10} \to A_{21} \to A_9$	$C_2 \to C_7 \to C_{14} \to \neg C_{14} \neg C_7 \to \neg C_2$	(0.81) (0.84)
Making Pizza CA_3	$A_3 \to A_2 \to A_8 \to A_3 \to A_2 \to A_{14} \to A_{15} \to A_3 \to A_2 \to A_5 \to A_6 \to A_3 \to A_2 \to A_7 \to A_{10} \to A_{22} \to A_3 \to A_2 \to A_5 \to A_3 \to A_2 \to A_{19} \to A_{13} \to A_{11} \to A_{12} \to A_9$	$C_2 \to C_7 \to C_{14} \to \neg C_{14} \neg C_7 \to \neg C_2$	(0.82) (0.95)
Getting Ready CA_4	$A_3 \to A_2 \to A_{25} \to A_{24} \to A_3 \to A_2 \to A_{26} \to A_3 \to A_2 \to A_{27} \to A_3 \to A_2 \to A_{28}$	$C_4 \to C_9 \to C_3 \to C_8 \to \neg C_8 \to C_4 \to \neg C_9 \to \neg C_4$	(0.67) (0.60)
Eating Breakfast CA_5	$A_1 \to A_{29} \to A_9 \to A_{30} \to A_{31} \to A_{32}$	$C_6 \to C_{11} \to \neg C_{11} \to \neg C_6$	(0.95) (0.98)
Preparing Coffee CA_6	$A_3 \to A_2 \to A_{32} \to A_{33} \to A_{32}$	$C_2 \to C_7 \to \neg C_7 \to \neg C_2$	(0.98) (1.0)
Drinking Coffee CA_7	$A_1 \to A_{32}$	C_{12}	(1.0) (1.0)
Watching Videos CA_8	$A_1 \to A_{36} \to A_{37} \to \neg A_{37}$	C_{13}	(0.95) (1.0)
Laundry CA_9	$A_{38} \to A_3 \to A_2 \to A_{26} \to A_3 \to A_2 \to A_{42} \to A_3 \to A_2 \to A_{27} \to A_3 \to A_2 \to A_{39}$	$C_4 \to C_9 \to C_3 \to C_8 \to \neg C_8 \to C_2 \to C_7 \to \neg C_7 \to C_4 \to \neg C_9$	(0.73) (0.68)
Cleaning Kitchen CA_{10}	$A_2 \to A_{40} \to A_{41} \to \neg A_{40}$	$C_2 \to C_7 \to \neg C_7 \to \neg C_2$	(0.75) (0.84)
Eating Dinner CA_{11}	$A_1 \to A_{29} \to A_9 \to A_{30} \to A_{43} \to A_{44} \to A_{45}$	$C_6 \to C_{11} \to \neg C_{11} \to \neg C_6$	(0.80) (0.85)
Working on Presentation CA_{12}	$A_1 \to A_{47} \to A_{36} \to A_{46} \to \neg A_{46}$	C_{13}	(0.67) (0.95)
Working on Document CA_{13}	$A_1 \to A_{47} \to A_{36} \to A_{48} \to \neg A_{48}$	C_{13}	(0.55) (0.98)
Searching the Internet CA_{14}	$A_1 \to A_{47} \to A_{36} \to A_{49} \to \neg A_{49}$	C_{13}	(0.57) (0.85)
Jogging in the Gym CA_{15}	$A_4 \to A_{50} \to \neg A_{50}$	C_{12}	(0.85) (0.95)
Going to Work CA_{16}	$A_{51} \to A_3$	$C_1 \to C_{19} \to C_{12}$	(1.0) (1.0)

concurrency and interleaving occurred. Our algorithm was easily able to recognize this behaviour.

7.2.3. Recognition of Concurrent and Interleaved Activities. The concurrent and interleaved nature of complex activities in our experimental data is shown in Figure 9 using a heat map. It shows the activity pairs which had maximum and minimum concurrency and interleaving. It can be observed that (CA_{14}, CA_{13}) and (CA_{14}, CA_{12}) have the highest values for concurrency and interleaving. Activities in the morning are often interleaved and performed concurrently. Also, activities such as "*Drinking coffee*," CA_7 and

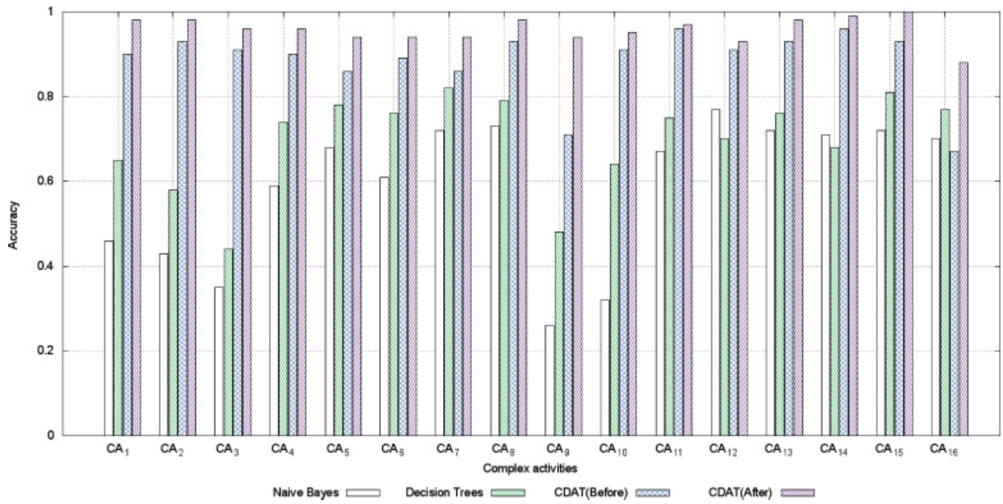

Fig. 8. Complex activity recognition accuracy: (1) CDAT from Section 6.1 as "Before," (2) updated CDAT from Section 6.3 as "After," (3) decision trees (J48), and (4) naive Bayes (NB).

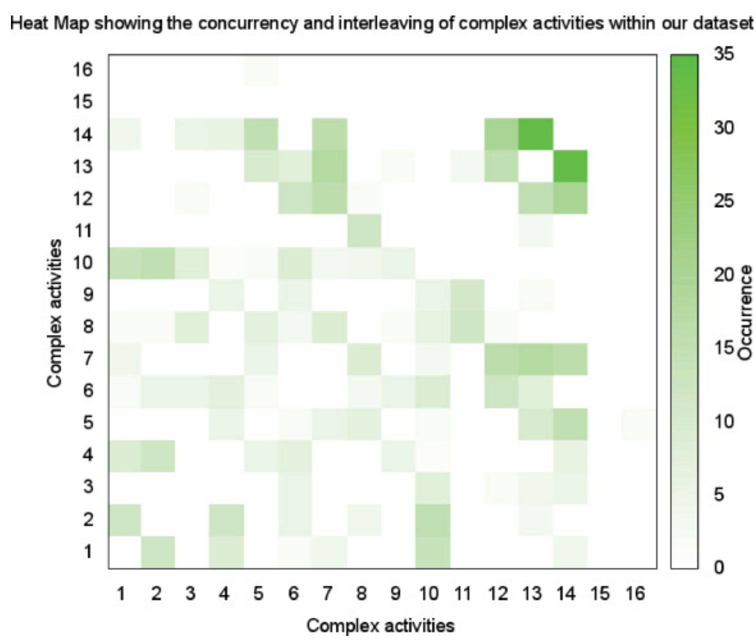

Fig. 9. Heat map showing concurrency and interleaving between different pairs of complex activities from Table V. The x- and y-axis depict the complex activities.

"*Watching videos*," CA_8 are interleaved with a number of other activities. We highlight that the interleaving of complex activities is observed from the interleaving and concurrency of context and atomic activities belonging to different complex activities. Concurrency of complex activities is mostly observed when complex activities occur within same time periods.

7.3. Experiment 2: Recognition of Activities based on Updated Definitions in CDAT

In the previous section, using probabilistic analysis, we showed how weights were recomputed and how CDAT definitions were updated with complex activity signatures. In this section, we discuss the results related to further experiments conducted based on the updated definitions in our CDAT. The users perform the same 16 complex activities for another 2 weeks. The results for these experiments are shown in Figure 8 as “After” (in comparison to Experiment 1 shown as “Before”). In Experiment 2, the complex activity recognition Algorithm 2 is used with the recomputed weights for complex activity inference in Lines 15 and 18. Also, in Line 21, the complex activity signature is matched for recognizing a complex activity. We find that with the updated definitions and recomputed weights SACAAR performed with an improved accuracy of 95.73%.

ALGORITHM 2: Complex Activity Recognition after Probabilistic Analysis and Discovered Complex Activity Signatures

Input: A_i, C_i, S_i.
Output: CA_k.
1 **Initialization:**
2 findStartAtomicActivity(A_i, C_i);
3 check for current situation S_i ;
4 findComplexActivitiesList(S_i)
5 **foreach** (CA_k) **do**
6   **if** $A_i == A_S$ **then**
7         $add(CA_{list} \longleftarrow CA_i = (\gamma A, \rho C, A_S, A_E, C_S, C_E, T_L))$
8   **end**
9 **end**
10 **return** CA_{list} ;
11 findComplexActivity(A_i, C_i)
12 **foreach** ($CA_{list} \longleftarrow CA_k$) **do**
13   **while** $timecounter < T^{CA_k}_{L_{max}}$ **do**
14   **if** ($A_i == element\ in\ \gamma A_i$ **then**
15   **add** $A_i \longrightarrow \gamma A_i$ **and** recalculate $w^{A_i}_{CA_k}$using recomputed weights
16   **end**
17   **if** ($C_i == element\ in\ \rho C_i$ **then**
18   **add** $C_i \longrightarrow \rho C_i$ **and** recalculate $w^{C_i}_{CA_k}$ using recomputed weights
19   **end**
20   **end**
21   **if** ((A_E, C_E *found for* CA_i) **and** ($\rho C_i and \gamma A_i$ *are complete* **and** $\omega_{CA_k} \geq \omega^T_{CA_k}$**and** *complex activity signature matched*)) **then**
22         $foundCA_k$
23   **end**
24   **return** CA_k ;
25 **end**

7.4. Reducing the Amount of Training Data, Atomic Activities, and Context Attributes Used

We are able to reduce the amount of data required for training activity models by 66% by using our CDAT and SACAAR system as compared to decision trees and naïve Bayes classifiers for the same complex activities. Training data was only used at the atomic

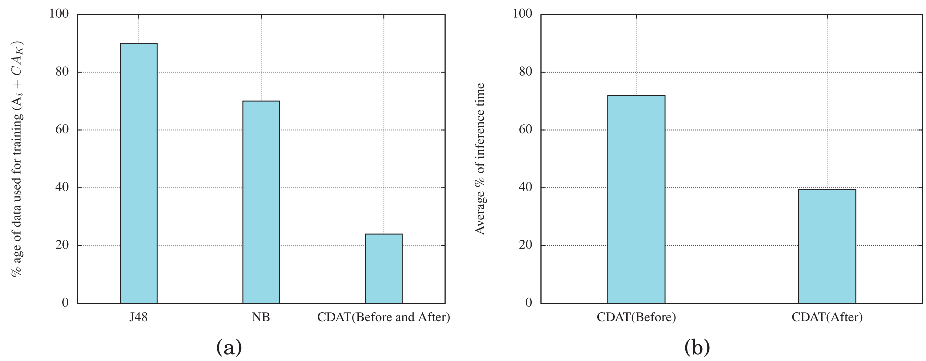

Fig. 10. (a): Training data used for atomic activities and complex activities for decision trees (J48), naive Bayes (NB), and CDAT (Before and After). (b): Average percentage of time for inferring complex activities for CDAT (Before) and CDAT (After).

activity level where we have used existing techniques for inferencing activities like body motion, object interaction, and so on. Training data was not required by our system at the complex activity inference level. Moreover, by updating the CDAT definitions using probabilistic analysis and discovering complex activity signatures, we were able to reduce the amount of sensor information required to infer complex activities. This was also achieved using the inference of situations, which eliminated the need to infer certain complex activities which do not belong to the observed situation. This in turn implied that complex activities can be inferred with lesser number of atomic activities and context attributes without adversely affecting recognition accuracy. Figure 10(a) shows the comparison for amount of data used in training for both atomic and complex activities taken together.

8. DISCUSSION, COMPARISON WITH RELATED WORK AND LESSONS LEARNED

From the extensive experiments detailed earlier, SACAAR has proved to be an effective mechanism to infer complex activities that are sequential or concurrent and interleaved. The activity definitions in CDAT used for preliminary inference using real-life data accompanied by probabilistic analysis and use of Markov chains to update CDAT definitions significantly improved complex activity recognition accuracy. Our approach is significantly different from existing techniques [Choudhury et al. 2008; Tao et al. 2009; Rashidi et al. 2011; Riboni and Bettini 2009; Wang et al. 2007; Stikic et al. 2008] used for complex activity recognition, as we are able to create activity signatures for complex activities for individual users as well as for different users taken together, which is then used for inferring complex activities. In Table IV, we compare our approach with those presented in Table I, using a number of criteria such as recognition accuracy, use of context, and type of activities inferred. The success of the SACAAR approach lies in the combined the use of CDAT, the probabilistic analysis of real-life data, and the use of Markov chains, which brings together domain knowledge as well as real-life experimentation-based knowledge. The levels of associations of different atomic activities and context attributes to complex activities are discovered along with the activity signatures. This enables us to assign weights according to these atomic activities and context attributes for each complex activity. This knowledge enables improved recognition accuracies as well as eliminates redundant sensor data and context and can deal with inaccuracies and uncertainty.

The knowledge engineering effort in creating definitions within CDAT is similar to ontological approaches such as those of Riboni and Bettini [2011] and Chen et al.

Table IV. Comparison of Our Approach with Related Work in Table I

Research Project	Sensor(s) used	Activities Recognized (Atomic/ Complex)	Con-current/ Interleaved Activities	Inferencing Technique(s)	Accuracy Achieved	Use of Other Context	Other Pros/Cons
SACAAR system (author's approach)	Multiple sensors such as accelerometer, RFID, GPS and Wi-Fi location, virtual sensors	16 complex activities	Yes	Two step approach: State-of-the-art techniques for atomic activity recognition and use of CDAT, CAR algorithm, probabilistic analysis, Markov chains and situation association for complex activity recognition	95.73%	Yes, extensive use of context	Our SACAAR system uses a variety of sensory input along with rich context to infer 16 complex activities and outperforms existing techniques

[2012]. These are initial overheads that can be considerably minimized after the creation of CDAT definitions for users. These definitions can later be shared across different domains, applications, and users to recognize ADLs. For example, a definition can be created for the complex activity "*cooking omelette*" for a particular user. This definition can be shared for other users who perform this activity in a similar manner. The basic building blocks for a complex activity in terms of CDAT definitions will be the same for most users. ADLs are complex in terms of how they are performed but are fairly straightforward in terms of their constituent elements such as atomic activities, context attributes, and the start and end activities. These complex definitions created for different users can be added and updated within our CDAT repository over time and reused for other new users. Generalization can be achieved by creating definitions where common and uncommon aspects can be considered for each definition and different users. Also, in the future, we can look at applying similarity measurement techniques to match definitions for different users. At the atomic activity level, there are already many successful applications which recognize user body motions like running, walking, standing as well as for object interaction. Some recent examples for body motion include coaching applications such as Nike+ and miCoach used for personal training and fitness which use GPS and accelerometer sensors present on the mobile phone.

In traditional activity recognition models, large amounts of training data is required to infer complex activities. By using our SACAAR approach, we considerably reduce the amounts of training data required (66% as compared to decision trees and naive Bayes) for complex activity recognition. The inference time was reduced by 32.5% when complex activities were recognized using the updated CDAT (Section 7.3), as shown in Figure 10(b). The percentage was calculated using the time taken to infer all complex activities together for each case "Before" and "After." The time taken for decision trees and naive Bayes is not shown, as they are not temporal models and the time for inference was the same as the time to perform a complex activity. Again, this is achieved by using a combination of initial activity definitions and probabilistic analysis of real-life experimentation data. This is also achieved by associating situation inference with complex activities. We differ from existing approaches on complex activity recognition, as we create links between activities and situations and use these associations in our activity recognition algorithm. The success of our method provides a useful mechanism based in theory and experimentation to enhance the understanding, development, and implementation of complex activity recognition systems. In particular, one useful finding is that our promising results provide strong evidence to show that complex activities have varying associations with different atomic activities and context attributes, which, when known to the activity recognition system, can lead to high accuracy of recognition. Our complex activity recognition algorithm performed with an average accuracy of 95.73%. We also demonstrate and validate this by the reduction in required training data as well as the inference time as previously mentioned.

8.1. Limitations

Complex activity recognition approaches in existing work mostly use machine learning techniques and the results are promising in certain scenarios. There are large amounts of training data required as well as a change in sensor deployments and settings. Activity models are created specifically for each complex activity and need to be retrained for changes in activity steps. Since our work builds on the existing techniques at atomic activity level the inherent weaknesses of the existing approaches can adversely affect the working of our models at the atomic activity level. Although, this can be addressed by carefully picking only successful and high accuracy achieving machine learning techniques for atomic activity inference. For example, decision trees are proven to give high accuracy results for simple body motion atomic activities. Since our system is dependent on CDAT, it requires the collection of sensory data in accordance with the definitions available in CDAT. It is important to identify the sensors and the associated atomic activities before data is collected and used for inference. This also makes it difficult to test our system on the publicly available activity datasets.

9. CONCLUSION AND FUTURE WORK

We propose, develop, and validate a novel Context Driven Activity Theory (CDAT) to recognize complex activities. We build a mechanism which utilizes our CDAT to recognize complex activities that are sequential, concurrent, and interleaved. We use probabilistic analysis and Markov chains to discover complex activity signatures, assign weights to atomic activities, and update complex activity definitions within our CDAT. Our CDAT-based complex activity recognition algorithm achieves high accuracy of 95.73% while recognizing complex activities that are concurrent and interleaved in nature. Moreover, the mechanism introduced in this article is original and unique, as it discovers complex activity signatures for users and finds associations between a complex activity and its constituent atomic activities and context attributes. The levels of association are then used to assign weights to atomic activities and context attributes which help in dealing with inaccurate and uncertain data. Further, situation inference is used within our system to detect complex activities, which enables faster activity recognition by identifying certain complex activities that are unrelated to the situation at hand. This reduces inference time by 32.5% and also reduces training data by 66%.

In the future, we will build an activity ontology to incorporate the associations discovered between different atomic activities, context attributes, complex activities, and situations using our CDAT. This will enable sharing of such activity semantics as well as make development and deployment of activity recognition systems easy. An activity ontology along with extensions to CDAT can help in providing cognitive descriptions of activities, for example, "*working on document*" can be described more appropriately as "*developing a proposal for funding next quarter*." We plan to find the most common and least common ways in which users perform various complex activities. This can help in creating a consolidated repository of complex activities and related activity semantics that can be shared with ease.

APPENDIX

COMPLEX ACTIVITIES, ATOMIC ACTIVITIES AND CONTEXT ATTRIBUTES USED IN OUR EXPERIMENTATION

The following three tables provide the list of complex activities, atomic activities, and the context used in our experimentation along with the labels associated with them. Table V gives the complex activities with the associated atomic activities and context attributes.

Table V. Complex Activities with Their Constituent Atomic Activities and Context Attributes

Complex Activity CA_k	Atomic Activities γA_i	Context Attributes ρC_i
Making Sandwich CA_1	$A_2, A_3, A_{20}, A_6, A_{10}, A_{17}, A_7, A_{19}, A_{11}, A_{12}, A_9$	C_2, C_7
Making Omelette CA_2	$A_2, A_3, A_5, A_9, A_{18}, A_{21}, A_6, A_{20}, A_{16}, A_{23}, A_{23}, A_{10}$	C_2, C_7, C_{14}
Making Pizza CA_3	$A_2, A_3, A_8, A_{15}, A_{14}, A_6, A_7, A_{19}, A_{13}, A_{11}, A_{12}, A_9, A_{10}, A_{22}$	C_2, C_7, C_{14}
Getting Ready CA_4	$A_2, A_3, A_{24}, A_{25}, A_{26}, A_{27}, A_{28}$	C_4, C_9, C_3, C_8
Eating Breakfast CA_5	$A_1, A_9, A_{29}, A_{30}, A_{31}, A_{32}$	C_6, C_{11}
Preparing Coffee CA_6	$A_2, A_{32}, A_{33}, A_{34}, A_{35}$	C_2, C_7
Drinking Coffee CA_7	A_1, A_2, A_3, A_{32}	C_{12}
Watching Videos CA_8	A_1, A_2, A_{36}, A_{37}	C_{13}
Laundry CA_9	$A_1, A_2, A_{38}, A_{39}, A_{26}, A_{27}, A_{42}$	$C_2, C_3, C_4, C_7, C_8, C_9$
Cleaning Kitchen CA_{10}	A_2, A_{40}, A_{41}	C_2, C_7
Eating Dinner CA_{11}	$A_1, A_9, A_{29}, A_{30}, A_{43}, A_{44}, A_{45}$	C_6, C_{11}
Working on Presentation CA_{12}	$A_1, A_{36}, A_{46}, A_{47}$	C_{13}
Working on Document CA_{13}	$A_1, A_{36}, A_{48}, A_{47}$	C_{13}
Searching the Internet CA_{14}	$A_1, A_{36}, A_{49}, A_{47}$	C_{13}
Jogging in the Gym CA_{15}	A_4, A_{50}	C_{12}
Going to Work CA_{16}	A_3, A_{51}	C_1, C_{12}, C_{19}

Table VI. List of Atomic Activities

Atomic Activity A_i	Description	Atomic Activity A_i	Description
A_1	Sitting	A_{27}	Shoe_Rack
A_2	Standing	A_{28}	Grooming_Equipment
A_3	Walking	A_{29}	Dining_Table
A_4	Running	A_{30}	Cuttlery
A_5	Fridge	A_{31}	Juice_Glass
A_6	Vegetable_Drawer	A_{32}	Coffee_Mug
A_7	Vegetable_Basket	A_{33}	Coffee_Machine
A_8	Freezer	A_{34}	Sugar
A_9	Plate	A_{35}	Spoon
A_{10}	Knife	A_{36}	Laptop/PC
A_{11}	Salt	A_{37}	Video_App
A_{12}	Pepper	A_{38}	Laundry_Basket
A_{13}	Seasoning	A_{39}	Washer_Dryer
A_{14}	Oven	A_{40}	Cleaning_Sponge
A_{15}	Pizza_Tray	A_{41}	Cleaning_Spray
A_{16}	Bowl	A_{42}	Kitchen_Shelves
A_{17}	Slicer	A_{43}	Cola_Glass
A_{18}	Eggs	A_{44}	Salad_Bowl
A_{19}	Cheese	A_{45}	Yogurt_Bowl
A_{20}	Bread_Box	A_{46}	Presentation_App
A_{21}	Frypan	A_{47}	Desk
A_{22}	Oil	A_{48}	Document_App
A_{23}	Whisker	A_{49}	Browser_App
A_{24}	Tooth_Brush	A_{50}	Treadmill
A_{25}	Bathroom_Faucet	A_{51}	Office_Bag/Laptop_Bag
A_{26}	Wardrobe		

Table VII. List of Context Attributes

Context Attribute C_i	Description	Context Attribute C_i	Description
C_1	Home	C_{12}	Office
C_2	Kitchen	C_{13}	Office_Room
C_3	Bedroom	C_{14}	Stove_On
C_4	Bathroom	C_{15}	Morning
C_5	Lounge	C_{16}	Afternoon
C_6	Dining_Room	C_{17}	Evening
C_7	Kitchen_Light_On	C_{18}	Night
C_8	Bedroom_Light_On	C_{19}	Direction_To_Office
C_9	Bathroom_Light_On	C_{20}	Direction_To_Home
C_{10}	Lounge_Light_On	C_{21}	Direction_To_Gym
C_{11}	Dining_Light_On	C_{22}	Direction_To_City

REFERENCES

Robert P. Abelson. 1981. Psychological status of the script concept. *American Psychologist* 36, 7 (1981), 715–729.

Gregory D. Abowd and Elizabeth D. Mynatt. 2000. Charting past, present, and future research in ubiquitous computing. *ACM Trans. Comput.-Hum. Interact.* 7, 1 (March 2000), 29–58. DOI: http://dx.doi.org/10.1145/344949.344988.

Fahd Albinali, Nigel Davies, and Adrian Friday. 2007. Structural learning of activities from sparse datasets. In *Proceedings of the 5th International Conference on Pervasive Computing and Communications (PerCom'07)*. 221–228.

Oliver Amft and Gerhard Troster. 2008. Recognition of dietary activity events using on-body sensors. *Artif. Intell. Med.* 42, 2 (2008), 121–136. 0933-3657

Android-Platform. 2010. Android Platform. Retrieved May 17, 2010 from http://www.android.com/.

Ling Bao. 2003. *Physical Activity Recognition from Acceleration Data under Semi-Naturalistic Conditions*. Ph.D. dissertation. Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science.

L. Bao and S. Intille. 2004. Activity Recognition from User-Annotated Acceleration data. In *Proc. Pervasive*. 1–17.

Roland Bennett. 2007. Time Tracker: PC Activity Monitor for Windows. Retrieved May 17, 2010 from http://ttracker.sourceforge.net/.

O. Brdiczka, J. Crowley, and P. Reignier. 2007. Learning situation models for providing context-aware services. In *Universal Access in Human-Computer Interaction. Ambient Interaction*, Constantine Stephanidis (Ed.). Lecture Notes in Computer Science, Vol. 4555. Springer, Berlin, 23–32.

Oliver Brdiczka, Norman Makoto Su, and James Bo Begole. 2010. Temporal task footprinting: Identifying routine tasks by their temporal patterns. In *Proceedings of the 15th International Conference on Intelligent User Interfaces (IUI'10)*. ACM, New York, NY, 281–284. DOI: http://dx.doi.org/10.1145/1719970.1720011.

Liming Chen, C. D. Nugent, and Hui Wang. 2012. A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering* 24, 6 (June 2012), 961–974. 1041-4347 DOI: http://dx.doi.org/10.1109/TKDE.2011.51.

T. Choudhury, S. Consolvo, B. Harrison, J. Hightower, A. LaMarca, L. LeGrand, A. Rahimi, A. Rea, G. Bordello, B. Hemingway, P. Klasnja, K. Koscher, J. A. Landay, J. Lester, D. Wyatt, and D. Haehnel. 2008. The mobile sensing platform: An embedded activity recognition system. *Pervasive Computing, IEEE* 7, 2 (2008), 32–41.

Sunny Consolvo, David W. McDonald, Tammy Toscos, Mike Y. Chen, Jon Froehlich, Beverly Harrison, Predrag Klasnja, Anthony LaMarca, Louis LeGrand, Ryan Libby, Ian Smith, and James A. Landay. 2008. Activity sensing in the wild: a field trial of ubifit garden. In *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems (CHI'08)*. ACM, New York, NY, 1797–1806. DOI: http://dx.doi.org/10.1145/1357054.1357335.

Nigel Davies, Daniel P. Siewiorek, and Rahul Sukthankar. 2008. Activity-based computing. *Pervasive Computing, IEEE* 7, 2 (2008), 20–21.

J. W. Deng and H. T. Tsui. 2000. An HMM-based approach for gesture segmentation and recognition. In *Proceedings of the 15th International Conference on Pattern Recognition*, Vol. 3. 679–682.

Anind Kumar Dey. 2000. *Providing Architectural Support for Building Context-Aware Applications*. Ph.D. Dissertation. Georgia Institute of Technology.

GreenerBuildings. 2010. GreenerBuildings. Retrieved May 17, 2010 from http://www.greenerbuildings.eu/sites/greenerbuildings.eu/files/GreenerBuildingsFiche.pdf.

G. Guerra-Filho and Y. Aloimonos. 2007. A language for human action. *Computer* 40, 5 (2007), 42–51.

R. Hamid, S. Maddi, A. Bobick, and M. Essa. 2007. Structure from statistics—unsupervised activity analysis using suffix trees. In *Proceedings of the 11th International Conference on Computer Vision (ICCV'07)*. 1–8.

Rim Helaoui, Mathias Niepert, and Heiner Stuckenschmidt. 2011a. Recognizing interleaved and concurrent activities: A statistical-relational approach. In *Proceedings of the 2011 IEEE International Conference on Pervasive Computing and Communications (PerCom'11)*. IEEE Computer Society, 1–9.

Rim Helaoui, Mathias Niepert, and Heiner Stuckenschmidt. 2011b. Recognizing interleaved and concurrent activities using qualitative and quantitative temporal relationships. *Pervasive and Mobile Computing* 7, 6 (2011), 660–670. DOI: http://dx.doi.org/10.1016/j.pmcj.2011.08.004.

I. Homed, A. Misra, M. Ebling, and W. Jerome. 2008. HARMONI: Context-aware filtering of sensor data for continuous remote health monitoring. In *Proceedings of the 6th International Conference on Pervasive Computing and Communications (PerCom'08)*. 248–251.

Xin Hong, Chris Nugent, Maurice Mulvenna, Sally McClean, Bryan Scotney, and Steven Devlin. 2009. Evidential fusion of sensor data for activity recognition in smart homes. *Pervasive and Mobile Computing* 5, 3 (2009), 236–252. DOI: http://dx.doi.org/10.1016/j.pmcj.2008.05.002.

Derek Hao Hu and Qiang Yang. 2008. CIGAR: Concurrent and interleaving goal and activity recognition. In *Proceedings of the 23rd National Conference on Artificial Intelligence, Volume 3*. AAAI Press, Chicago, Illinois.

T. Huynh and B. Schiele. 2006. Unsupervised discovery of structure in activity data using multiple eigenspaces. In *Proceedings of the International Conference on Location- and Context-Awareness (LoCA'06)*. 151–167.

E. Jovanov, A. O'Donnell Lords, D. Raskovic, P. G. Cox, R. Adhami, and F. Andrasik. 2003. Stress monitoring using a distributed wireless intelligent sensor system. *IEEE Engineering in Medicine and Biology Magazine* 22, 3 (2003), 49–55.

V. Kaptelinin, BA Nardi, and C Macaulay. 1999. Methods and tools: The activity checklist: A tool for representing the space of context. *interactions* 6, 4 (1999), 27–39.

T. Kasteren, G. Englebienne, and B. Krose. 2010. Transferring knowledge of activity recognition across sensor networks. In *Pervasive Computing*, Vol. 6030. Springer Berlin, 283–300. DOI: http://dx.doi.org/10.1007/978-3-642-12654-3_17.

Tim van Kasteren, Athanasios Noulas, Gwenn Englebienne, and Ben Krose. 2008. Accurate activity recognition in a home setting. In *Proceedings of the 10th International Conference on Ubiquitous Computing (Ubicomp'08)*. ACM, Seoul, Korea.

Young-Seol Lee and Sung-Bae Cho. 2011. Activity recognition using hierarchical hidden Markov models on a smartphone with 3D accelerometer. In *Hybrid Artificial Intelligent Systems*. Lecture Notes in Computer Science, Vol. 6678. Springer, Berlin, 460–467. DOI: http://dx.doi.org/10.1007/978-3-642-21219-2_58.

Ian Li, Anind K. Dey, and Jodi Forlizzi. 2012. Using context to reveal factors that affect physical activity. *ACM Trans. Comput.-Hum. Interact.* 19, 1, Article 7 (May 2012), 21 pages. DOI: http://dx.doi.org/10.1145/2147783.2147790.

L. Liao, T. Choudhury, D. Fox, and H. Kautz. 2007. Training conditional random fields using virtual evidence boosting. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'07)*. 2530–2535.

Paul Lukowicz, Jamie A. Ward, Holger Junker, Mathias Stäger, Gerhard Tröster, Amin Atrash, and Thad Starner. 2004. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing*. 18–32.

Sven Meyer and Andry Rakotonirainy. 2003. A survey of research on context-aware homes. In *Proceedings of the Australasian Information Security Workshop Conference on ACSW Frontiers 2003, Volume 21*. 159–168.

S. P. Meyn and R. L. Tweedie. 1993. *Markov Chains and Stochastic Stability*. Springer-Verlag.

D. Minnen, T. Starner, M. Essa, and C. Isbell. 2006. Discovering characteristic actions from on-body sensor data. In *Proceedings of the 10th IEEE International Symposium on Wearable Computers (ISWC'06)*. 11–18.

Mulle-v3. 2010. Mulle v3 Sensor Platform. Retrieved May 17, 2010 from http://www.eistec.se/.

D. Ocone. 2010. Lecture Notes in Discrete and Probabilistic Models in Biology: Markov Chain Models. Retrieved May 17, 2010 from http://www.math.rutgers.edu/courses/338/coursenotes/chapter5.pdf.

Nuria Oliver, Eric Horvitz, and Ashutosh Garg. 2002. Layered representations for human activity recognition. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces (ICMI'02)*. IEEE Computer Society.

Amir Padovitz, Seng Wai Loke, and Arkady Zaslavsky. 2004. Towards a theory of context spaces. In *Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops (PerCom'04).*

Amir Padovitz, Seng Wai Loke, and Arkady B. Zaslavsky. 2008. Multiple-agent perspectives in reasoning about situations for context-aware pervasive computing systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part A* 38, 4 (2008), 729–742.

Etienne Pardoux. 2008. *Markov Processes and Applications: Algorithms, Networks, Genome and Finance*. John Wiley and Sons Ltd.

M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel. 2004. Inferring activities from interactions with objects. *Pervasive Computing, IEEE* 3, 4 (2004), 50–57.

P. Rashidi, D. Cook, L. Holder, and M. Schmitter-Edgecombe. 2011. Discovering activities to recognize and track in a smart environment. *IEEE Transactions on Knowledge and Data Engineering* 23, 4 (2011), 527–539.

Parisa Rashidi and Diane J. Cook. 2011. Activity knowledge transfer in smart environments. *Pervasive and Mobile Computing* 7, 3 (2011), 331–343.

Daniele Riboni and Claudio Bettini. 2009. Context-aware activity recognition through a combination of ontological and statistical reasoning. In *Proceedings of Conference on Ubiquitous Intelligence and Computing (UIC'09)*. 39–53. http://dx.doi.org/10.1007/978-3-642-02830-4_5.

Daniele Riboni and Claudio Bettini. 2011. COSAR: Hybrid reasoning for context-aware activity recognition. *Personal Ubiquitous Comput.* 15, 3 (March 2011), 271–289. DOI: http://dx.doi.org/10.1007/s00779-010-0331-7.

M. S. Ryoo and J. K. Aggarwal. 2006. Recognition of composite human activities through context-free grammar based representation. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2. 1709–1718.

Saguna, Arkady Zaslavsky, and Dipanjan Chakraborty. 2011. Complex activity recognition using context driven activity theory in home environments. In *Smart Spaces and Next Generation Wired/Wireless Networking*, Sergey Balandin, Yevgeni Koucheryavy, and Honglin Hu (Eds.). Lecture Notes in Computer Science, Vol. 6869. Springer, Berlin, 38–50. DOI: http://dx.doi.org/10.1007/978-3-642-22875-9_4.

Roger C. Schank and Robert P. Abelson. 1975. Scripts, plans, and knowledge. In *Proceedings of the 4th International Joint Conference on Artificial Intelligence, Volume 1(IJCAI'75)*. Morgan Kaufmann, San Francisco, CA, 151–157. http://dl.acm.org/citation.cfm?id=1624626.1624649.

Thomas Stiefmeier. 2008. *Real-Time Spotting of Human Activities in Industrial Environments*. Ph.D. Dissertation. ETH, Zürich.

Thomas Stiefmeier, Daniel Roggen, and Gerhard Troster. 2007. Gestures are strings: Efficient online gesture spotting and classification using string matching. In *Proceedings of the 2nd International Conference on Body Area Networks*.

T. Stiefmeier, D. Roggen, G. Troster, G. Ogris, and P. Lukowicz. 2008. Wearable activity tracking in car manufacturing. *Pervasive Computing, IEEE* 7, 2 (2008), 42–50.

M. Stikic, T. Huynh, K. Van Laerhoven, and B. Schiele. 2008. ADL recognition based on the combination of RFID and accelerometer sensing. In *Proceedings of the 2nd International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth'08)*. 258–263.

Holger Storf, Thomas Kleinberger, Martin Becker, Mario Schmitt, Frank Bomarius, and Stephan Prueckner. 2009. An Event-Driven Approach to Activity Recognition in Ambient Assisted Living Ambient Intelligence. In *Lecture Notes in Computer Science*. Springer Berlin/Heidelberg, 123–132. http://dx.doi.org/10.1007/978-3-642-05408-2_16.

Gu Tao, Wu Zhanqing, Tao Xianping, Pung Hung Keng, and Lu Jian. 2009. epSICAR: An Emerging Patterns based approach to sequential, interleaved and Concurrent Activity Recognition. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom'09)*. 1–9.

Emmanuel Munguia Tapia, Stephen S. Intille, and Kent Larson. 2004. Activity recognition in the home using simple and ubiquitous sensors. In *Proceedings of the 2nd International Conference on Pervasive*. 158–175.

M. Tentori and J. Favela. 2008. Activity-aware computing for healthcare. *Pervasive Computing, IEEE* 7, 2 (2008), 51–57.

Huawei Tu, Xiangshi Ren, and Shumin Zhai. 2012. A comparative evaluation of finger and pen stroke gestures. In *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems (CHI'12)*. ACM, New York, NY, 1287–1296. DOI: http://dx.doi.org/10.1145/2207676.2208584.

Kristof Van Laerhoven. 2001. Combining the self-organizing map and K-means clustering for on-line classification of sensor data. In *Proceedings of the International Conference on Artificial Neural Networks*. 464–469–. http://dx.doi.org/10.1007/3-540-44668-0_65.

K. Van Laerhoven and O. Cakmakci. 2000. What shall we teach our pants?. In *Proceedings of the 4th International Symposium on Wearable Computers*. 77–83.

Shiaokai Wang, William Pentney, Ana-Maria Popescu, Tanzeem Choudhury, and Matthai Philipose. 2007. Common sense based joint training of human activity recognizers. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*.

J. A. Ward, P. Lukowicz, G. Troster, and T. E. Starner. 2006. Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 28, 10 (2006), 1553–1567.

Tsu-Yu Wu, Chia-Chun Lian, and Jane Hsu. 2007. Joint Recognition of multiple concurrent activities using factorial conditional random fields. In *Proceedings of the 2007 AAAI Workshop on Plan, Activity, and Intent Recognition.* AAAI Press, Menlo Park.

Guang-Zhong Yang, Benny Lo, and Surapa Thiemjarus. 2006. *Body Sensor Networks*. Springer, London.

Juan Ye, L. Coyle, S. Dobson, and P. Nixon. 2007. Using situation lattices to model and reason about context. In *Proceedings of the 4th International Workshop on Modeling and Reasoning in Context (MRC'07).* 1–12.

Juan Ye, L. Coyle, S. Dobson, and P. Nixon. 2009. Using situation lattices in sensor analysis. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom'09).* 1 –11. DOI: http://dx.doi.org/10.1109/PERCOM.2009.4912762.

Shumin Zhai and Victoria Bellotti. 2005. Introduction to sensing-based interaction. *ACM Trans. Comput.-Hum. Interact.* 12, 1 (March 2005), 1–2. DOI: http://dx.doi.org/10.1145/1057237.1057238.

Received June 2012; revised October 2012; accepted May 2013