Ontology-based Realtime Activity Monitoring Using Beam Search

Wilfried Bohlken, Bernd Neumann, Lothar Hotz, Patrick Koopmann

FB Informatik, Universität Hamburg, Germany
{bohlken, neumann, koopmann}@informatik.uni-hamburg.de
hotz@hitec-hh.de

Abstract. In this contribution we present a realtime activity monitoring system, called SCENIOR (SCEne Interpretation with Ontology-based Rules) with several innovative features. Activity concepts are defined in an ontology using OWL, extended by SWRL rules for the temporal structure, and are automatically transformed into a high-level scene interpretation system based on JESS rules. Interpretation goals are transformed into hierarchical hypotheses structures associated with constraints and embedded in a probabilistic scene model. The incremental interpretation process is organised as a Beam Search with multiple parallel interpretation threads. At each step, a context-dependent probabilistic rating is computed for each partial interpretation reflecting the probability of that interpretation to reach completion. Low-rated threads are discarded depending on the beam width. Fully instantiated hypotheses may be used as input for higher-level hypotheses, thus realising a doubly hierarchical recognition process. Missing evidence may be "hallucinated" depending on the context. The system has been evaluated with real-life data of aircraft service activities.

1 Introduction

This paper is about realtime monitoring of object behaviour in aircraft servicing scenes, such as arrival preparation, unloading, tanking and others, based on video streams from several cameras[1]. The focus is on high-level interpretation of object tracks extracted from the video data. The term "high-level interpretation" denotes meaning assignment above the level of individually recognised objects, typically involving temporal and spatial relations between several objects and qualitative behaviour descriptions corresponding to concepts used by humans. For aircraft servicing, interpretation has the goal to recognise the various servicing activities at the apron position of an aircraft, beginning with arrival preparation, passenger disembarking via a passenger bridge, unloading and loading operations involving several kinds of vehicles, refuelling, catering and other activities. Our work can be seen as an alternative to an earlier approach reported in [1], which does not possess the innovative features reported here.

It is well established that high-level vision is essentially an abductive task with interpretations providing an "explanation" for evidence [2-4]. In general, there may be

[1] This work was partially supported by EC Grant 214975, Project Co-Friend.

several possible explanations even for perfect evidence, and still more if evidence is incomplete or uncertain. Hence any scene interpretation system must deal with multiple solutions. One goal of this paper is to show how a probabilistic preference measure can be combined with an abductive framework to single out the most probable solution from a large set of logically possible alternatives. Different from Markov Logic Networks which have been recently proposed for scene interpretation [5] we combine our logical framework with Bayesian Compositional Hierarchies (BCHs) specifically developed for hierarchical scene models [6].

A second goal is to present an approach where a scene interpretation system is automatically generated from a conceptual knowledge base represented in the standardised ontology language OWL-DL. This facilitates the interaction with reasoners (such as Pellet or RacerPro) and the integration with other knowledge bases.

A third innovative contribution of this paper is a recognition strategy capable of handling highly contaminated evidence and in consequence a large number of alternative interpretations. This is mainly achieved by maintaining up to 100 alternative interpretation threads in a Beam Search [8]. Results show that a preference measure can be used effectively to prune the beam at intermediate stages and to select the best-rating from several final interpretations.

2 Behaviour Modelling

In this section we describe the representation of activity models in a formal ontology. Our main concern is the specification of aggregate models adequate for the activities of the aircraft servicing domain, but also to exemplify generic structures for other domains.

In a nutshell, an aggregate is a conceptual structure consisting of

- a specification of aggregate properties,
- a specification of parts, and
- a specification of constraints between parts.

To illustrate aggregate specifications, consider the aggregate `Unloading-Loading-AFT` as an example. It consists of three partial activities as shown in Fig. 1, which must meet certain constraints to combine to an unloading or loading activity.

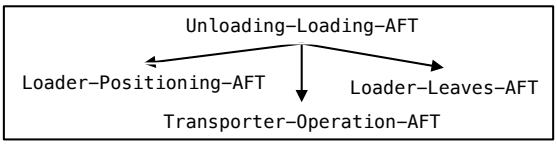

Fig. 1. Part structure of the aggregate `Unloading-Loading-AFT`

First, temporal constraints must be met: The loader must be placed at the aircraft before any transporter operations can take place, and must leave after completion of these operations. Similarly, spatial constraints must be met, in our domain realised by fixed zones defined for specific servicing activities (e.g. the `AFT-Loading-Zone`). Finally, the same physical object occurring in separate parts of an aggregate must be

referred to by an identity constraint. Please note that the graphical order of aggregate parts shown in Figs. 1 and 2 does nor imply a temporal order.

As mentioned in the introduction, we have chosen the web ontology language OWL-DL for defining aggregates and related concepts. OWL-DL is a standardised formalism with clear logical foundations and provides the chance for a smooth integration with large-scale knowledge representation and reasoning. Furthermore, the object-centered style of concept definitions in OWL and its support by mature editors such as Protégé[2] promise transparency and scalability. Simple constraints can be represented with SWRL, the Semantic Web Rule Language, albeit not very elegantly.

In OWL-DL, the aggregate `Unloading-Loading-AFT` is defined as follows:

```
Unloading-Loading-AFT ⊑ Composite-Event ⊓
has-part1 exactly 1 Loader-Positioning-AFT ⊓
has-part2 exactly 1 Transporter-Operation-AFT ⊓
has-part3 exactly 1 Loader-Leaves-AFT
```

The left-hand side implies the right-hand side, corresponding to an abductive reasoning framework. In our definition, the aggregate may name only a single taxonomical parent because of the intended mapping to single-inheritance Java templates. Furthermore, the aggregate must have exactly one part for each hasPartRole. While the DL syntax would allow number restrictions for optional or multiple parts, we found it useful to have different aggregate names for different part configurations and a distinct hasPartRole for each part to simplify the definition of conceptual constraints.

Our aircraft servicing domain is described by 41 aggregates forming a compositional hierarchy. The leaves are primitive aggregates with no parts, such as `Loader-Leaves-AFT`. They are expected to be instantiated by evidence from low-level image analysis. In addition to the compositional hierarchy, all objects, including aggregates, are embedded in a taxonomical hierarchy which is automatically maintained by OWL-DL. Thus, all activities can be related to a general activity concept and inherit roles such as `has-agent`, `has-start-time`, and `has-finish-time`.

Fig. 2 gives an overview of the main components of aircraft servicing activity concepts. Besides the logical structure, we provide a hierarchical probabilistic model as a preference measure for rating alternative interpretations [6]. In our domain, the model is confined to the temporal properties of activities, i.e. durations and temporal relations between activities, which are represented as Gaussian distributions with the range -2σ .. 2σ corresponding to crisp temporal constraints. Using this model, the probabilities of partial interpretations can be determined and used to control the Beam Search. Unfortunately, OWL-DL and its approved extensions do not offer an efficient way for representing probabilities, so the probabilistic model is kept in a separate database.

Our approach to activity representation can be summarised as follows:

- The main conceptual units are aggregates specifying the decomposition of activities into subactivities and constraints between the components.
- The representation language for the logical structure is the standardised language OWL-DL which offers integration with high-level knowledge bases and reasoning services, e.g. consistency checking.

[2] http://protege.stanford.edu/

- A hierarchical probabilistic model is provided as a preference measure for temporal aggregate properties.

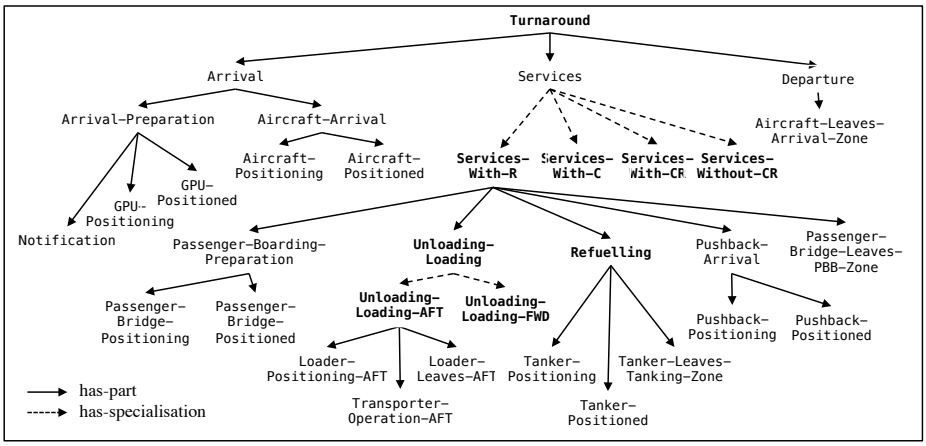

Fig. 2. Activity concepts for aircraft servicing

3 Initialising the Scene Interpretation System from the Ontology

In this section we describe the scene interpretation system SCENIOR, beginning with an overview. In Subsection 3.2 we describe the generation of rules for rule-based scene interpretation and the generation of hypotheses templates as interpretation goals. The interpretation process itself is described in Subsection 3.3.

3.1 System Overview

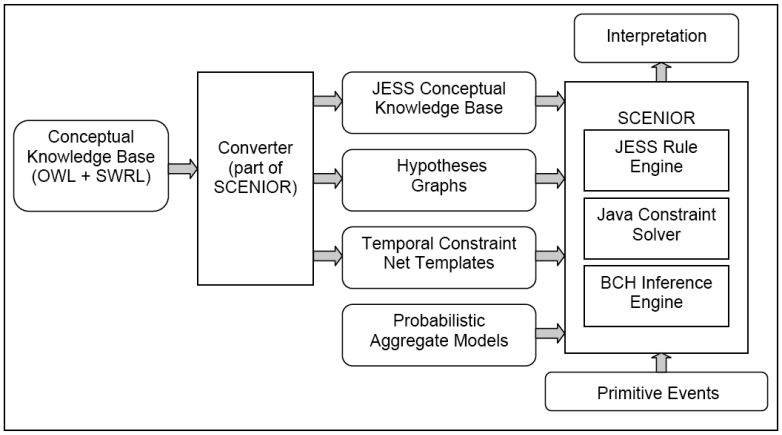

Fig.. 3. Main components of the scene interpretation system SCENIOR

Fig. 3 shows the architecture of the interpretation system SCENIOR. In the initialisation phase of the system, the conceptual knowledge base, represented in OWL-DL and SWRL, is converted into a JESS conceptual knowledge base, with rules for both bottom-up and top-down processing. Furthermore, hypotheses graphs are created corresponding to submodels of the compositional hierarchy, providing intermediate goals for the interpretation process. The temporal constraints defined with SWRL rules are translated into temporal constraint nets (TCNs) which maintain constraint consistency as the scene evolves. The interpretation process is organised as a Beam Search to accommodate alternative interpretations. A probabilistic scene model, realised as a Bayesian Compositional Hierarchy (BCH), provides a preference measure. For the sake of compactness, TCN and BCH are not described in detail in this paper, see [9] and [6].

3.2 Rule Generation from the Ontology

As shown in [7], scene interpretation can be viewed as a search problem in the space of possible interpretations defined by taxonomical and compositional relations and controlled by constraints. Four kinds of interpretation steps are required to navigate in interpretation space and construct interpretations:

- Aggregate instantiation (moving up a compositional hierarchy)
- Aggregate expansion (moving down a compositional hierarchy)
- Instance specialisation (moving down a taxonomical hierarchy)
- Instance merging (unifying instances obtained separately)

In our framework we create rules for the first three steps, together with some supporting rules. The step "instance merging" is dispensable with the use of hypotheses graphs and parallel search.

Submodels and Hypotheses Graphs. Usually, many models have to be considered in a scene interpretation task. To cope with model variants and to avoid redundancies, we define *submodels* which may be part of several alternative models and are treated as interpretation subgoals (e.g. `Refuelling`). After instantiation, they can be used as "higher-level evidence" for other aggregates (e.g. various kinds of services).

Submodels (marked as `context-free` in the conceptual knowledge base) give rise to hypotheses graphs. Formally, they represent the partonomical structure of a submodel and the equality constraints described with SWRL rules. Their main function is to provide coherent expectations about possible activities. During interpretation hypotheses graphs can be used to "hallucinate" missing evidence and thus continue a promising interpretation thread.

Rules. During the initialisation process, the following interpretation rules are created fully automatically from the ontology:

- *Evidence-assignment rules* assign evidence provided by lower-level processing to a leaf of a hypotheses graph. The premise of the rule addresses a template created for each aggregate (referred to as `template-x` below).

- *Aggregate-instantiation rules* instantiate a hypothesised aggregate (status `hypothesised`) if all its parts are instantiated or hallucinated. This is a bottom-up step in the compositional hierarchy and the backbone for the scene interpretation process.
- *Specialisation rules* refine an instance to a more specialised instance. This can happen if more information becomes available as the scene evolves (for example, `Vehicle-Inside-Zone` may be specialised to `Tanker-Inside-Zone`).
- *Aggregate-expansion rules* instantiate part of an aggregate if the aggregate itself is instantiated or hallucinated. A separate rule is created for every part of the aggregate. This is a top-down step in the compositional hierarchy. The rule will be invoked if a fact has not been asserted bottom-up but by other means, e.g. by common-sense reasoning (so far this is only rudimentary realised by the hallucination mechanism).

A simplified generic patterns for the evidence-assignement rule is given below, the other rules are defined in a similar way.

```
(defrule aggregate-x-ea-rule
        ?e-id <- (template-x (name ?e)(status evidence))
        ?h-id <- (template-x (name ?h)(status ?status_1))
                (test (or (eq ?status_1 hypothesised)
                          (eq ?status_1 hallucinated)))
        ;;check temporal constraints
=>
        (modify ?e-id (status assigned))
        (modify ?h-id (status instantiated))
        ;;update temporal constraint net)
```

3.3 Interpretation Process

In the initialisation phase of the system, a separate thread is created for each submodel. Each thread has its own independent JESS engine, initialised with all rules and the hypotheses graph corresponding to this submodel.

Now the system is ready to start the interpretation process. It receives primitive events as input and feeds these as working memory elements to every alive rule engine (in the beginning, these are the initialised interpretation threads). Then the rules are applied, eventually leading to instantiated aggregates. These may in turn provide input for higher-level aggregates. If there is more than one activation for an evidence-assignment rule within one thread (i.e. if multiple evidence assignments are possible), this thread is cloned into several threads, one for each possible assignment. A newly created thread is an exact copy of the original thread. This way, a search tree is established which examines all interpretation possibilities in parallel.

So far, we have not yet discussed how to deal with noise, which can either occur in terms of activities not modelled in the ontology, or due to errors of low-level processing. Various kinds of vehicles not taking part in a service or performing some unknown task enter and leave the servicing area throughout a turnaround. Also, low-level processing in our application is difficult and not at all perfect, hence strange events not corresponding to any real-world activities are delivered as input to SCENIOR. Since there is no way to distinguish correct evidence from noise, as long

as both satisfy the constraints, SCENIOR follows both interpretations in parallel, expanding the search tree at each step.

SCENIOR can process in real-time up to ca. 100 threads in parallel on an ordinary PC. Our experiments with airport activities showed that this maximal number of interpretation threads is normally reached while recognising a complete turnaround (see Section 4). At this point, the rating provided by the BCH comes into play and all lowest-rated threads in excess of the maximal beam width are discarded.

Finally, upon termination of the input data stream, all complete turnaround interpretations are ranked using the BCH, and the highest-ranking interpretation is delivered as the result.

4 Experimental Results and Evaluation

In this section we show results of SCENIOR obtained for concrete turnaround scenes at Blagnac Airport in Toulouse. We first illustrate the effects of context-dependent ratings. We then provide a performance evaluation of SCENIOR for 20 turnarounds. The results are explained by the noise statistics of the data which show that the correct interpretation will not always receive the highest rating.

4.1 Illustration of probabilistic rating

We now describe the initial phase of a concrete scene interpretation task to demonstrate the effect of the ranking provided by the BCH in a Beam Search. The input data have been obtained from one of the 60 turnarounds by low-level processing of project partners in France and England.

To rate interpretations in this experiment, the probability density of clutter has been set to 0.01 which is less than the typical probability of a regular piece of evidence for a turnaround. Note that the probability density is taken to measure the "probability" of an event. A small constant factor Δt for a time span, over which a density must be integrated, is omitted for clarity. Since the ratings are naturally decreasing with each step and may reach very small numbers, the natural logarithm of a probability is taken, resulting in negative ratings. The primitive events used here belong to an ontology version different from the one presented in Section 2.

In the scene interpreted in this experiment, an `Airplane-Enters-ERA` event has been generated erroneously by low-level processing for a tanker crossing the ERA (Entrance Restricted Area) shortly before the arrival of the airplane. Fig. 4 left shows the corresponding video frame taken by one of the eight cameras with the crossing tanker in the far background. Two threads are generated, Thread A interpreting this evidence as part of an `Arrival`, the Thread B as clutter. Later on, the true aircraft arrives (Fig. 4 right), generating an `Airplane-Enters-ERA` event in the Thread B and a clutter event in a new third thread.

The ratings for the partial interpretations of both alternatives are shown in Table 1. Interpretation A is the erroneous and Interpretation B is the correct one. Initially, the arrival of the GPU sets a context where a vehicle is expected to enter the ERA, hence

the crossing tanker is a candidate. But as soon as the true airplane enters, an alternative arises and is favoured because the probabilistic model expects an `Airplane-Enters-ERA` event 8 minutes after `GPU-Enters-GPU-Zone`, and the airplane's arrival is closer to that estimate than the tanker's. Note that clutter events not assigned to either of the two interpretations are not shown in the table.

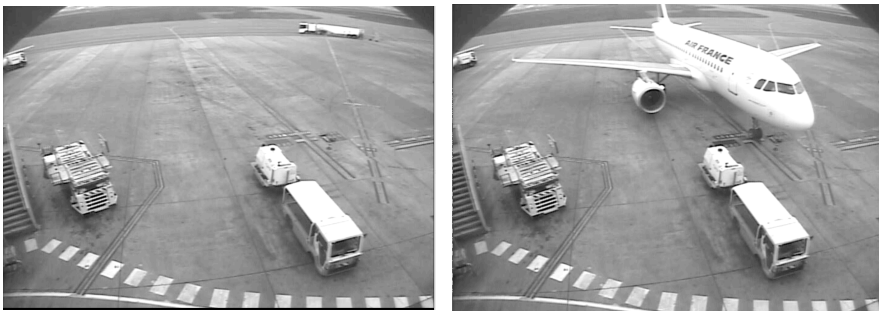

Fig. 4. Snapshots of the ERA (Entrance Restricted Area) after completing Arrival-Preparation. The GPU (Ground Power Unit) is in place. The tanker crossing the ERA in the background (left) causes an erroneous interpretation thread (see text).

Table 1. Initial ratings of the two alternative interpretations

e1 = mobile-inside-zone-86
e2 = mobile-stopped-90
e3 = mobile-inside-zone-131
e4 = mobile-inside-zone-155
est = estimated event

Evidence	Time	Interpretation A	Ranking A	Interpretation B	Ranking B
e1	17:10:31	GPU-Enters-GPU-Zone	0	GPU-Enters-GPU-Zone	0
e2	17:10:32	GPU-Stopped-In...	-2,16	GPU-Stopped-In...	-2,16
e3	17:13:31	Airplane-Enters-ERA	-5,32	Clutter	-2,16
e4	17:20:35	Clutter	-5,32	Airplane-Enters-ERA	-5,09
est	≥17:13:35	Airplane-Stopped...	-6,24		
est	≥17:13:35	Stop-Beacon	-7,71		
est	≥17:20:35			Airplane-Stopped...	-6,01
est	≥17:28:35			Stop-Beacon	-7,48

The table also includes the estimated times of the expected next events `Airplane-Stopped-Inside-ERA` and `Stop-Beacon` together with the expected ratings for the competing interpretations. Note that estimated time windows may begin earlier than the actual time, allowing for hallucinated events in the past. Considering that `Stop-Beacon` will occur after the true aircraft arrival and not at the time expected in Interpretation A, the rating of this interpretation will surely be much lower than the estimated value, further increasing the distance between the right and the wrong interpretation.

The performance of SCENIOR was evaluated for 20 annotated turnarounds, with primitive events provided by low-level image analysis of the project partners. The

ontology and the probabilistic model were derived from 32 other turnarounds. Because of the noisy input data, it was necessary to interpret each evidence both as belonging to a turnaround (given that the constraints were satisfied) and as clutter. 17 of the 20 turnarounds resulted in complete interpretations. This was facilitated by Special Vision Tasks with controlled cameras for three crucial events and by "hallucinations" for missing evidence in certain contexts. The three problematic sequences were highly irregular and did not match the conceptual model (e.g. GPU arrival after aircraft arrival). SCENIOR showed a reliable system performance with up to 100 parallel threads (limited by a preset beam width) for partial alternative interpretations, as shown in Fig. 5. It can be seen that alltogether more than 1000 partial interpretations have been initialised, many caused by the context-free submodels which posed interpretation goals throughout the sequence.

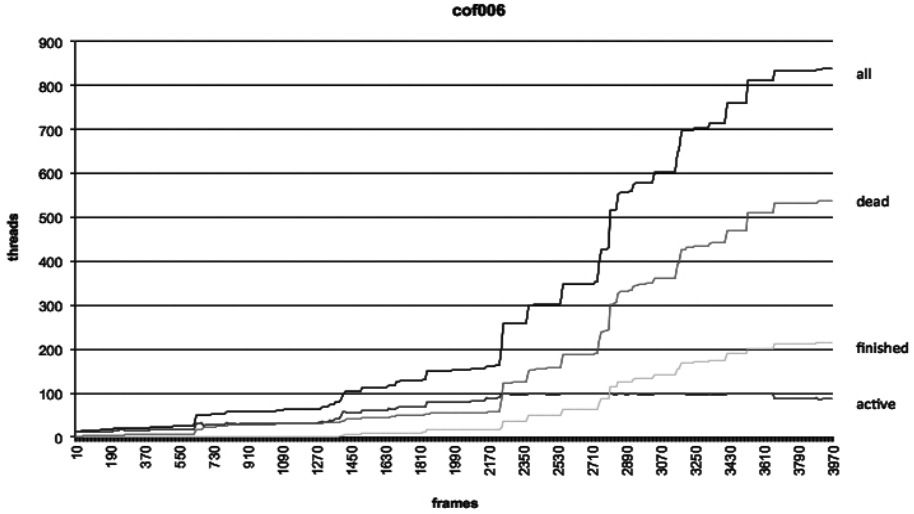

Fig. 5. Number of threads for a typical interpretation process as a function of time (in frames). The maximum nember of active threads has been set to 100.

The recognition rate of subactivities is shown in the table below. It was limited to 75% because of the noisy low-level input data with missing crucial evidence.

Table 2. Correctly recognised subactivities in 20 test sequences.

SEQUENCE	1	2	3	4	5	6	8	9	18	25	29	58	59	62	63	66
Arrival	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Passenger-Boarding-Preparation	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Unloading-Loading-AFT	1	1	1	0	1	0	1	1	1	1	0	0	0	0	0	0
Unloading-Loading-FWD				1		1	0	0				1	1		1	1
Refuelling			0	0	0			0		0		0		0		
Pushback-Arrival	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
Passenger-Bridge-Leaves-PBB-Zone	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Departure	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

To prove the domain-independence of SCENIOR, we also applied the system to activity data of the smart-home environment CASAS[3]. After establishing an ontology for the new domain, SCENIOR recognised all activities without any problems.

5 Conclusions

We have presented the scene interpretation system SCENIOR, designed to work with (i) conceptual knowledge bases expressed in the standardised ontology language OWL-DL, (ii) extended by SWRL rules for constraints, and (iii) supported by a probabilistic scene model for a preference measure. An interpretation strategy employing up to 100 parallel interpretation threads has been realised with JESS rule engines, and successful real-time interpretations have been achieved for noisy aircraft turnaround scenes. The results show that high-level interpretation of activities in low-structured domains and with noisy input data may face formidable ambiguity problems. We believe that the system architecture presented in this contribution has all ingredients to cope with such problems and may prove its worth in diverse applications. A first proof has been obtained in terms of a successful application SCENIOR to the CASAS smart-home environment by simply exchanging the ontology.

References

1. Fusier, F., Valentin, V., Brémond, F., Thonnat, M., Borg, M., Thirde, D., Ferryman, J.: Video Understanding for Complex Activity Recognition. Machine Vision and Applications, 18(3), 167-188 (2007)
2. Cohn, A.G., Magee, D., Galata, A., Hogg, D., Hazarika, S.: Towards an architecture for cognitive vision using qualitative spatio-temporal representations and abduction. In: Freksa, C., Brauer, W., Habel, C., Wender, K.F. (eds.), Spatial Cognition III, LNCS (LNAI), vol. 2685, 232–248. Springer, Heidelberg (2003)
3. Shanahan, M.: Perception as abduction: Turning sensor data into meaningful representation. Cognitive Science 29, 103–134 (2005)
4. Moeller, R.; Neumann, B.: Ontology-Based Reasoning Techniques for Multimedia Interpretation and Retrieval. In: Kompatsiaris, Y., Hobson, P. (eds.) Semantic Multimedia and Ontologies: Theory and Applications, 55–98. Springer, Heidelberg (2008)
5. Morariu, V.I., Davis, L.S.: Multi-agent event recognition in structured scenarios. CVPR-2011 - IEEE Conference on Computer Vision and Pattern Recognition (2011)
6. Neumann, B.: Bayesian Compositional Hierarchies - A Probabilistic Structure for Scene Interpretation. TR FBI-HH-B-282/08, Univ. of Hamburg, Dep. of Informatics (2008)
7. Neumann, B., Moeller, R.: On Scene Interpretation with Description Logics. In: Christensen, H.I., Nagel, H.-H. (eds.), Cognitive Vision Systems, LNCS, vol. 3948, 247-275. Springer, Heidelberg (2006)
8. Norvig, P.: Paradigms of Artificial Intelligence, Morgan Kaufmann, San Francisco (1992)
9. Bohlken, W., Neumann, B.: Generation of Rules from Ontologies for High-level Scene Interpretation. In: Governatori, G., Hall, J., Paschke, A. (eds.) RuleML 2009. LNCS, vol. 5858, 93–107. Springer, Heidelberg (2009)

[3] ailab.wsu.edu/casas/