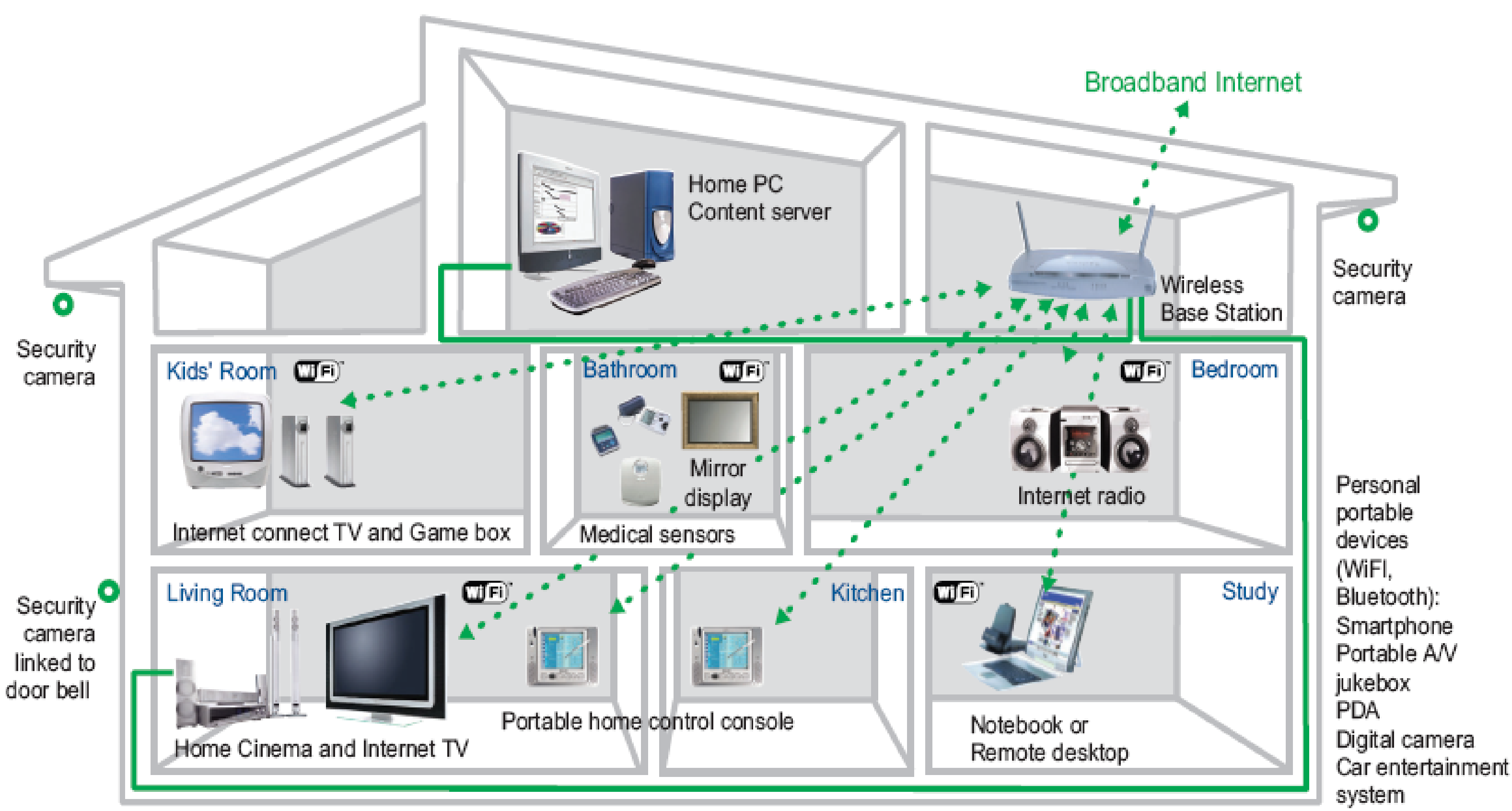
Rule-Based Activity Recognition in Ambient Intelligence

Grigoris Antoniou

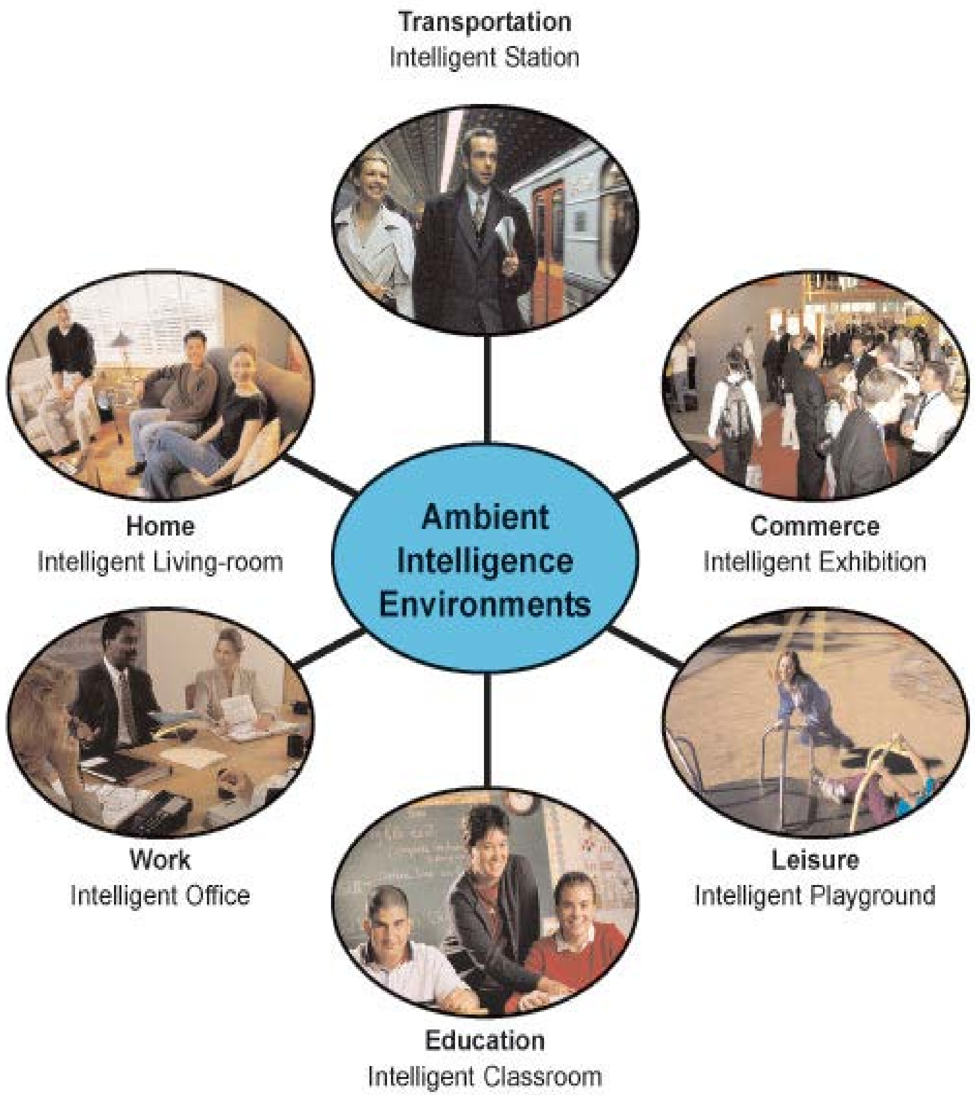
FORTH-ICS & University of Crete, Greece

Work done in collaboration with Ioannis Tsamardinos and Hrisi Filipaki

AmI: Sensor-Rich Collaborative Environments



Activity Recognition



- Derive high-level knowledge from low-level (sensor) input

- Potential application in Ambient Intelligence Environments:
  - Ambient Assisted Living (AAL)
  - Intelligent workspaces
  - Intelligent classrooms
  - Energy-efficient buildings

Overview

- **Problem Description**
- Related Work
- Reasoning System
- Experimental Results
- Conclusions

The Challenge

- Prior work identifies all occurrences of a specific type of activity
  - E.g. has the person fainted, did she fall?
- General-purpose activity recognition systems need to:
  - identify all user activities and their relations
  - report activities that are logically consistent
  - decide the level of detail (granularity) of the reported activities, depending on the context of use

Innovation (1/2)

- Logic and rule-based system that:
  - deals with noise and uncertainty
  - detects and resolves conflicts of the identified activities
  - reports logically consistent scenarios
  - takes preference parameters that adjust the the abstraction levels in the scenarios returned

Innovation (2/2)

- Computation of a complete picture versus query evaluation
  - Existing systems answer the question: “*Was the complex activity E occurring at time t?*”
  - Our system answers the question: “*Which complex activities have occurred in the given time interval?*”.
- Generic approach
  - Works for a variety of activities and settings
  - Works for a variety of input (various sensors, videos, ...)

Motivating Scenario

- Elderly person living in an AAL environment
- Patient’s nurse:
  - determine whether and when patient is taking his medication, and if he needs help
  - **detailed results**
- Patient’s doctor
  - considering the patient’s lifestyle (sleep patterns, amount of rest etc.)
  - **abstract results**

Overview

- Problem Description
- **Related Work**
- Reasoning System
- Experimental Results
- Conclusions

Related Work: Main Approaches

- Logic-based
- Probabilistic-based
- Combinations of both

Logic-based Approaches

- Artikis et al. - LTAR-EC: Event Calculus dialect implemented in Prolog [1]
- Dousson et al. - Chronicle Recognition System (CRS): a purely temporal reasoning system [2]
- Shet et al. - VidMAP: real time computer vision algorithms with logic programming to represent and recognize activities [9]

+ No training data needed
− Do not deal with missing events and noise
− Do not store the intervals of the recognized complex activities
− Do not handle conflict detection and detail control

Probabilistic Approaches

- Systems using Hidden Markov Models (HMMs) and their variations
  - Patterson et al.: HMMs to recognize interleaving activities based on sensor data from users morning routines [3].
  - Nguyen et al.: Hierarchical HMMs for recognizing single person indoor activities from movement trajectories extracted from camera data [4].
  - Oliver et al.: a multilayer representation of HMMs (LHMMs) to diagnose states of a user's activity based on real-time streams of video [5].
- Systems using Conditional Random Fields (CRFs) and their variations
  - Vail et al.: CRFs for activity recognition in multi-agent systems [6].
  - Liao et al.: Skip-Chain CRFs used to model interleaved activities [7].
  - Wu et al.: Factorial CRFs used to model concurrent activities [8].

+ Noise and uncertainty are handled well
− Require training data
− Do not handle conflict detection and detail control

Logic and Probabilistic Combinations

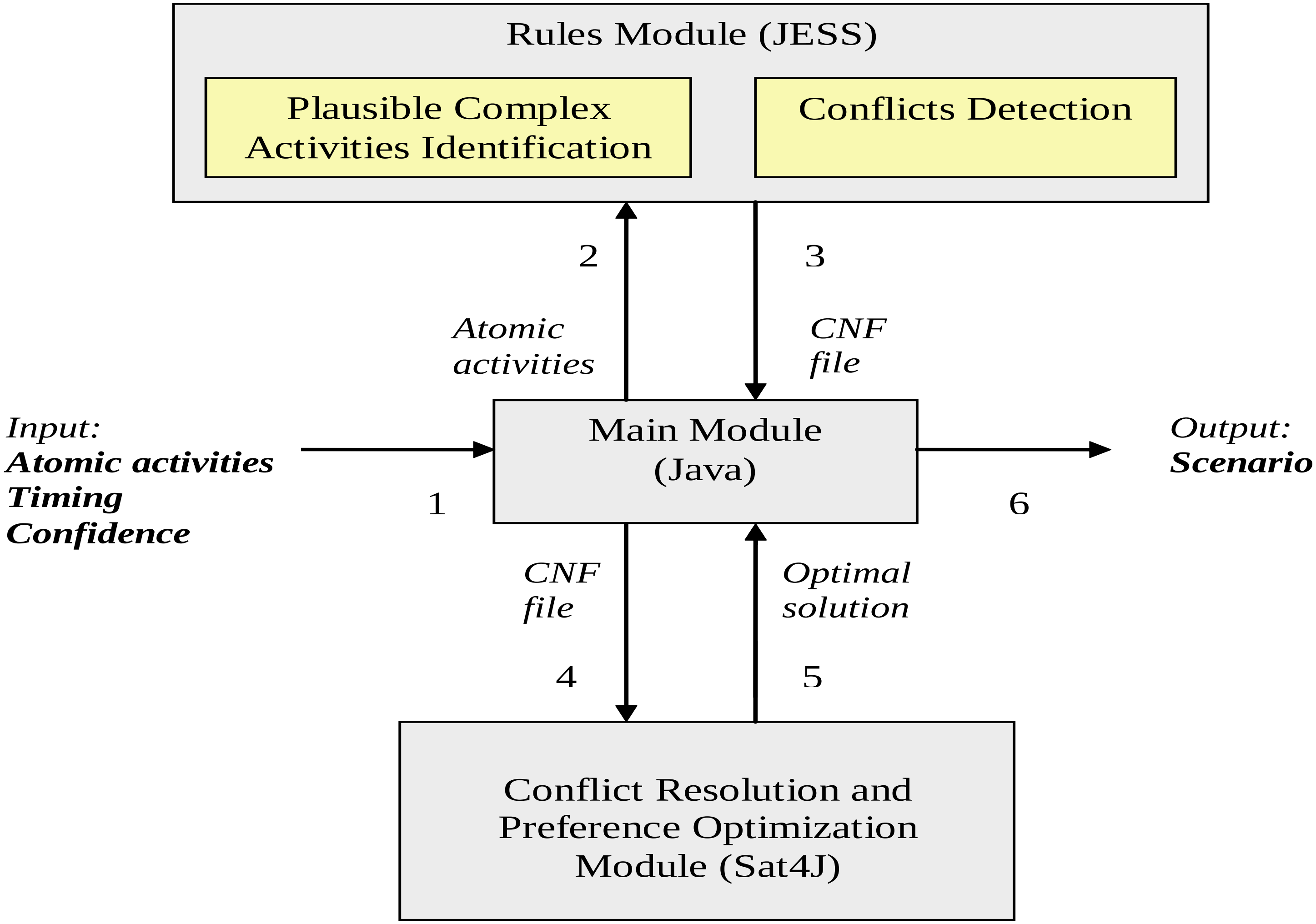
- Shet et al.: Prolog rules and a Bilattice framework for human detection.
- Hongeng et al.: Stochastic Finite Automaton and Bayesian methods for single- and multiple- actor activity recognition.
- Systems using Markov Logic Networks (MLNs) and their variations
  - Tran and Davis: MLNs to probabilistically infer activities in a parking lot (from video data).
  - Biswas et al.: Dynamic MLNs that groups fifteen first-order logic propositions applied to an office setting (from video data).
  - Helaoui et al.: MLNs for recognizing interleaved and concurrent activities incorporating input from sensors and common-sense background knowledge.

- (+) Noise and uncertainty are handled well
- (−) Require training data
- (−) Poor temporal reasoning - no rules for computing the intervals
- (−) Do not handle conflict detection and detail control

Overview

- Problem Description
- Related Work
- **Reasoning System**
- Experimental Results
- Conclusions

System Architecture



Step 1: Identification of Plausible Activity Occurrences

- *Activity instance* (or simply, activity): denoted by $E[\,t_1,t_2]\ _{cf}$
  - E: unique identifier of the activity
  - t_1: its start time
  - t_2: its end time
  - cf : the confidence value we have for this activity

- An atomic activity E is defined as an instantaneous activity:

$$E\left[t_1, t_2\right]_{cf} \text{ is atomic activity} \Leftrightarrow t_1 = t_2$$

- Complex activities are constructed recursively from the atomic and lower-level complex activities based on some event algebra operators over activity types. Activity recognition rules are implemented in Jess .

Operators (1/2)

- **Negation As Failure (not)**: used to derive not *E* from failure to derive *E*. The pattern is considered to match if a fact (or set of facts) which matches the pattern is not found.

- **Disjunction operator ($\vee$)**: at least one of the specified instances has to occur. Disjunction of two events E_1 and E_2 occurs when E_1 occurs or E_2 occurs.

- **Conjunction operator ($\wedge$)**: the specified event instances must occur at the same interval. Conjunction of two events E_1 and E_2 occurs when both E_1 and E_2 occur, irrespective of their order of occurrence.

Operators (2/2)

- **Optional-activity operator (optional):** an optional activity still allows the recognition of higher-level activities that may depend on it, with smaller confidence: an activity is still recognized, if flagged as optional, with *0* confidence even if it never occurred.

- **Sequence operator (;)**:the activity $(E_1 ; E_2)$ is recognized when E_1 and E_2 occur in this order. The activities have to follow each other within at most w time-units from each other. This precludes the situation the set is recognized from activities separated by an arbitrarily long time interval

- **Set operator (set):** the activity $set(E_1 , E_2)$ is recognized when both E_1 and E_2 occur in any order. The activities have to follow each other within at most w time-units from each other.

Examples of Complex Activity Types

- $$UserIsWatchingTv \leftarrow TurnOnTv \;\; ; \;\; set\begin{pmatrix} optional\,(ChangeTvChannels), \\ optional\,(ChangeTvVolume) \end{pmatrix}; \;\; TurnOffTv$$

- $$UserIsRelaxingAtHome \leftarrow set\begin{pmatrix} optional\;\,(UserIsRestingOnBed), \\ optional\;\,(UserIsWatchingTv), \\ optional\;\,(UserIsTalkingOnTelephone), \\ optional\;\,(UserIsWatchingSlideshow) \end{pmatrix}$$

Step 2: Conflict Detection – Simple Approach

- Pairs of activities that a user cannot perform at the same time
  - e.g. “User is relaxing at home” and “User is watching slideshow” (part of user’s work).
- Detect conflicts → define conflicting pairs of activity types, e.g., relaxing vs. working.
- This approach complicates knowledge engineering: whenever a new type is defined, all conflicting predefined types should be declared.

Conflict Detection - Our Approach

- Concept of activity resources: e.g. “chair”, “user’s attention”.
- For each activity type a list of activity resources is specified.
- Two complex activities are in conflict, if their time-intervals overlap and they use common resources, or are recognized based on activities that are in conflict.
- Implemented with Jess rules.

Step 3: Conflict Resolution – Simple Approach (1/2)

- B_i : propositional (binary) variable denoting whether a recognized activity E_i is selected in the final output.
- $(\neg B_i \vee \neg B_j)$: constraint on the propositional variables B_i, B_j when events E_i and E_j are conflicting (only one of them should be selected for the returned scenario).
- Resolving all conflicts is equivalent to solving a satisfiability problem (SAT) of the form:

$$(\neg B_k \vee \neg B_m) \wedge \ldots \wedge (\neg B_i \vee \neg B_j)$$

Conflict Resolution - Naïve Approach (2/2)

- Trivial solution: setting all B_i to false, thus not returning any activities and avoiding all conflicts.
- Desired solution: recognizing as many activities as possible, or even better, high-confidence activities that “explain” a large percentage of user’s time and atomic activities.

Conflict Resolution - Optimization (1/5)

- Convert to a **Weighted Partial MaxSAT** problem:
  - generalization of the SAT problem
  - *hard constraints*: clauses that specified must be satisfied
  - *soft constraints*: desirable to be satisfied.
    - weights are assigned → represent the penalty to falsify the clause
  - Optimal solution: assignment s.t. satisfies all the hard clauses, and the sum of the weights of the falsified soft clauses is minimal.
  - We used Sat4j, an open source library of SAT-solvers.

Conflict Resolution – Optimization (2/5)

- For each plausible activity E_i we define the following:
  - B_i *:* a binary variable denoting the selection of E_i in the output
  - $D(E_i)$: the temporal duration of E_i
  - $C(E_i)$: the confidence of E_i
  - $A(E_i)$: the number of atomic activities we used to recognize (explained-by) E_i

Conflict Resolution – Optimization (3/5)

- For each conflict between E_i and E_j we create the clause **$(\neg B_i \vee \neg B_j)$ as a hard constraint**.

- For each activity E_i we create the **singleton clause B_i as a soft constraint**. The weight given to B_i is:

$$w_i = a \cdot D(E_i) + b \cdot C(E_i) + c \cdot A(E_i)$$

where $a, b, c > 0$ are preference parameters.

Conflict Resolution – Optimization (4/5)

- If $E_1,\dots,E_n$ are all the plausible activities we have recognized, our Weighted Partial MaxSAT problem is going to have the form:

$$B_1^{w_1} \wedge \ldots \wedge B_n^{w_n} \wedge \left(\neg B_k \vee \neg B_m\right) \wedge \ldots \wedge \left(\neg B_i \vee \neg B_j\right)$$

where the superscripts of B_i denote the corresponding weight.

- Thus, the Weighted Partial MaxSAT solves the following optimization problem:

$$\max_{B_1 \cdots B_n} \sum_{i=1}^{n} w_i \cdot B_i$$

$$\text{s.t. all conflicts are resolved}$$

Conflict Resolution – Optimization (5/5)

- So with the above optimization problem we want to get a set of recognized complex activities that are:
  - as many as possible
  - with high-confidence
  - “explain” a large percentage of user’s time
  - “explain” a large percentage of detected atomic activities.

Overview

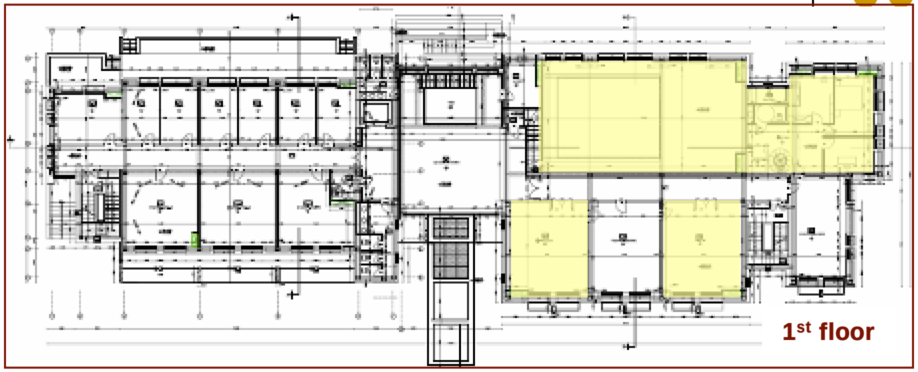
- Problem Description
- Related Work
- Reasoning System
- **Experimental Results**
- Conclusions

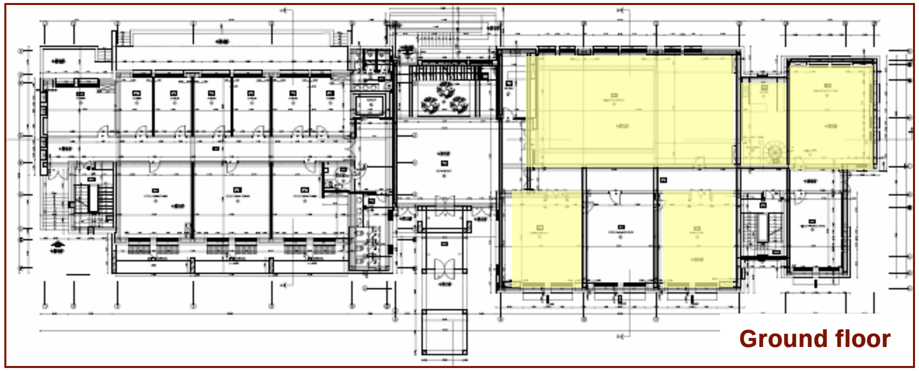
AmI Sandbox (1/2)

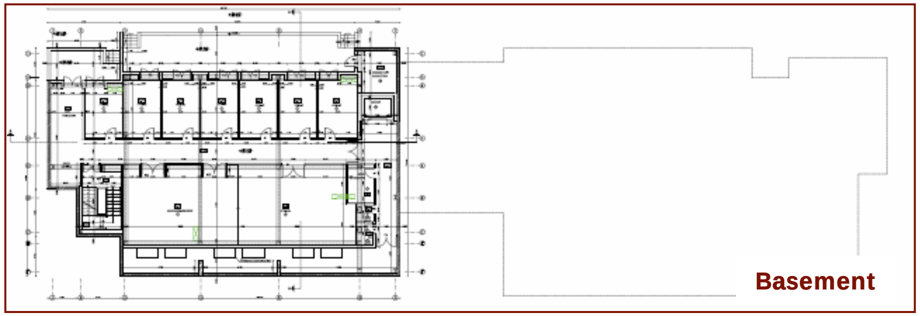
- An experimental space within ICS-FORTH (~ 100m^2).
- Several AmI technologies and applications are installed, integrated and demonstrated, and multiple ideas and solutions are cooperatively developed, studied and tested.

AmI Sandbox (2/2)

AmI Facility – Blueprints



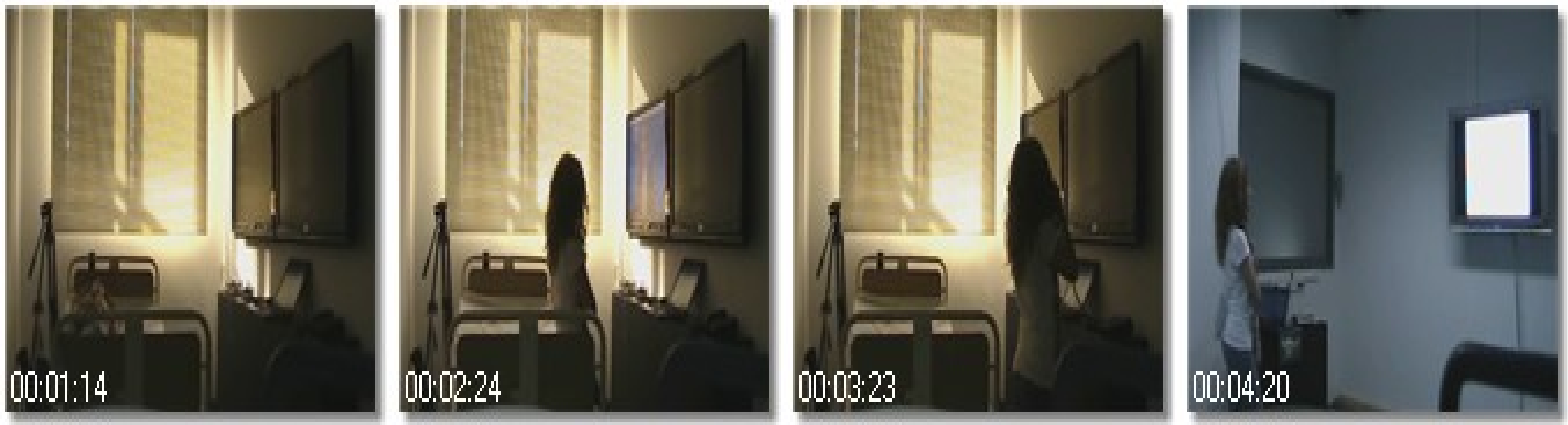




Demo Implementation in AmI Sandbox (1/4)

- Implemented the recognition system and integrated it within AmI Sandbox.
- User was given the instructions to enter the facility and perform a set of atomic activities. The user was not given any other instructions or guidance.
- We ran the system with the atomic activities detected from the facility and it correctly identified all user activities.

Demo Implementation in AmI Sandbox (2/4)



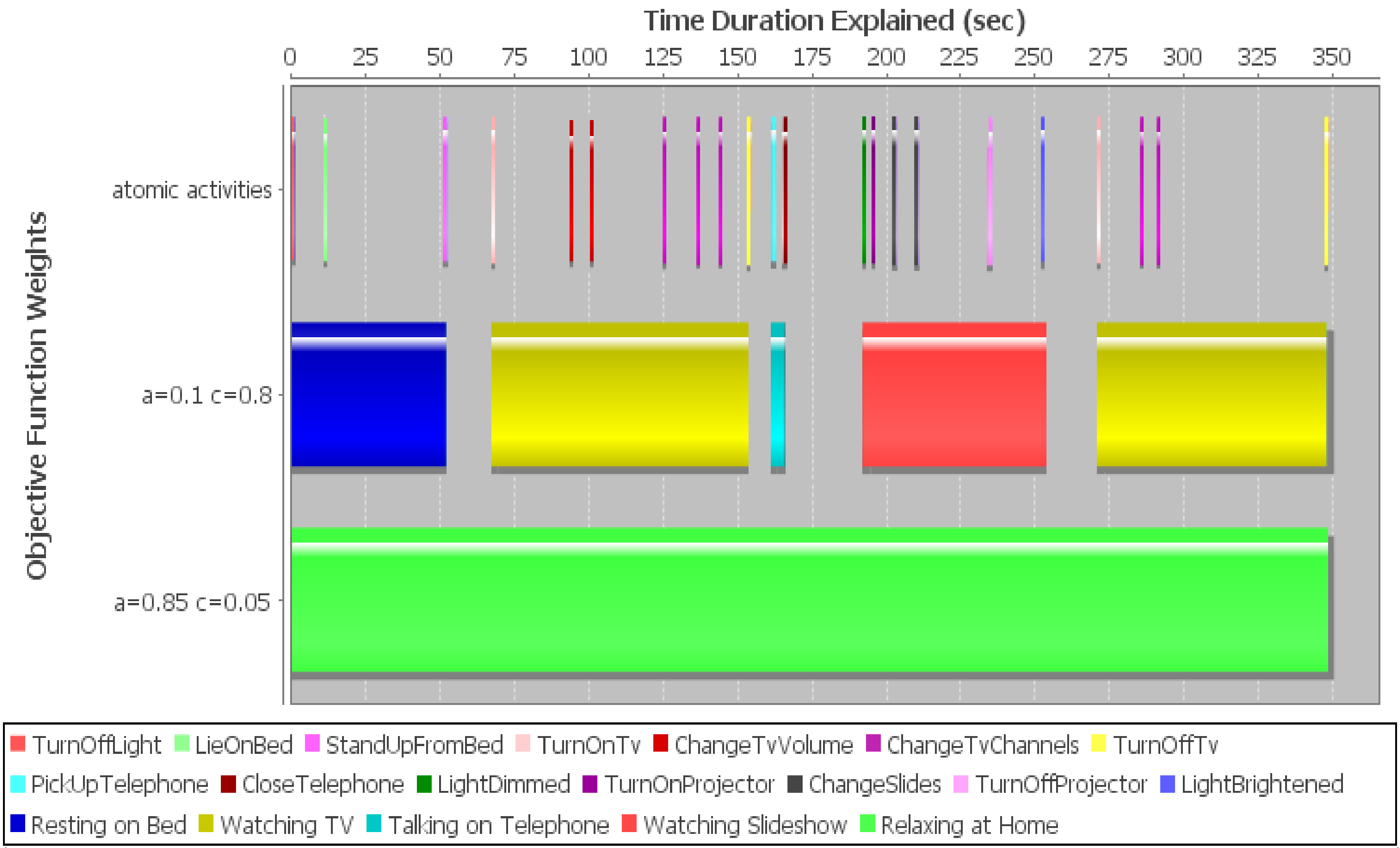
Screenshots from demonstration. From left to right user is:

- resting on bed
- watching TV
- talking on telephone
- watching slideshow in a different room

Demo Implementation in AmI Sandbox (3/4)

- Demonstrate the system's ability to report activities at different levels of detail → we ran the recognition algorithm with various settings of the preference parameters:
  - **(*a=0.1, b=0.1, c=0.8*)**
    - higher preference to scenarios that explain more atomic activities, i.e., detailed scenarios.
    - returned scenario: "Resting on bed", "Watching TV", "Talking on Telephone", "Watching Slideshow", "Watching TV".
  - **(*a=0.85, b=0.1, c=0.05*)**
    - higher preference to scenarios with activities of longer temporal duration, even if some atomic activities are not explained.
    - Returned scenario: "User is relaxing at home".
    - "Talking on the phone" was ignored due to short duration.

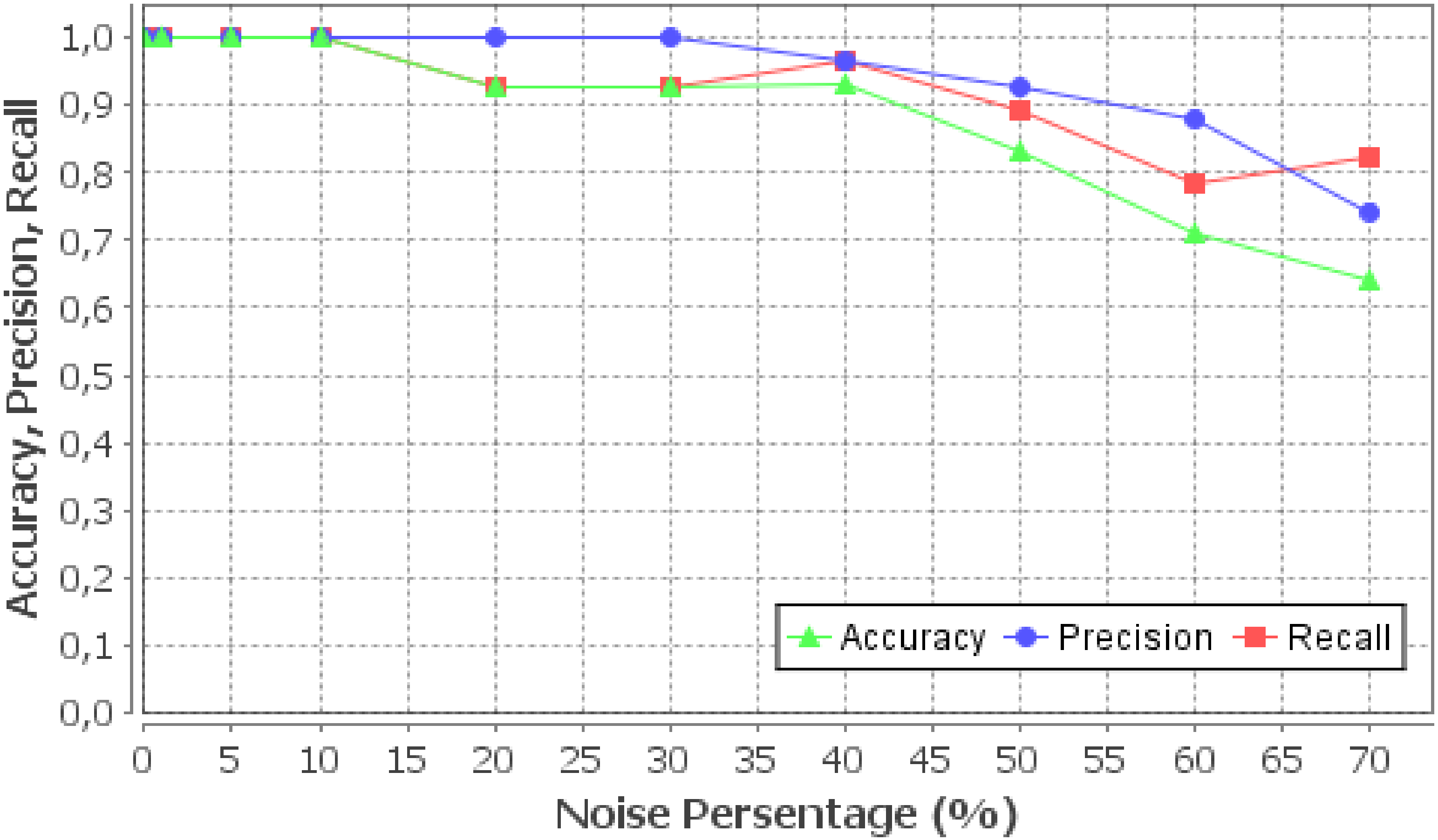
Demo Implementation in AmI Sandbox (4/4)



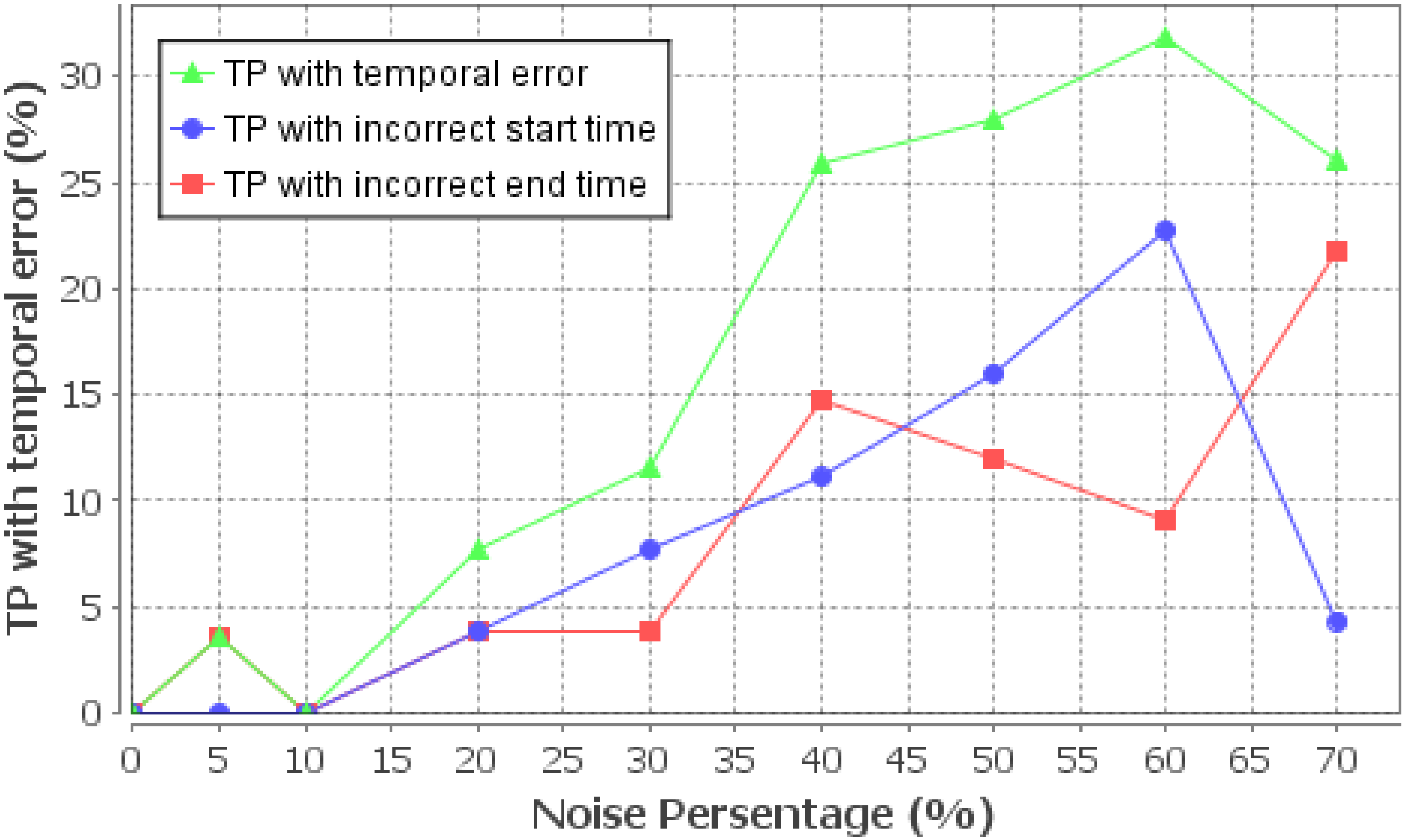
Simulation Studies (1/3)

- Evaluate the robustness of the system
- Running 40 datasets containing *1592 atomic activities*.
- Generated the datasets and then added randomly different percentages of noise. For a given level *l* of noise
  - $l \times 90\%$ random atomic activities are inserted in the dataset (random activities)
  - $l \times 10\%$ of total atomic activities in the datasets is deleted (lost activities)

Simulation Studies (2/3)



Simulation Studies (3/3)



Overview

- Problem Description
- Related Work
- Reasoning System
- Experimental Results
- **Conclusions**

Conclusion

- Rule-based activity recognition system for hierarchically-organized complex events that returns only logically consistent sets of activities (scenarios).
- Fully implemented scenario in AmI environment demonstrated that
  - the system is efficiently working
  - the level of detail can be easily adjusted according to our preferences.
- System handles noise and uncertainty, with the use of optional activities and confidence factors in its facts
- Experimental results have shown that the system is robust to noise.

Future Work

- Improve the system's performance
  - With systematic ways of pruning the search space
  - With heuristics that may sacrifice optimal solution to achieve good performance

- Optimization techniques presented in this work could accommodate other types of preferences and be generalized to other settings.
  - include more preference factors in our system, for controlling the abstraction level in the scenarios returned.

- More extensive experiments in order to work out the relative merits and weaknesses compared to other approaches.

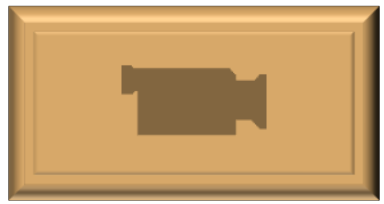
Future Work – Rules and AI

- Ambient intelligence is a rich testbed for rule technology and a variety of other AI methods
  - Distributed rule-based reasoning about context
  - Complex event processing
  - Reasoning about action
  - Multi-agent coordination

AmI Demo Video

- A video of the demo is freely available:
  https://rapidshare.com/files/1954778061/AmI_Demo.rar
- Some machines (e.g. TV) have to be operated through software for some atomic activities (e.g. TurnOnTV) to be registered.

AmI Demo Video



- A video of the demo is freely available: https://rapidshare.com/files/1954778061/AmI_Demo.rar
- Some machines (e.g. TV) have to be operated through software for some atomic activities (e.g. TurnOnTV) to be registered.

Questions

Thank you!

References (1/2)

1. Artikis, A., Sergot, M., Paliouras, G.: A Logic Programming Approach to Activity Recognition. In: Proc. of ACM International Workshop on Events in Multimedia (2010)
2. Dousson, C., Maigat, P.L.: Chronicle recognition improvement using temporal focusing and hierarchisation. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp. 324-329. (2007)
3. Patterson, D.J., Fox, D., Kautz, H., Philipose, M.: Finegrained activity recognition by aggregating abstract object usage. In ISWC ’05: Proceedings of the Ninth IEEE International Symposium on Wearable Computers.: IEEE Computer Society. Washington, DC, USA (2005)
4. Nguyen, N.T., Phung D.Q., Venkatesh, S., Bui, H.H.: Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 955–960. San Diego (2005)
5. Oliver, N., Horvitz, E., Garg, A.: Layered representations for human activity recognition. In: Computer Vision and Image Understanding Journal, vol. 96:2, pp.163–180 (2004)
6. Vail, D.L., Veloso, M.M, Lafferty, J.D.: Conditional random fields for activity recognition. In: International Conference on Autonomous Agents and Multi-agent Systems (AAMAS). (2007)
7. Liao, L., Fox, D., Kautz, H.: Hierarchical Conditional Random Fields for GPS based Activity Recognition. In: Robotics Research the Twelfth International Symposium (ISRR-05). Springer-Verlag (2006)
8. Wu, T., Lian, C., Hsu, J.Y.: Joint recognition of multiple concurrent activities using factorial conditional random fields. In: Proceedings of AAAI Workshop on Plan, Activity, and Intent Recognition. California (2007)

References (2/2)

9. Shet, V., Harwood, D., Davis, L.: VidMAP: video monitoring of activity with Prolog. In: Advanced Video and Signal Based Surveillance IEEE. (2005)
10. Shet, V., Neumann, J., Ramesh, V., Davis, L.: Bilattice-based logical reasoning for human detection. In: Proc. Of IEEE Computer Vision and Pattern Recognition (CVPR) (2007)
11. Tran, S.D., Davis, L.S.: Event modeling and recognition using Markov logic networks. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Computer Vision. LNCS, vol. 5303, pp. 610-623. Springer, Heidelberg (2008)
12. Biswas, R., Thrun, S., Fujimura, K.: Recognizing activities with multiple cues. In: Elgammal, A., Rosenhahn, B., Klette, R. (eds.) Human Motion. LNCS, vol. 4814, pp. 255-270. Springer, Heidelberg (2007)
13. Helaoui, R., Niepert, M., Stuckenschmidt, H.: Recognizing Interleaved and Concurrent Activities: A Statistical-Relational Approach. In: Proceedings of the 9th Annual IEEE International Conference on Pervasive Computing and Communications (2011)
14. Hongeng, S., Nevatia, R., Brémond, F.: Video-based event recognition: Activity representation and probabilistic recognition methods. In: Comput. Vis. Image Understand., vol. 96:2, pp. 129–162. (2004)
15. JESS , The Rule Engine for the Java Platform, http://www.jessrules.com/
16. Le Berre, D., Parrain, A.: The Sat4j library, release 2.2. Journal on Satisfiability, Boolean Modeling and Computation (JSAT). 7, 59-64 (2010)

Artikis et al. LTAR-EC

- The Event Calculus (EC) first presented by Kowalski and Sergot in 1986 is a set of first-order predicate calculus, including temporal formalism, for representing and reasoning about events and their effects.
- Artikis et al. developed LTAR-EC (event calculus for long-term activity recognition), an activity recognition system consisting of an Event Calculus dialect implemented in Prolog.
- The input of the system is a set of time-stamped short-term activities (atomic activities in our context) detected on video frames e.g. “walking”, “inactive”.
- The output of the system is a set of recognized long-term activities (complex activities in our context), which are predefined temporal combinations of short-term activities e.g. “fighting”, “leaving an object”.
- LTAR-EC does not currently store the outcome of query computation, i.e. the intervals of the recognised activities.

Shet et al. VidMAP

- Visual surveillance system that combines real time computer vision algorithms with logic programming to represent and recognize activities involving interactions amongst people, packages and the environments through which they move [9].

- The higher level Prolog based reasoning engine uses these facts in conjunction with predefined rules to recognize various activities in the input video streams.

- They answer specific queries about events that have already transpired in the archived video.

- Positive and negative information from different sources, as well as uncertainties from detections and logical rules, are integrated within the bilattice framework in [10].