ReDef: Context-aware Recognition of Interleaved Activities using OWL 2 and Defeasible Reasoning

Georgios Meditskos, Efstratios Kontopoulos, Ioannis Kompatsiaris

Information Technologies Institute, Centre for Research and Technology Hellas
{gmeditsk, skontopo, ikom}@iti.gr

Abstract. Understanding human activities in pervasive environments is a key challenge that involves fusion and correlation of multimodal sensor information. Many research efforts have been recently focused on knowledge-driven solutions to human activity recognition, using ontologies for defining activity models and for capturing contextual information. In most cases, however, the unrealistic assumption is made that activities are performed in a sequential, non-interrupted manner, hampering their applicability in real-world scenarios. In this paper, we present a framework for detecting interleaved activities of daily living (ADL) using (a) OWL 2 for implementing the underlying model semantics capturing contextual dependencies among activities, and (b) defeasible reasoning for introducing a flexible conflict resolution mechanism. The proposed framework has been integrated in an existing context-aware ADL recognition framework, which is being used for supporting the diagnosis of the Alzheimer's disease in a controlled environment.

Keywords: ontologies, defeasible reasoning, interleaved activities, context

1 Introduction

In recent years, the demand for intelligent, customized user task support has proliferated across a multitude of application domains, ranging from healthcare and smart spaces to transportation and energy control [1]. In this quest, pervasive computing and sensor technologies have driven the construction of ubiquitous computing environments, transforming regular physical spaces into intelligent spaces capitalizing on the ability to sense, process, combine and interpret data of different modalities.

Out of the numerous domains of interest, the recognition of human activities is a notable case where pervasive frameworks provide unique solutions for the contextualized monitoring and assessment of daily activities and human behaviour. For example, in the healthcare sector, the employment of multiple sensors and modalities for monitoring daily activities of elderly people has many benefits towards improving healthcare support [25]. A key challenge in such domains is the ability to effectively fuse multiple sources of heterogeneous, noisy and potentially inconsistent information in such a way that will provide accurate and useful outputs.

Given the inherently open nature of pervasive, sensor-driven systems, where a crucial requirement is the need to aggregate low-level contextual information and meaningfully integrate domain knowledge, it comes as no surprise that Semantic Web technologies have been acknowledged as affording a number of highly desirable features. In this context, ontologies provide the vocabulary for the representation of low-level sensory observations (e.g. from video cameras, contact sensors, wearable devices etc.), while background knowledge is captured using complex class descriptions (axioms) that encapsulate contextual information specific to the domain (e.g. complex activity models). In many cases, the domain ontology models are further enhanced with rules for expressing richer relationships, like e.g. temporal. This coupling of (low-level) data models and semantically rich domain descriptions enables the derivation of high-level interpretations regarding the behaviour of individuals, e.g. by recognizing activities of daily living (ADLs), through intelligent fusion and reasoning mechanisms.

Several ontology-based reasoning architectures and prototypes have been proposed for activity recognition (see Section 2), each of which follows a different approach for handling intrinsic characteristics of the domain, such as data heterogeneity, temporal extension, noise, uncertainty and missing information. However, little focus has been given on the recognition of *interleaved activities* (i.e. non-consecutive), simplifying the problem of activity recognition to only recognizing sequential activities, which is usually an unrealistic assumption. In real-world situations, activities may be performed in an interleaving manner, where one activity may be temporarily paused in order to perform one or more other activities. For example, an individual may be preparing a tea when the phone rings, so they have to pause the activity to answer the phone. Key challenges in this context involve the recognition of the start and end timestamps of all the activities involved and the derivation of the contextual interval when each activity was active, e.g. to classify interrupted instances of the same task as a single activity.

In this paper, we investigate the use of *defeasible reasoning* [19] for detecting and classifying interleaved activities. Defeasible reasoning deploys a flexible conflict resolution framework for handling inconsistent and conflicting information, which is typical for (inherently uncertain and noisy) data coming from heterogeneous sensors. More specifically, we define a defeasible reasoning layer that can be used on top of existing ADL frameworks to facilitate the recognition of interleaved activities. Our framework (ReDef) is based on the use of OWL 2 ontology models for capturing common sense knowledge regarding the context of the domain activities, and provides a set of defeasible rules that introduce semantic relationships among interleaved activities, such as telicity and contextual dependencies. The proposed framework has been integrated in a multi-level context-aware framework for ADL recognition [16] that is being used for assessing the diagnosis of Alzheimer's disease in control environments.

The paper is structured as follows: Section 2 reviews existing ontology-based approaches in recognizing ADLs and interleaved activities. Section 3 features a brief introduction to defeasible logics, followed by Section 4 that describes the problem. Section 5 presents the OWL 2 ontology models we have developed for modelling contextual information of activities that are fed into the defeasible logic layer (Section 6) for supporting the recognition of interleaved activities. Section 7 elaborates on the deployment of the framework in a real-world scenario and Section 8 concludes the paper.

2 Related Work

OWL (and OWL 2) has been widely used for modelling activity semantics, reducing complex activity definitions to the intersection of their constituent parts [5]. In most cases, the activity recognition process involves the segmentation of the data into snapshots of atomic events that are fed into the ontology reasoner for classification. Time windows [20] and slices [23], background knowledge about the order or duration [22] of activities are common approaches for segmentation. In addition, rules have been embraced as a means for compensating for the expressive limitations of OWL [26, 18]. In this paradigm, ontologies are used for modelling domain information, whereas rules aggregate activities, describing the conditions that drive the derivation of complex activities, e.g. temporal relations. In order to address additional intrinsic characteristics of the domains, such as uncertainty and missing information, several approaches have been also devoted to extending formalisms and reasoning services. Examples include, among others, fuzzy and probabilistic extensions of OWL and SWRL [6, 12, 24].

People often multitask, interrupt and switch between different types of activities, such as making lunch and answering the phone. Those activities can be characterized as interleaved activities. In other cases, individuals pursue different goals at the same time without interrupting any of them. For example, eating and watching TV at the same time would classify as concurrent activities. Therefore, a key challenge for human activity recognition in realistic pervasive environments is the ability to correctly segment and recognize non-sequential and uninterrupted activities, such as interleaved and concurrent activities. In this paper, we focus on the recognition of interleaved activities.

Despite the benefits that ontology-based reasoning solutions offer to activity recognition frameworks (e.g. modelling of complex logical relations, sharing information coming from heterogeneous sources, availability of sound and complete reasoning engines), little focus has been given on the recognition of interleaved activities. In [11], the problem of detecting interleaved activities is approached by combining statistical-temporal models obtained from training data and background knowledge in the form of temporal first-order rules. Although the combination of data- and knowledge-driven solutions seems promising, the definition of strict temporal rules often fails to incorporate the level of flexibility required in pervasive environments. The framework presented in [27] is able to recognize multi-user concurrent activities using ontologies. Although this work focuses on the detection of activities performed simultaneously by different individuals, the adopted approach for recognizing false sensor activations where activities are mapped on is based on the Pyramid Match Kernel technique.

In [28], activities are inferred using an ontology model and rules that check the knowledge base for temporal overlaps between atomic activities relating to different complex activities. The limitation of this approach is that the rules are static and predefined, meaning that all the temporal relations need to be explicitly defined at design time. In [21], a knowledge-driven agent-mediated approach based on hybrid ontological and temporal formalisms for composite activity recognition is presented. Data segmentation is performed using time windows. Ontological reasoning is used both for deriving primitive actions and complex activities using subsumption and equivalence reasoning. In each segment, more than one activity might be detected, which is considered as interleaved. However, no information is provided about the semantic conditions that drive the derivation and further aggregation of individual interleaved activity instances.

Finally, regarding the deployment of defeasible logics in pervasive computing environments, the work presented in [3] constitutes the main recent research effort investigating this setting. In their work, the authors propose a distributed reasoning approach based on the representation of context knowledge shared by the ambient agents in the environment. Taking into consideration the highly dynamic nature of the setting, defeasible logic is proposed as the basis for representing the context knowledge possessed by each agent (i.e. the agent's local rule base). Additionally, defeasible logic is also applied for resolving the potential conflicts that arise from the information exchange between the agents.

3 Defeasible Logics

Defeasible logics is a non-monotonic logics formalism that delivers intuitive knowledge representation and advanced conflict resolution mechanisms [19]. In defeasible logics there are three distinct types of rules:

- *Strict rules* are denoted by $A \rightarrow p$, where A is a set of literals and p is a (positive or negative) literal, and are interpreted in the typical sense: whenever the premises are indisputable, then so is the conclusion.
- *Defeasible rules* are denoted by $A \Rightarrow p$ and, contrary to strict rules, they can be defeated by contrary evidence. Two examples of defeasible rules are r_1: *holdsFork*(X) $\Rightarrow$ *havingLunch*(X), which reads as "*if individual X* (i.e. the inhabitant of the house) *is holding a fork then he/she is probably having lunch*", and r_2: *onThePhone*(X) $\Rightarrow$ $\neg$*havingLunch*(X), which reads as "*when X is on the phone then he/she is probably not having lunch*".
- *Defeaters* are denoted by $A \rightsquigarrow p$ and do not actively support conclusions, but can only prevent deriving some of them. In other words, they are used to defeat respective defeasible conclusions, by producing evidence to the contrary. A defeater example is: r_1': *sleep*(X) $\rightsquigarrow$ $\neg$*havingLunch*(X) ("*when X is sleeping then he/she is definitely not having lunch*"), which can defeat e.g. rule r_1 mentioned previously.

Additionally, the *superiority relationship* is used for resolving conflicts among defeasible rules. For example, given the defeasible rules r_1 and r_2 above, no conclusive decision can be made about whether X is having lunch or not. But, if the superiority relationship $r_2 > r_1$ is introduced, then r_2 overrides r_1 and we can eventually conclude that X is not having lunch after all. In this case rule r_2 is called *superior* to r_1 and r_1 *inferior* to r_2. Note that the relation > is acyclic.

The advantages of applying defeasible instead of classical logics are outlined as follows:

- Defeasible logics have low computational complexity [15];

- They allow for reasoning with incomplete information; this is a critical trait in sensor environments, where perfect knowledge of the environment is very hard, if not impossible, to achieve;
- They introduce non-monotonicity, which leads to a more intuitive type of reasoning, much closer to human reasoning especially for the non-accustomed users (e.g. doctors, patients, etc.), where the emergence of new information can lead to abandoning (i.e. defeating) previously established conclusions and adopting new ones.

4 Problem Description

In previous work [16], we investigated the viability of a multi-level context-aware framework for recognizing ADLs. A key feature of the framework lies on the use of ontologies for defining activity models as dependencies (links) between complex activities and their low-level observations that we call *situation descriptors*. For example, the situation descriptor of making tea links the `MakeTea` domain class to its lower level observation types, such as objects used (e.g. `Cup`, `Spoon`) and location (e.g. `TeaZone`). Given a set of low-level observations and a set of situation descriptors, the context-aware algorithm segments the initial trace of observations into meaningful contexts, i.e. clusters of observations, that are classified (with some plausibility) as complex activities, generating semantically enriched knowledge graphs with activity traces.

Despite the promising results we obtained by evaluating the framework in realistic environments, the assumption that individuals carry out a single activity each time falls short when handling interleaved activities. In this case, the interleaved contexts are recognized as individual activities, affecting the performance of the algorithm. In order to support the recognition of interleaved ADLs and to subsequently improve the accuracy of the framework, we have developed *ReDef*, a knowledge-driven decision making layer for the context-aware aggregation of non-sequential contexts. More specifically, given an RDF graph with detected activities, along with their pertinent lower-level observations, our framework aims to identify and link non-consecutive activity contexts that belong to the same overall activity task. In the following section we describe the ontologies we use for modelling domain knowledge, capturing the concept of *activity telicity*, along with the defeasible rules that implement the underlying model semantics.

5 Modelling Activity Telicity

ReDef provides two lightweight ontology patterns for capturing the concept of activity telicity, i.e. the context that designates that an activity has been completed. Both patterns implement the *descriptions and situations* (*DnS*) ontology pattern [8] of *DOLCE Ultra Lite* (*DUL*) ontology and make use of the meta-modelling capabilities of OWL 2, namely *punning* [13], allowing property assertions to be made among activity classes. In that way, we enable the representation of contextualised views on complex activities, and afford reusable pieces of knowledge that cannot otherwise be directly expressed by the standard ontology semantics, e.g. temporal correlations among activities that are not connected in a tree-like manner.

5.1 Telic Event Pattern

The telic event pattern enables to formally define the terminating state of a complex activity, i.e. the observation type that belongs to the activity's situation descriptor and denotes the completion of the activity. This pattern can be used for modelling telicity either for activities that do have endpoints, e.g. the event of turning off the TV can be considered as the telic event of watching TV. Fig. 1 (a) depicts the schema of the telic event pattern, while Fig. 1 (b) illustrates an example instantiation for modelling the telic event of watching TV. Following the conceptual model of DnS, the instantiation of the pattern involves the definition of a description instance that captures information about the activity type of interest and the telic event. The conceptual model of DnS also requires the assertion of a situation instance that references (via the `hasDescription` property assertion) the description instance. It is worth noting that the instantiation of the pattern involves the use of ontology classes in property assertions, e.g. in `define-ActivityType`. The circles in Fig. 1 (b) denote anonymous ontology instances that instantiate the pattern's concepts.

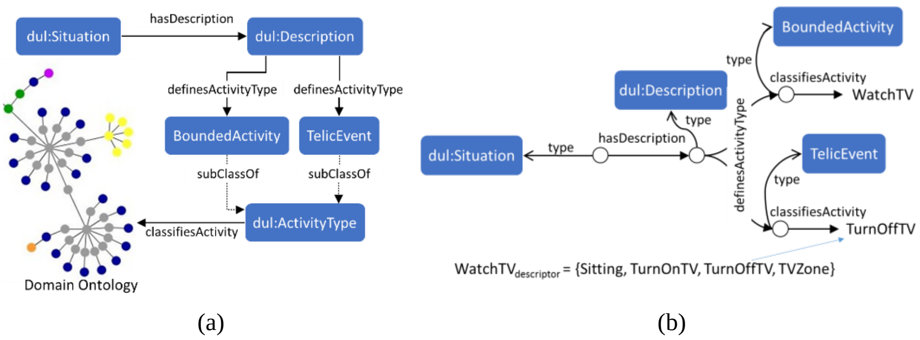

Fig. 1. (a) Telic event pattern; (b) Example instantiation for the `WatchTV` activity.

5.2 Inter-context Telicity

While for some activities it is possible to select an observation from their situation descriptors to play the role of the telic event, there are other activities that cannot be bounded to specific endpoints. For example, preparing breakfast is a dynamic task that involves many activities without a predefined order or terminating contexts. For such activities, telicity cannot be defined by means of an observation that belongs to the situation descriptors.

In order to support the concept of telicity for activities that cannot be explicitly linked with a terminating state, ReDef provides the pattern depicted in Fig. 2 (a). The idea behind this pattern is to capture activity telicity by means of existence of another context (inter-context telicity). For example, the detection of an activity relevant to cleaning the table in the morning is an indication that the individual may have prepared a breakfast earlier, which can be considered as completed. Similar to the telic event pattern, the instantiation of this pattern requires the assertion of situation and description instances, designating the role of each instance by assigning it to the available concepts (`BoundedActivity` or `TelicContext`). Moreover, this pattern allows us to capture temporal dependencies among the bounded activities and the respective contexts. For example, the instantiation of the pattern in Fig. 2 (b) explicitly models that the cleaning table context should follow the prepare breakfast activity.

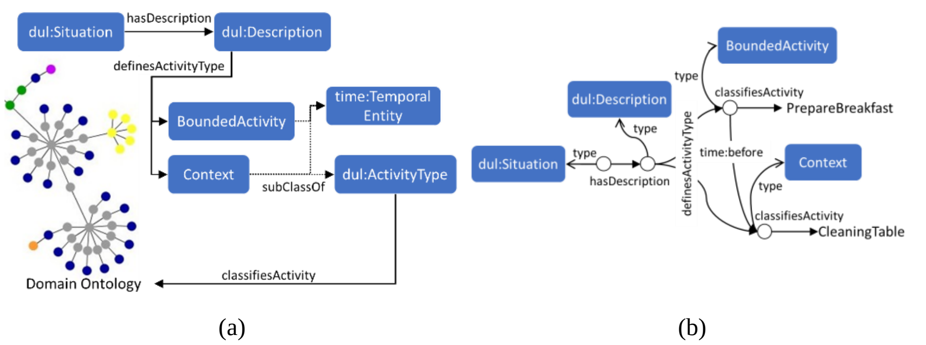

Fig. 2. (a) Inter-context telicity pattern; (b) Example instantiation for `PrepareBreakfast`.

6 Recognizing Interleaved Activities

In the previous section we presented the patterns supported by ReDef for modelling activity telicity. In this section, we describe the defeasible reasoning layer of ReDef that utilises this knowledge in order to aggregate and derive interleaved activities.

6.1 Prerequisites

The aim of ReDef is to provide a framework that can be used on top of existing activity recognition solutions in order to enhance their performance with respect to the detection of interleaved activities. This is achieved by examining the already detected activities and their constituent observations to detect situations when the telicity patterns are satisfied in order to further aggregate the individual activities and derive interleaved tasks. As such, ReDef requires as input the following information:

- **Activity traces:** set of detected complex activities with start/end timestamps.
- **Sub-events:** the constituent parts (observations) of the complex activities.
- **Activity telicity patterns:** instantiations of the patterns described in Section 5.

In the following, we assume that the rule-based methodology for determining which activities are interleaved is based on the following set of core predicates:

- `activity(A,T1,T2)`: `A` is an activity starting at `T1` and ending at `T2`.
- `type(A,P)`: Resource (observation/activity) `A` is of activity type `P`.
- `subEvent(O,A)`: Observation `O` belongs to activity `A`.

6.2 Interleaved Activities Through Direct Telicity

The following set of defeasible rules implements the semantics of the telic event pattern described in Section 5.1, asserting pairs of interleaved activities. In addition to the core predicates, the predicate `telic(TL,A)` is defined that denotes that `TL` is the telic event for activity `A`.

```
r1: activity(A1,T11,T12), activity(A2,T21,T22), T21 > T12,
type(A1,A), type(A2,A), telic(TL,A), subEvent(Z,A2), type(Z,TL)
⇒ interleaved(A1,A2)

r2: activity(A1,T11,T12), activity(A2,T21,T22), T21 > T12,
type(A1,A), type(A2,A), telic(TL,A), subEvent(Z,A1), type(Z,TL)
⇒ ¬interleaved(A1,A2)

r3: activity(A1,T11,T12), activity(A2,T21,T22), activ-
ity(A3,T31,T32), T21 > T12, T31 > T22, type(A1,A), type(A2,A),
type(A3,A), telic(TL,A), subEvent(Z1,A2), subEvent(Z2,A3),
type(Z1,TL), type(Z2,TL)
⇒ ¬interleaved(A1,A3)

r2, r3 > r1
```

More specifically, rule r_1 determines when two separate activities constitute a single, interleaved one, based on the existence of the corresponding telic observation in the activity context that takes place last. On the other hand, rule r_2 establishes an exception to r_1 that takes place when the first activity (also) includes a telic observation. An additional exception, r_3, ensures that an activity is linked only with the most recent telic context. Consequently, these exceptions are introduced as superior to r_1 via the superiority relationship. When the execution of rules terminates, the pair of intervened activities are traversed to select the one with the longest duration as the final activity.

6.3 Interleaved Activities Through Inter-context Telicity

In order to implement the semantics of the inter-context telicity pattern described in Section 5.2, the `telic` predicate is replaced by predicate `final(A)` indicating that activity `A` is completed (no subsequent activities of the same type may be appended to `A`), according to the pattern in Fig. 2. The following rule determines the `final` activities:

```
r4: activity(A1,T11,T12), activity(B1,T21,T22), latest(A1,B1),
type(A1,A), type(B1,B), telicContext(A,B)
⇒ final(A1)
```

where [`latest(A1,B1), type(A1,A), type(B1, B)`] retrieves the closest most recent activity of type `A` to type `B`.

Having detected the `final` activities, a rule set similar to the one presented in the previous subsection (rules r_2-r_3) has to be deployed, where the `telic` predicate is substituted by `final`.

7 Use Case and Discussion

ReDef is part of an ADL recognition framework deployed in a hospital for monitoring Alzheimer's disease patients[1]. The aim of this deployment is to help clinicians assess the condition of individuals, based on a goal-directed protocol where participants perform predefined activities in an experimentation room. The participants have to perform a list of 10 Instrumental Activities of Daily Living (IADL), i.e. tasks that support an independent life style, such as preparing the drug box, talking on phone, preparing tea and watering the plant. Automated ADL recognition is employed in this context for detecting the IADLs performed by the participants and for informing the clinicians, who are not in the room during the execution of the protocol, about activities that have been missed or repeated, or problems regarding the duration of activities. The setting involves ambient and wearable video and audio sensors, accelerometers and physiological sensors. The collected sensor data, such as location with respect to predefined zones, objects the participants interact with, posture and state of appliances are analysed by software modules to recognise activities of participants.

The majority of the tasks involved in the protocol can be performed in a sequential manner, such as watering the plant or making a phone call. However, despite the promising ADL recognition results we obtained, we observed a low accuracy in detecting the preparation of hot tea. This was due to the fact that the majority of the participants performed this task in an interleaved manner: after putting water in the kettle and turning the kettle on, participants went on with other tasks before coming back and completing the preparation of the tea. In this case, the ADL recognition framework detects two separate activities that trigger the generation of a problem to be highlighted to the clinical experts regarding activity repetition. ReDef has been integrated in this setting in order to overcome this limitation and support the detection of interleaved activities.

Fig. 3 depicts the instantiation of the telic event pattern that defines telicity by means of the `FillCup` event. Fig. 4 presents example observations and detected activities during a protocol. As explained above, the ADL recognition algorithm recognizes two `PrepareTea` activities (with different plausibility, since different numbers of tea-related observations are involved in each context) based on the provided situation descriptor. In this example, ReDef will aggregate the two individual activities, taking into account the information encapsulated in the pertinent telic event pattern[2].

ReDef has been tested so far with a small number of protocol participants, since the experiment is still ongoing. Preliminary results indicate that the system is able to correctly detect the start/end times of interleaved activities in the majority of the situations.

---

[1] The system has been installed in the Memory Resource and Research Centre (CMRR) of the University Hospital in Nice (CHUN), under the Dem@Care FP7 EU Project.

[2] The implementation of the defeasible reasoning layer is currently based on SPINdle, a Java-based defeasible reasoning engine [14].

Problems have been identified in cases when the analysis modules fail to detect the telic event of an activity, e.g. the `FillCup` events in our example. In this case, telicity cannot be inferred and the detection of interleaved activities fails. We are currently investigating the extension of the defeasible rules presented in Section 6, so as to handle missing information, e.g. by integrating negation-as-failure or more refined/explicit rules expressing exceptions.

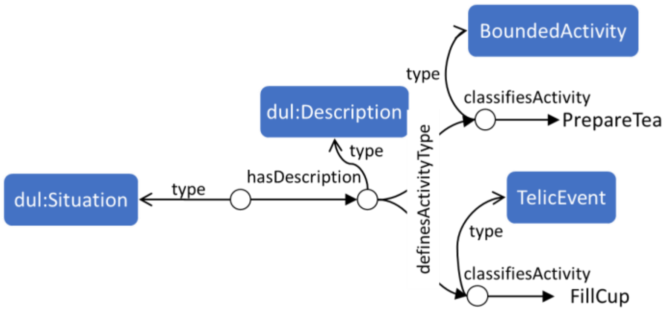

Fig. 3. Instantiation of the telic event pattern for the `PrepareTea` activity.

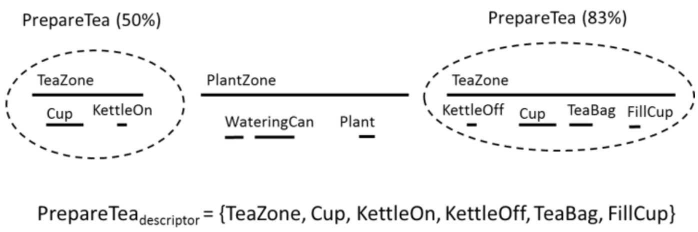

Fig. 4. Example observations and detected activities

8 Conclusions and Future Work

In this paper, we presented the ReDef framework for detecting interleaved activities in multi-sensor pervasive environments. The aim of the framework is to enrich existing activity recognition solutions that support the detection of sequential only activities with the ability to handle interleaved tasks. To this end, two lightweight ontology patterns have been defined to capture the concept of activity telicity. The semantics of these models is implemented by a set of defeasible rules, providing a context-aware decision making layer for aggregating interrupted activities into single activities.

ReDef has been integrated in an existing framework for ADL recognition, supporting the diagnosis of Alzheimer's disease. Preliminary results indicate that ReDef is able to correctly detect the start/end times of interleaved activities in the majority of the situations in our setting, failing though to handle cases where the telic events and contexts are not detected by the underlying monitoring framework.

The key directions that underpin our ongoing research involve the definition of additional patterns to capture more complex notions of activity telicity, e.g. taking into account the starting context of activities. Moreover, we are investigating a data-driven extension to our framework, using machine learning algorithms to automatically extract telic events for certain activities in order to support personalisation capabilities and adaptive services.

Acknowledgments. This work has been supported by the EU FP7 project Dem@Care: Dementia Ambient Care - Multi-Sensing Monitoring for Intelligent Remote Management and Decision Support under contract No. 288199.

References

1. Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., Patel-Schneider, P. F.: *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press (2003).
2. Bikakis, A., Antoniou, G.: Defeasible Contextual Reasoning with Arguments in Ambient Intelligence. *IEEE Trans. on Knowledge and Data Engineering* 22(11), pp. 1492-1506 (2010).
3. Bikakis, A., Hassapis, P., Antoniou, G.: Strategies for contextual reasoning with conflicts in ambient intelligence. *Knowledge and Information Systems* 27(1), pp. 45-84 (2011).
4. Buettner, M., Prasad, R., Philipose, M., Wetherall, D.: Recognizing daily activities with rfid-based sensors. *11th International Conference on Ubiquitous Computing*, pp. 51-60 (2009).
5. Chen, L., Nugent, C.D.: Ontology-based activity recognition in intelligent pervasive environments. *Int. Journal of Web Information Systems* 5(4), 410-430 (2009).
6. Ciaramella, A. Cimino, M. G. C. A., Marcelloni, F., Straccia, U.: Combining fuzzy logic and semantic web to enable situation-awareness in service recommendation. *21st International Conference on Database and Expert Systems Applications: Part I, DEXA'10*, Berlin, Heidelberg, pp. 31-45 (2010).
7. Cook, D.J., Augusto, J.C., Jakkula, V.R.: Ambient intelligence: Technologies, applications, and opportunities. *Perv. and Mobile Computing* 5(4), pp. 277-298 (2009).
8. Gangemi, A., Mika, P. Understanding the semantic web through descriptions and situations. *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE* (pp. 689-706). Springer Berlin Heidelberg (2003).
9. Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: OWL 2: The Next Step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web* 6(4), pp. 309-322 (2008).
10. Gu, T., Chen, S., Tao, X., Lu, J: An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data & Knowledge Engineering* 533-544 (2010).
11. Helaoui, R., Niepert, M., Stuckenschmidt, H.: Recognizing interleaved and concurrent activities using qualitative and quantitative temporal relationships. *Pervasive and Mobile Computing* 7(6), pp. 660-670 (2011).
12. Helaoui, R., Riboni, D., Stuckenschmidt, H.: A probabilistic ontological framework for the recognition of multilevel human activities. *2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 345-354 (2013).

13. Jekjantuk, N., Grner, G., Pan, J.: Modelling and reasoning in metamodelling enabled ontologies. *Knowledge Science, Engin. and Management*, 51-62 (2010).
14. Lam, H. P., Governatori, G. The Making of SPINdle. *2009 Int. Symposium on Rule Interchange and Applications (RuleML '09)*, Governatori, G., Hall, J., & Paschke, A. (Eds.), Springer-Verlag, Berlin, Heidelberg, p.p. 315-322 (2009).
15. Maher, M.: Propositional defeasible logic has linear complexity. Theory and Practice of Logic Programming, 1(6):691–711 (2001)
16. Meditskos, G., Kontopoulos, E., Kompatsiaris, I.: Knowledge-driven Activity Recognition and Segmentation Using Context Connections. *International Semantic Web Conference (ISWC)*, pp. 260-275, Riva del Garda, Trento, Italy, 19-23 October (2014).
17. Modayil, J., Bai, T., Kautz, H.: Improving the recognition of interleaved activities, *10th International Conference on Ubiquitous Computing*, pp. 40–43 (2008).
18. Motik, B., Cuenca Grau, B., Sattler, U.: Structured objects in OWL: representation and reasoning. *World Wide Web*, pp. 555-564 (2008).
19. Nute D.: Defeasible Reasoning, *Proc. 20th Int. Conference on Systems Science*, IEEE Press, pp. 470-477 (1987).
20. Okeyo, G., Chen, L., Wang, H., Sterritt, R.: Dynamic sensor data segmentation for real-time knowledge-driven activity recognition. *Pervasive and Mobile Computing 10*, pp. 155-172 (2014).
21. Okeyo, G., Chen, L., Wang, H.: An Agent-mediated Ontology-based Approach for Composite Activity Recognition in Smart Homes, *J. UCS* 19(17), pp. 2577-2597 (2013).
22. Patkos, T., Chrysakis, I., Bikakis, A., Plexousakis, D., Antoniou, G.: A reasoning framework for ambient intelligence. *6th Hellenic Conf. on Artificial Intelligence*. pp. 213-222. SETN'10, Springer-Verlag, Berlin, Heidelberg (2010).
23. Riboni, D., Pareschi, L., Radaelli, L., Bettini, C.: Is ontology-based activity recognition really effective? *Perv. Comp. and Commun*., pp. 427-431 (2011).
24. Rodrguez, N. D., Cullar, M. P., Lilius, J., Calvo-Flores M. D.: A fuzzy ontology for semantic modelling and recognition of human behaviour. *Knowledge-Based Systems*, 46-60, (2014).
25. Tiberghien, T., Mokhtari, M., Aloulou, H., Biswas, J.: Semantic reasoning in context-aware assistive environments to support ageing with dementia. *11th International Conference on The Semantic Web - Volume Part II, ISWC'12*, pp. 212-227 (2012).
26. Wessel, M, Luther, M., Wagner, M.: The difference a day makes - recognizing important events in daily context logs. *C&O:RR, volume 298 of CEUR Workshop Proceedings*. CEUR-WS.org, 2007
27. Ye, J., Stevenson, G., Dobson, S.: KCAR: A knowledge-driven approach for concurrent activity recognition. *Pervasive and Mobile Computing* 19, pp. 47-70 (2015).
28. Zaslavsky, A., Chakraborty, D.: Recognizing concurrent and interleaved activities in social interactions. *Dependable, Autonomic and Secure Computing (DASC)*, pp. 230-237 (2011).
29. Zhang, S., McCullagh, P., Nugent, C., Zheng, H.: An ontology-based context-aware approach for behaviour analysis. *Activity Recognition in Pervasive Intelligent Environments*, Vol. 4 of Atlantis Ambient and Pervasive Intelligence, Atlantis Press, 127-148 (2011).