Expert Systems with Applications 38 (2011) 4068–4079

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Determining the best suited semantic events for cognitive surveillance

C. Fernández *, P. Baiget, F.X. Roca, J. Gonzàlez

Computer Vision Center, UAB, Edifici O, Campus UAB, 08193 Barcelona, Spain

ARTICLE INFO

Keywords:
Cognitive surveillance
Event modeling
Content-based video retrieval
Ontologies
Advanced user interfaces

ABSTRACT

State-of-the-art systems on cognitive surveillance identify and describe complex events in selected domains, thus providing end-users with tools to easily access the contents of massive video footage. Nevertheless, as the complexity of events increases in semantics and the types of indoor/outdoor scenarios diversify, it becomes difficult to assess which events describe better the scene, and how to model them at a pixel level to fulfill natural language requests. We present an ontology-based methodology that guides the identification, step-by-step modeling, and generalization of the most relevant events to a specific domain. Our approach considers three steps: (1) end-users provide textual evidence from surveilled video sequences; (2) transcriptions are analyzed top-down to build the knowledge bases for event description; and (3) the obtained models are used to generalize event detection to different image sequences from the surveillance domain. This framework produces user-oriented knowledge that improves on existing advanced interfaces for video indexing and retrieval, by determining the best suited events for video understanding according to end-users. We have conducted experiments with outdoor and indoor scenes showing thefts, chases, and vandalism, demonstrating the feasibility and generalization of this proposal.

1. Introduction

Automatic content-based video indexing has been requested for digital multimedia databases for the last two decades, and more recently, this need has also been emphasized in particular for video surveillance applications (Foresti, Marcenaro, & Regazzoni, 2002). Surveillance systems have strong storage and computer power requirements, deal with continuous 24/7 monitoring, and manage a type of content that is susceptible to be highly compressed. Moreover, the number of security cameras increases exponentially worldwide, opening windows of opportunity for smart forensic analyses as vast archives of recordings constantly grow.

Current surveillance solutions for video annotation are robust when solving common visual tasks like segmentation, object recognition, or tracking, and handling specific issues, e.g. shadows, occlusions, or weather conditions. However, emulating the potential of human labor demands a deeper analysis. In particular, semantic context plays a fundamental role in the recognition of complex events (Smeulders et al., 2000). As a consequence, recent tracks in this field aim to enhance the results of tracking techniques by incorporating video understanding capabilities that detect and describe complex events observed in the video sequences, by means of expert knowledge (Fernandez-Caballero, Gomez, & Lopez-Lopez, 2008; Vallejo, Albusac, Jimenez, Gonzalez, & Moreno, 2009).

Nevertheless, modeling semantic events becomes a difficult task for experts: which ones are best suited for the description of a specific scene? The chosen events can be excessively particularized to ad hoc scenarios, or be too generic thus giving no relevant information; some events may be redundant, and some may be of no use for the concrete objectives of the system. These problems are augmented by the fact that most of the symbolic approaches used nowadays model knowledge in a bottom-up fashion, thus distancing themselves from the requirements of end-users (Albanese et al., 2008; Borzin, Rivlin, & Rudzsky, 2007; Fusier et al., 2007; Nagel & Gerber, 2008). As a result, even for sophisticated video understanding systems it is especially difficult to assess how complete and appropriate are the semantic descriptions of events described.

In order to cope with these issues, we propose a methodology that guides the modeling and evaluation of the best suited semantic events given image sequences from the surveillance domain. It considers the following steps:

1. First, in order to learn which events are important for our selected surveillance domains, several video sequences from indoor and outdoor scenarios are textually described by end-users.
2. The semantic descriptions are then used to build up the different ontological knowledge bases of the system, by means of a detailed top-down procedure accomplished by experts that makes events extensive to the tackled domain.

* Corresponding author. Tel.: +34 93 581 18 28; fax: +34 93 581 16 70.
E-mail addresses: perno@cvc.uab.cat, perno@cvc.uab.es (C. Fernández).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.09.070

3. Finally, the generality and extensibility of the produced models within the domain is tested on new, semantically rich indoor and outdoor video sequences.

Our proposal bases on the cognitive vision system presented in Fernández, Baiget, Roca, and Gonzàlez (2008) Gonzàlez, Rowe, Varona, and Roca (2009), and extends it with the following contributions:

- an expert-based ontological procedure models semantic events for a video surveillance system and assesses their suitability and completeness;
- the top-down modeling of the ontological models facilitates user interaction capabilities toward advanced video indexing and retrieval; and
- the method unifies scenario-dependant models into generally applicable ones by using the evidence given by end-users.

The resulting system builds upon the effective recognition of semantic context that is user-oriented, i.e., modeled according to the expectations of end-users.

This contribution is structured as follows: next section reviews similar work on the field. Section 3 overviews the two steps of the proposed methodology, i.e., top-down event modeling and bottom-up event inference. Subsequent sections explore the modeling procedure in more detail: Section 4 describes the construction of the ontological knowledge bases at different levels, and Section 5 implements natural language interfaces for description and query retrieval that will be used to demonstrate the effectiveness of the generated models. Sections 6 present experimental results of video indexation/retrieval and Section 7 and draw some final remarks.

2. Related work

In the literature, many methods for content-based video indexing deal with similarity measures based on trajectory, color, texture, and shape (Conci & Castro, 2002; Yoo, Park, & Jang, 2005). They commonly search for video shots by computing low-level features on entire or partitioned image frames, which are compared to those in consecutive frames to detect strong transitions (Lee, Yoo, & Jang, 2006; Zhu & Liu, 2009). Although low-level features are particularly useful for still image retrieval (Conci & Castro, 2002; Smeulders et al., 2000; Yoo et al., 2005) and video retrieval in movies, broadcast news, or sports (Xiong et al., 2006), they exhibit practical drawbacks for video surveillance. Firstly, surveillance footage hardly ever presents strong transitions between consecutive frames, since the changing image fractions are usually too small to result in detectable changes. Secondly, the semantic analysis assessed by low-level features is very limited, especially when working on very specific contexts. Finally, users prefer to retrieve content regarding higher-level features, e.g. the semantic explanation of the occurrences or their circumstances.

Few approaches on content-based video retrieval tend to incorporate understanding capabilities to their systems, thus allowing flexible user queries towards content retrieval (Le, Boucher, Thonnat, & Bremond, 2008). Towards this end, the use of top-down image retrieval techniques has been proven to assist the recognition of context by providing semantic guidance through the process (Torralba, Fergus, & Freeman, 2008). Top-down approaches are especially interesting as well in the case of video browsing, which enhances the retrieval capabilities by organizing the videos given their essential semantic content (Xiong et al., 2006).

The recognition of events in video sequences has been extensively tackled by the research community, ranging from simple actions like walking or running (Niebles, Wang, & Fei-Fei, 2008) to complex, long-term, multi-agent events (Laxton, Lim, & Kriegman, 2007). Nevertheless, the recognition of behaviors more complex than basic interactions has not been investigated as exhaustively as the rest. Two main approaches are generally followed in the recognition of non-basic events: probabilistic frameworks (Xiang & Gong, 2006), or rule-based approaches, in which complex events are recognized as the combination of atomic primitives structured by predefined or learnt rules (Zhang, Huang, & Tan, 2008).

Given that surveilled scenarios are usually specific environments like traffic locations, airports, banks, or border controls, to cite few, it is reasonable to make use of domain knowledge in order to deal with uncertainty and evaluate context-specific behaviors. Recently, different tools based on symbolic approaches have been proposed in order to define the domain of events appearing in selected environments, e.g. those based on conceptual graphs or conditional networks. Nagel and Gerber (2008) proposed a framework that combines situation graph trees (SGT) with fuzzy logic reasoning, in order to generate descriptions of observed occurrences in traffic scenarios. Extensions of Petri Nets have also been a common approach to model multi-agent interactions, and used as well for human activity detection (Albanese et al., 2008). Some other recent approaches have employed symbolic networks combined with rule-based temporal constraints, e.g. for activity monitoring applications (Fusier et al., 2007). Fig. 1 shows examples of these symbolic structures used in video surveillance.

All these symbolic models, which work with predefined behaviors, show good performances at behavior recognition, provide explanations of the decisions taken, and allow uncertainty to be incorporated to the analysis, thus making it more robust to noisy or incomplete observations. We choose SGTs over other symbolic approaches due to the efficacious mechanisms of specialization

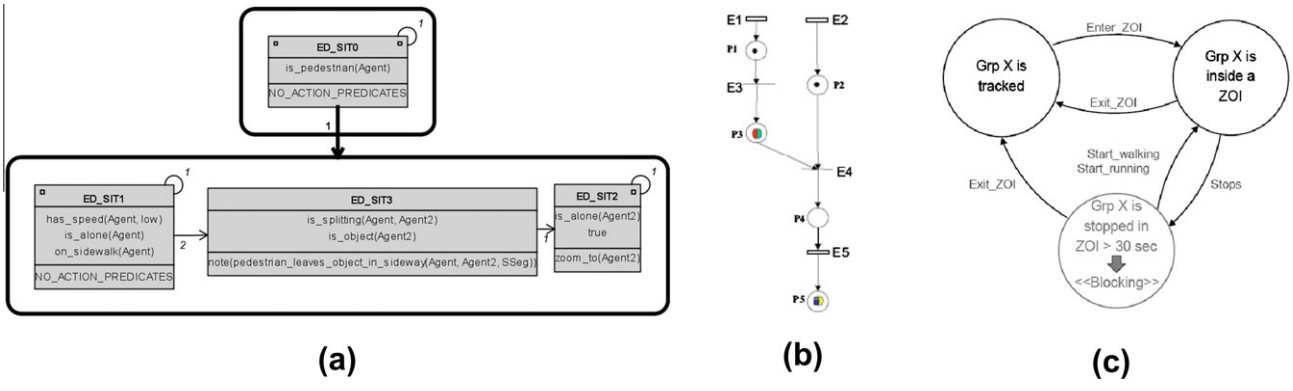


Fig. 1. Common symbolic approaches for behavior modeling: (a) situation graph tree (Nagel & Gerber, 2008); (b) Petri nets, (Albanese et al., 2008; Borzin et al., 2007) and (c) symbolic network (Fusier et al., 2007).

and prediction they incorporate, which help modeling the universe of situations in a clear, flexible, and controllable manner. SGTs and fuzzy metric-temporal logic, unlike Petri nets, are adapted to model and evaluate human behaviors on specific contexts, which we provide by means of ontologies.

The cited symbolic approaches allow semantic representations of the events detected, which facilitate implementing user-computer interfaces. Nonetheless, none of them carries out a thorough evaluation of the correctness or suitability of the selection of events, mainly due to the limited amount of semantics found in the video sequences. Other works have proposed lists of semantic events for the surveillance domain directly proposed by specific groups (Vezzani & Cucchiara, 2008), or based on the system capabilities to generate them (Fernandez-Caballero et al., 2008; Sanchez, Patricio, Garcia, & Molina, 2009). We propose instead to base the models on evidence provided by human participants.

3. General system

The general architecture of the proposal is presented in Fig. 2. We divide the system in three distinguished levels devoted to visual, conceptual, and user interfacing tasks, and the presented process is as well divided in two steps: an initial top-down modeling of the knowledge bases guided by an expert, and a subsequent automatic, bottom-up inference by the system using the resulting event models.

The top-down event modeling works as follows: first, based on several training videos, we gather event descriptions reported by a large number of non-expert users and assess the variability of these reports. The descriptions are then used to build the semantic models in a top-down fashion, which will be later used for the tasks of automatic event description in different videos. The conceptual models are designed in a strict top-down fashion, unlike the majority of current approaches for video indexing and understanding. Our integrative architecture incorporates a large component of domain-knowledge that is managed by dedicated modules, a common characteristic of expert systems.

Once the models are available, the system performs bottom-up event inference on new video sequences. Video footage is first analyzed by motion trackers: the visual stage simultaneously tracks multiple targets in unconstrained and dynamic open-world scenarios. In our experiments, the detection of targets follows a statistical background-subtraction approach based on color and intensity cues (Gonzàlez et al., 2009). Subsequently, the object trackers provide instantaneous target states over time, including quantitative data (e.g. velocity, size) and qualitative information (e.g. occlusions, groupings, splits, target births and deaths). Enhanced details and additional information can be found in Gonzàlez et al. (2009).

The bottom-up inference continues at the conceptual levels. The quantitative data obtained from tracking is conceptualized and processed by the spatiotemporal inference module, which reasons about basic facts using general dynamic rules and spatio-conceptual models. At the contextual reasoning stage, we use domain-specific knowledge to interpret the context of each occurrence and produce linguistic-oriented predicates. Each predicate involves concepts like agents, objects, or locations, and relational patterns from the ontology: these constitute the indexes stored in a relational database to enable video retrieval. Final modules for user interfacing allow richer interactions with end-users. In our case, we have implemented (i) a module to generate natural language descriptions of the event indexes, and (ii) a module that interprets natural language texts to accomplish efficient query-based retrieval.

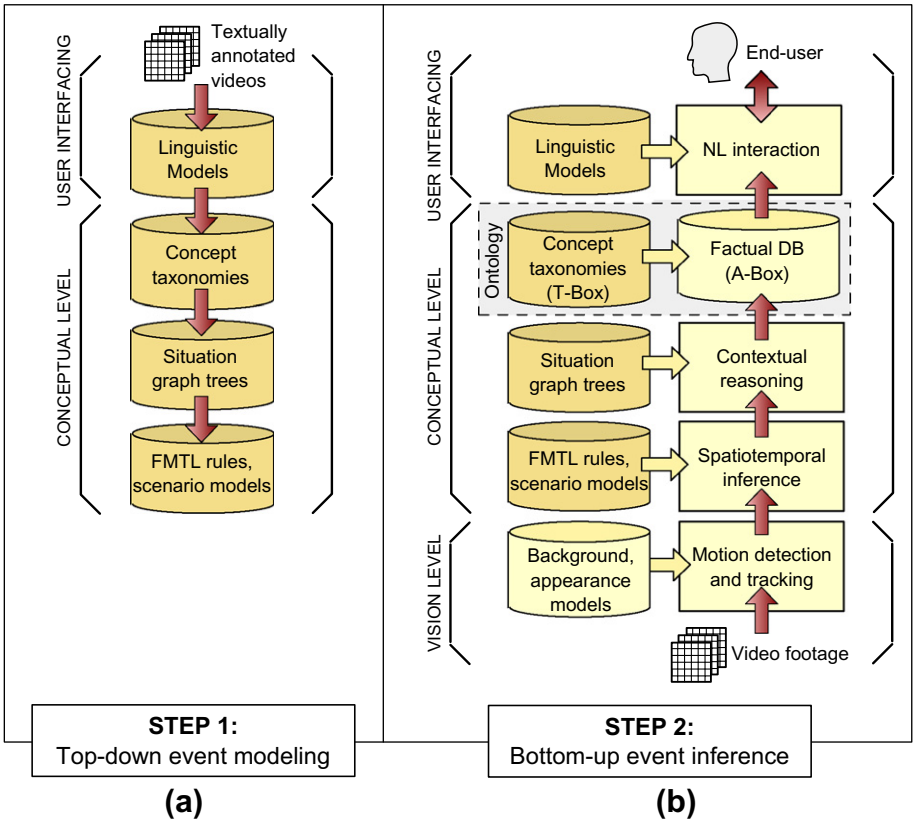


Fig. 2. General overview. (a) First, general knowledge bases are built top-down, based on end-user descriptions of events. (b) Once domain knowledge is available, automatic indexing and retrieval of any video in the domain is accomplished bottom-up.

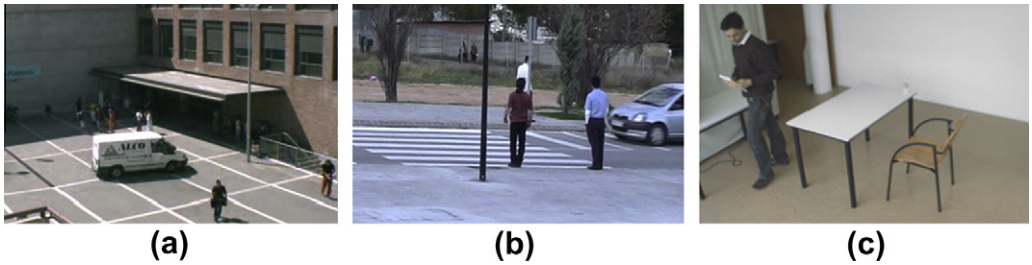


Fig. 3. Snapshots of outdoor (a,b) and indoor (c) video surveilled scenarios used for the ground-truth annotation of semantic evidence.

Next sections detail how to accomplish the top-down modeling of events.

4. Conceptual level

This section describes the top-down modeling employed to address tasks of knowledge management, inferential reasoning, and video understanding. The different steps include (i) building a domain ontology from linguistic psychophysical tests run on several subjects, (ii) contextualizing targeted events with concrete models that decompose them into simple facts, and (iii) link these facts to spatiotemporal data available from tracking.

4.1. Ontological modeling of relevant events

The target events to be detected in surveilled footage are typically determined by the purposed application. Nevertheless, assessing interpretations often becomes uncertain when dealing with complex events, leading to engineered solutions that may differ from end-user's perceptions. In order to deal with this, we have run questionnaires to identify which events are relevant to end-users in our restricted domain, in order to model them in a top-down fashion.

The ground-truth annotation of events has been extracted this way from psychophysical experiments of manual video annotation. Three scenes from indoor and outdoor scenarios have been recorded, showing different kind of interactions among people, objects, and vehicles, see Fig. 3. They show some complex events like stealing objects, crossing roads, waiting to cross, or getting almost run over by cars. A population of 60 English speakers were requested to visualize the videos.[1] 40 of the subjects were told to annotate at least 20 notable occurrences happening in each training sequence, the other 20 did the same for the two test sequences used for experimental results. Similar annotations were manually gathered together by experts, e.g. 'talk' – 'have a conversation' – 'discuss' → 'talk to someone'. Table 1 gives the frequency of common annotations for outdoor and indoor training videos. For events occurring more than once in the same video, the maximum frequency was considered.

An ontology of events has been created out of the results provided. Each annotation incorporates, explicitly or implicitly, the semantic context required to model an event, by means of a series of concepts that have been structured in three categories: events, entities, and constraints. The *Event* concepts identify the occurrence described, and are organized from simple to complex as (i) spatiotemporal inferences from tracking, (ii) interactions among entities, and (iii) interpretations of complex events in specific contexts. *Entity* concepts determine the nature of the participants in the event, which can be agents, objects, or locations. Finally, *Constraint* concepts account for the roles that entities are required to satisfy within an event, i.e., the list of agents, patients, locations, or objects needed. All these concepts are classified in taxonomies and together conform the terminological part of the ontology, the so-called T-Box $\mathcal{T}$ (Guarino, 1995). Table 2 reports how the annotated events are used to build the T-Box of the ontology: the entities required by each event are identified, and related to the particular event by means of constraints, which give additional information on the type of relationship held with each of the entities.

Table 1
Most common annotations for the two scenarios, sorted by percentage of people that used them to describe the semantic events.

Use (%)	Outdoor annotations	Use (%)	Indoor annotations
100	Leave object	100	Pick up/retrieve bag
100	Wait/try to cross	96	Leave a location
90	Walk in a location	96	Use vending machine
86	Cross the road	96	Sit down at a table
84	Run off/away	92	Talk to someone
84	Yield someone	90	Appear in a location
80	Chase after someone	88	Leave a bag on the floor
70	Pick up an object	85	Stand up
63	Join someone at a location	81	Shake hands with someone
60	Appear in a location	69	Kick/hit vending machine
50	Steal object from someone	62	Carry a bag
47	Do not allow someone to cross	58	Go/walk to a location
44	Danger of runover	50	Abandon/forget an object

Apart from $\mathcal{T}$, the ontology also incorporates an ABox $\mathcal{A}$ storing concept instances, i.e., factual information regarding the world state and the individuals existing on it (Guarino, 1995). Once the abstract events, constraints, and entities are satisfied for a certain world state, these concepts are instantiated into the factual database as *Facts*, *Constraint instances*, and *Entity instances*, respectively. For example, for the *theft* event in Table 3, the ontology requires a thief, *isAgent(Pedestrian)*, a victim, *has_agent_interaction(Pedestrian)*, and a stolen item, *has_object_interaction(Object)*, in this case fulfilled by instances *ped2*, *ped1*, and *obj1*, respectively.

In the end, the domain of interest is formally represented by a knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, the factual database, which includes both the concepts and their instances. Fig. 4 gives a concise view of the factual database implemented: the abstract concepts are *Events*, *Entities*, and *Constraints* that state which entities are needed for which events. On the other hand, instances for these three types of concepts are stored in the three other tables: *Entity instances* list appearing entities, *Facts* are detected occurrences of events, and *Constraint instances* link ones to the others.

[1] The subjects (half men, half women) were recruited from 5 different countries and from different age intervals: 18–25 (12%), 25–35 (66%), and over 35 (22%). They also came from different backgrounds: technical studies (27%), sciences (40%), humanities (30%), and other (3%).

Table 2
A list of examples on how the user annotations are used to populate the T-Box $\mathcal{T}$ of the ontology with concepts and relationships.

User annotation	Event	Entities	Constraints
pick up bag	`pick_up`	Pedestrian PickableObject	`is_agent` `hasObjectInteractionWith`
wait to cross	`wait`	Pedestrian Location	`is_agent` `hasLocationInteractionWith`
leave a location	`exit`	Agent Location	`is_agent` `hasLocationInteractionWith`

steal object from someone	`theft`	Pedestrian
		PickableObject
		Pedestrian
			`is_agent`
			`hasObjectInteractionWith`
			`hasPatientInteractionWith`

danger of runover	`danger_of_runover`	Vehicle Pedestrian	`is_agent` `hasPatientInteractionWith`
abandon/forget object	`abandoned_object`	PickableObject Location	`isObject` `hasLocationInteractionWith`

meet with someone	`meet`	Pedestrian
		Pedestrian
		Location
			`is_agent`
			`hasPatientInteractionWith`
			`hasLocationInteractionWith`

Table 3
Possible instances of entities (left) used in event indexes (right). For a *theft* to be indexed, *ped2*, *ped1*, and *obj1* must accomplish a certain semantic context.

Entity type ($\mathcal{T}$)	Instance ($\mathcal{A}$)		Event type ($\mathcal{T}$)	Indexed fact ($\mathcal{A}$)
Pedestrian	*ped2*	⇒	Spatiotemporal	*walk (ped2,fast)*
Vehicle	*veh1*		Interaction	*appear (ped2,sidewalk)*
Location	*sidewalk*		Interaction	*pick_up (ped2,obj1)*
Object	*obj1*		Interpretation	*theft (ped2,ped1,obj1)*
Descriptor	*fast*		Interpretation	*danger_of_runover (veh1,ped2)*

4.2. Contextual modeling

At this point, the ontology already states which elements are required by each event, but we still need to model the domain-specific context in which an event occurs. As stated before, events are *situated* in their context by means of SGTs.

An SGT defines the universe of possible situations in which an agent can participate. Each situation scheme evaluates a set of conditions in form of atomic predicates and reacts when all of them are asserted. In our case, reactions are `note` commands that produce the linguistic-oriented event indexes seen and facilitate NL-based retrieval (Nagel & Gerber, 2008). Fig. 5(a) and (b) show parts of SGTs that exemplify their basic mechanisms to contextualize: situations are hierarchically nested from general to specific by means of *specialization* edges forming a tree, and sequentially connected by unidirectional *prediction* edges producing graphs within the tree. Self-prediction edges hold a current situation until any continuing situation applies. This scheme recurrently decomposes the evaluation of complex facts into series of low-level facts, which need to be asserted sequentially.

Carrying on the top-down modeling of semantic events, we build SGTs to define a priori the situations agents can be in. To do so, complex actions are decomposed in a combination of simpler events that are sequentially connected in time. Table 4 details the decomposition of the situations `left_object`, `abandoned_object`, `pick_up`,

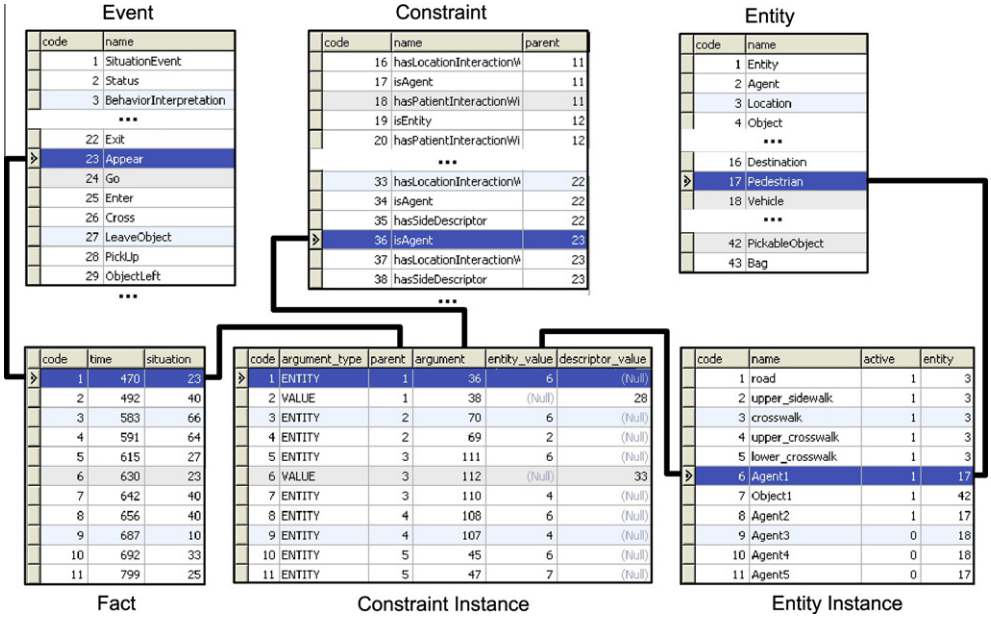


Fig. 4. Detail of the structured relations between concepts and instances in the factual database: upper tables contain T-Box concepts (events, constraints, and entities), and lower ones show their A-Box instances.

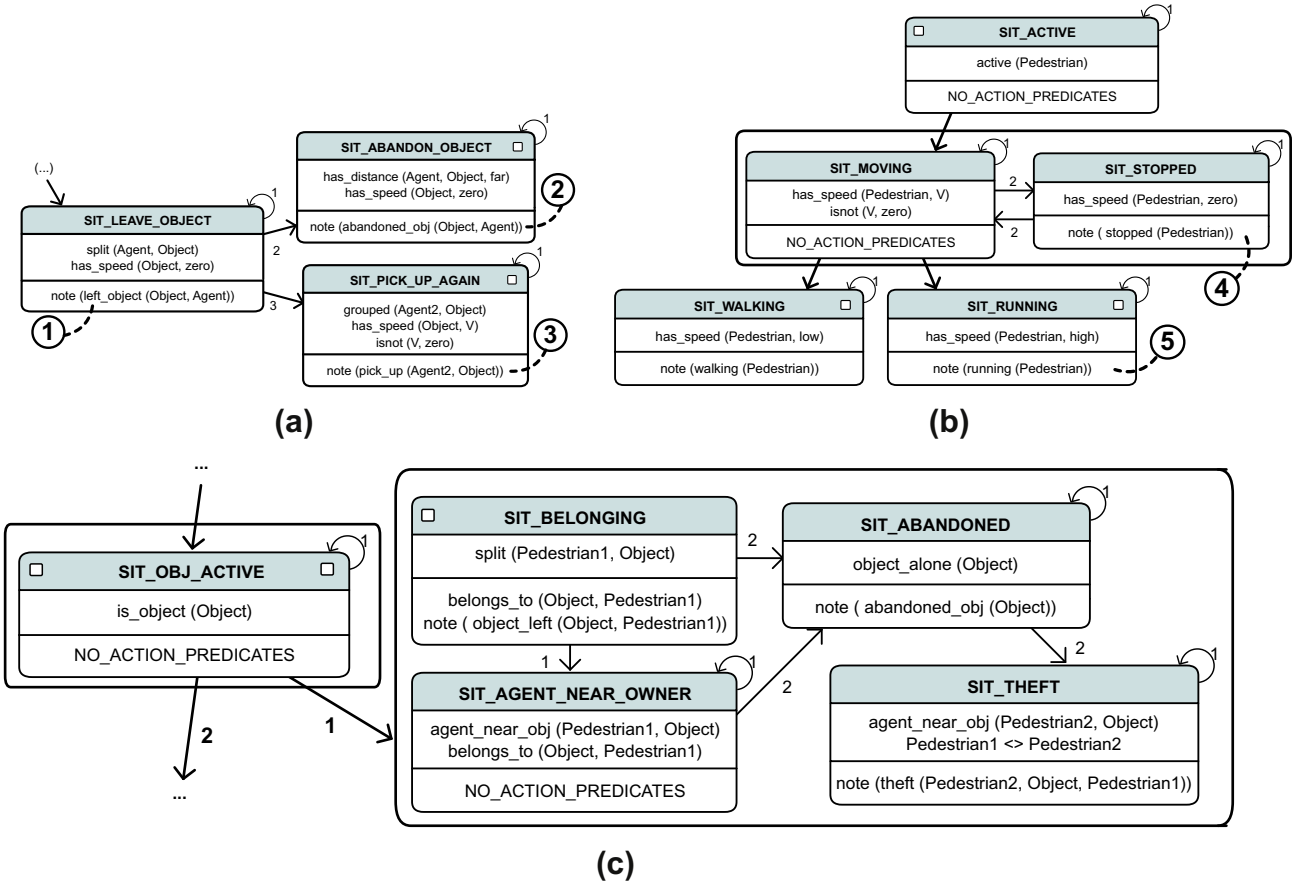


Fig. 5. SGT mechanisms to situate events in a context: (a) temporal prediction and (b) specialization. These SGTs incorporate the decompositions shown in Table 4. A part of an SGT used in outdoor scenes is shown in (c).

Table 4
To model SGTs, high-level events from the ontology are decomposed into conjunctions of simpler events that are temporally chained. The obtained decompositions are then merged in a single tree of situations for each type of agent.

ID	High-level predicate	Temporal decomposition
①	`left_object (Object,Agent)`	t_0: `split (Agent,Object)` ∧ `has_speed (Object,zero)`
②	`abandoned_object (Object,Agent)`	t_0: `left_object (Object,Agent)` t_1: `has_distance (Agent,Object,far)` ∧ `has_speed (Object,zero)`
③	`pick_up (Agent,Object)`	t_0: `_object (Object,Agent)` t_1: `grouped (Agent,Object)` ∧ `has_speed (Object,V)` ∧ `is_not (V,zero)`
④	`stopped (Pedestrian)`	t_0: `has_speed (Pedestrian,V)` ∧ is_not (V,zero) t_1: `has_speed (Pedestrian,zero)`
⑤	`running (Pedestrian)`	t_0: `has_speed (Pedestrian,high)`

`stopped`, and `running`. It can be observed that many elements in the various decompositions are common, and thus can be merged in a single SGT. Simpler events are recursively decomposed until reaching to a combination of mere spatiotemporal descriptions. The five examples of decomposition in Table 4 have partially generated the SGTs shown in Fig. 5(a) and (b). More complex events are also possible: for example, by combining actions like leave object, get close, pick up, and run, a *theft* event can be modeled, as shown in Fig. 5(c). Extra events are sometimes included into the ontology for better definition of a particular context, e.g. for the event *belongs_to*.

The role of SGTs in the overall scheme is twofold: on the one hand, they help understanding the full picture of a scene by assessing high-level interpretations from concrete pieces of information. And on the other hand, SGTs make it possible to distrust or simply neglect certain frames when the position of a target suddenly changes to a far distant location, e.g. if the tracker freezes for a while. These and similar situations make them a suitable tool to partially bridge both semantic and sensory gaps in our domain.

The current implementation of the SGT only asserts those predicates with highest confidence values, which unfits the system to handle multiple valid hypotheses at the same time, but in exchange avoids a combinatorial explosion of solutions. Only one event annotation is produced by the SGT per frame and tracked agent, which allows us to associate each predicate with an interval of validity, and build a history of events related to each detected object. When an alarm is missed at the vision level, an SGT instantiates the most specific of the events in the graph given the state

conditions available. The more levels we define in the hierarchy, the more robust the system is in front of lacking information, but the computational cost increases.

4.3. Spatiotemporal modeling

The last conceptual task involves describing the multiple atomic events used in the SGTs in terms of low-level information provided by the motion trackers. To do so, a set of basic spatiotemporal rules are defined for the domain, focusing on general rather than particular contexts.

The reasoning engine of the system is based on *fuzzy metric-temporal logic* (FMTL), which extends conventional logic by temporal and fuzzy components. The first component permits the engine to represent and reason about propositions evaluated at each time-step, while the last one enables it to cope with uncertain or partial information, by allowing degrees of validity. Temporally-valid numerical status vectors from tracking are converted into `has_status` fuzzy predicates at each time-step, which convey information about the id and type of the target, its spatial location in a ground-plane representation of the scenario (X,Y), and his instantaneous orientation (*Theta*) and velocity (V) at time t

```
t ! has_status (Agent, X, Y, Theta, V).
```

General spatiotemporal rules for each type of agent assign fuzzy values like *slow* or *very fast*, according to the membership functions modeled. A schematic representation of the locations in the scenario is as well predefined in terms of factual atomic predicates. In addition to these two sources of information, the reasoner provides inferences of new facts based on temporal-geometric conditions: the role of the designer at this point consists of explaining every generic low-level predicate found in the SGTs in terms of the `has_status` variables. For instance, for a `similar_direction` predicate, the tracking data is derived to symbolic information as follows:

```
always (similar_direction (Agent,Agent2):-
    has_status (Agent,_,_,_,Or1,_),
    has_status (Agent2,_,_,_,Or2,_),
    Dif1 is Or1 − Or2,
    Dif2 is Or2 − Or1,
    maximum (Dif1,Dif2,MaxDif),
    MaxDif < 30).
```

Hence, the FMTL reasoner engine converts geometric information into qualitative knowledge that is time-indexed and incorporates uncertainty. Note that FMTL rules are defined generally for the domain, and not dependent on particular scenes: only the semantic zones must be modeled for a new scenario. This way, the models are extensible and tracking information is easily conceptualized and forwarded to the upper levels discussed.

5. User interfacing level

Video search and retrieval interfaces are used by end-users, thus demanding flexible and user-friendly tools for natural language interaction. In order to demonstrate the validity of our semantic framework to connect with advanced NL interfaces and fulfill non-trivial requests in English language, this section describes a possible extension to natural language generation (NLG) and natural language understanding (NLU).

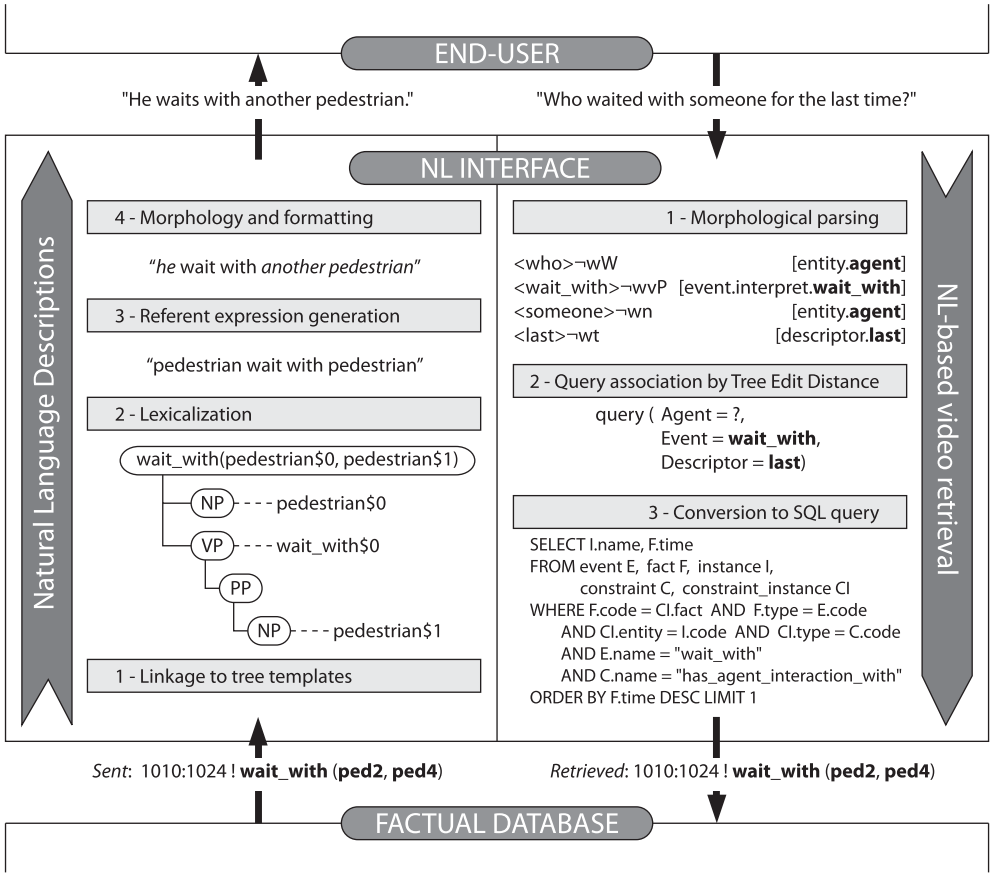


Fig. 6. Step results for the processes involved in the NLG and NLU modules. Notice that the concepts linked to words at different steps are either *Facts* or *Entity Instances* from the factual database, as seen in Fig. 4.

NLG has been often considered a process of choosing suitable expressions to communicate some content, whereas NLU has usually been regarded as a process of hypothesis management that decides for the most probable interpretation of linguistic inputs (Reiter & Dale, 2000). In our case, the first module facilitates the generation of NL sentences for the indexed events, while the latter enables video and information retrieval from NL textual queries. Fig. 6 illustrates these processes, explained next.

5.1. Natural language descriptions

The first stage of the NLG module enhances standard parsing techniques in order to convert an incoming predicate into a tree structure, which gives a unique predicate interpretation and provides a background structure for the final surface sentence. Predicate types are linked beforehand to tree templates, whose shapes come predefined by the already seen ontological constraints held by the event; for instance, *is_agent* determines the agent (subject of active sentence) for *wait_with*, see Fig. 6. In addition, different templates are possible depending on the information available: instead of "*X waits with Y*" we could have "*X waits with Y in Z*", thus producing an extended tree.

A lexicalization process maps semantic elements into linguistic resources (units or subtrees) that communicate their contents. Tree templates already assign lemmata to events and prepositions, but additional steps are required for entities. First, *particularizations* must be applied when available, e.g. replacing a general predicate *appear(agent,location)* by *appear(vehicle,left)*. Subsequently, lexical choices are given for specific parts of the domain, such as `upper_right` being expanded as "upper right side".

At this point, we must solve the issue on how to refer to entities so that they can be easily identified in the context of the discourse. This task is known as referring expression generation (REG) (Reiter & Dale, 2000), and we accomplish it with the help of onomasticons (Fernández et al., 2008). An onomasticon is a repository that tracks instances of entities along the discourse, allowing the system to answer questions like: *has it ever been instantiated?*, *more than once?*, *are there other instances of the same concept?*, *was it the central entity in the last sentence generated?*, or *was the last instance definite*? The proper combination of these REG cases allows the NLG module to choose the most appropriate referring expression, like *an [entity]*, *a new [entity]*, *the [entity]*, *this last [entity]*, *the second [entity]*. For example, if we have seen a car in the scene previously, and a new agent of type car appears, we use "*a new car*"; otherwise, if none of the vehicles or other agents seen was specifically a car, we use simply "a car", thus highlighting the class instead of the actual instantiation.

Finally, the morphological and surface realization process involves mapping the specification of a text into a surface text form, i.e., a sequence of words, punctuation symbols, and mark-up annotations to be presented to the end-user (Reiter & Dale, 2000). In practice, it consists of applying parsing techniques to modify either independent words (verb inflections or conjugations, plurals) or words depending of their surrounding context (contractions, vowel adjacency, prosodic effects). In the example of Fig. 6, the third person of the verb has been conjugated; similarly, this step also updates tenses ("*leave*" → "*has left*") and changes words in context ("*a agent*" → "*an agent*"). As a result of the morphological process, a rich semantic/syntactic tree structure with referred expressions and morphological forms is generated. The linearization of the tree nodes and a final addition of orthographical and formatting information provides a final surface form for the end-user.

5.2. NL-based retrieval

Following the idea of hypothesis management, the NLU module links textual sentences to their most accurate interpretations in the domain, in form of predicates related to scene concepts and instances. Once a proper formatting has been applied, an input sentence is analyzed through a sequence of three processes (Fernández et al., 2008): first, a morphological parser tags words with linguistic features depending on the context of apparition, and a syntactic/semantic parser builds a dependency tree out of the tagged sentence. Secondly, the resulting tree with ontological references is assigned to the most related query predicate from a collection of patterns. Finally, the obtained predicate is used to query the factual database of indexed occurrences. The process is detailed next.

The semantic part of the analysis already starts with the word tagging process: the lexical models attach domain concepts to words that potentially refer to them. Hence, there are two issues to solve, since (i) a word can be linked to several concepts, e.g.

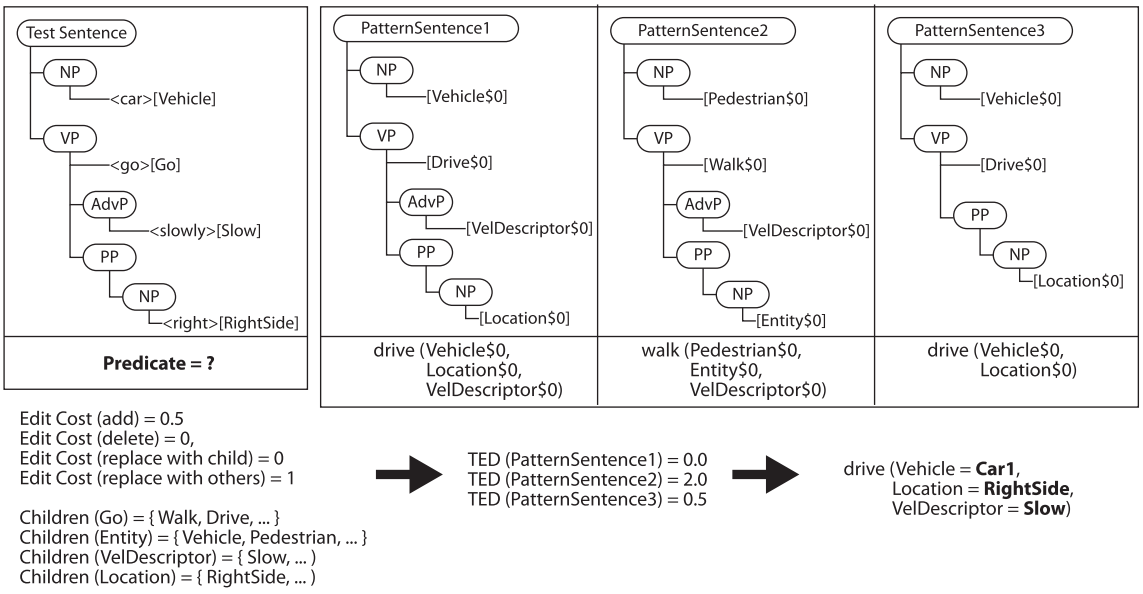


Fig. 7. A test sentence is compared to a collection of pattern trees, each one associated to a generic predicate. The predicate of that pattern with a lowest TED specializes its predicate with information from the sentence.

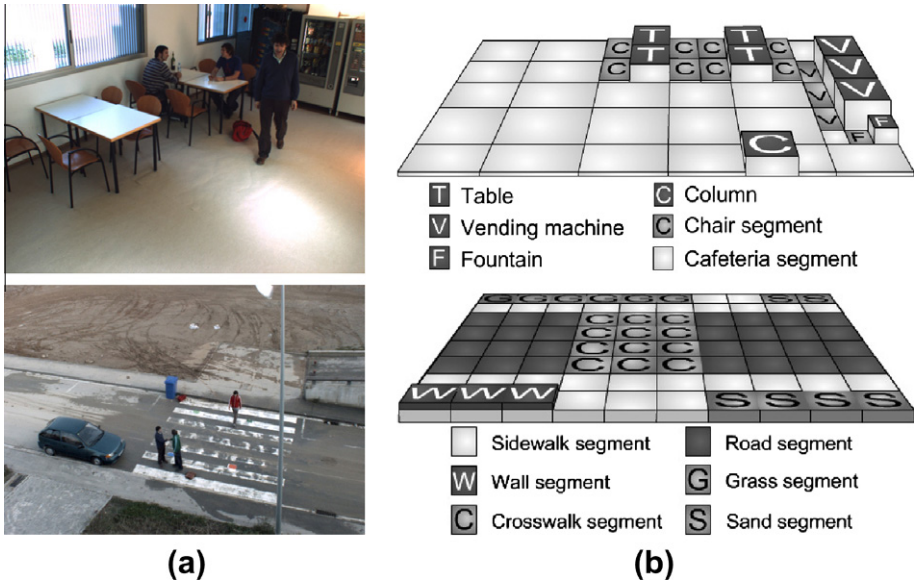


Fig. 8. Indoor and outdoor video footage to test indexing and retrieval (a) and their associated spatio-conceptual models (b).

word "turn *left*" (concept *OrientationDescriptor*) and "*left* entrance" (*Location*); and (ii) each concept may also have many words attached to it, as for the words "person", "pedestrian", or "walker" and the concept *Pedestrian*. Parsing rules solve the first ambiguity. Regarding the second issue, a robust system must be able to understand not modeled words, i.e., to sensibly link unknown words to a domain concepts. To this end, we rely on the WordNet lexical database (Fellbaum et al., 1998) to retrieve lists of closely related words, using semantic metrics based on synonymy and hypernymy. New word candidates are evaluated to determine the nature of the unknown word. As a result, the word is linked to a number of concepts that can explain it.

Next, a dependency tree is built with the help of syntactical rules, which first identify the heads of phrase classes and then recursively nest words and phrases hierarchically. The resulting tree is then compared to a collection of tree patterns by computing a semantically-extended tree edit distance (TED) (Bille, 2005), see Fig. 7. In order to compute the TED, the concepts at the leaves of the pattern trees are aligned to those from the test tree, and the TED evaluates the coincidence of each concept: it penalizes strongly the absences, penalizes the generalizations proportionally to the number of levels to the test concept, and does not penalize at all when the test concept matches or particularizes the pattern one. For example, the concept *Car* augments the distance with pattern tree 2 having *Pedestrian* at the corresponding leaf, but specializes the general concept *Vehicle* in the same position of pattern 3 with distance zero. The pattern tree with lowest distance to the test tree is decided as the most valid interpretation, and the fields of its associated predicate are particularized with specific information from the sentence.

A final step adapts the query to the relational language used for the factual database, in this case SQL. The retrieval process returns the entries that satisfy the query of the end-user, along with the interval of the video sequence corresponding to the event index. Some examples of NL-based retrieval are presented in the next section, along with the rest of the experimental results.

6. Experimental results

The ground truth annotation of events was accomplished using three different image sequences, two outdoor and one indoor. The first outdoor sequence (2250 frames@25 fps, 640 × 480 pixels) shows the entrance of a public building, where pedestrians come in and out and interact with some cars and motorbikes on their way. The second outdoor sequence (600 frames@15 fps, 1256 × 860 pixels) is a crosswalk scenario, in which 4 pedestrians enter a crosswalk in different manners, in the presence of vehicular traffic. The indoor training video (1575 frames@15 fps, 1256 × 860 pixels) contains specific events like leaving bags, greeting a person, taking objects from someone else, sitting down, or kicking a vending machine.

Two scenes from the same domain were recorded for tests, one in a traffic scenario and the other one in a cafeteria, see Fig. 8. These test scenes share similar events than the ones found in the test sequences, in completely different scenarios. The outdoor scene contains 1611 frames@15 fps of 720 × 576 pixels, in which pedestrians, pickable objects, and vehicular traffic interact in a pedestrian crossing. The indoor scene contains 2005 frames@15 fps of 1392 × 1040 pixels, in which people and objects interact among them and with the elements of a cafeteria, viz. a vending machine, chairs, and tables. Both sequences show complex events like abandoned objects, thefts, chases, or vandalism. These sequences have been automatically analyzed and indexed by the proposed system.[2]

The asserted events for every detected target have been stored in a SQL relational database to enable data retrieval. Every asserted event points to a temporal interval of validity in the sequence, and relates the involved target to its contextual blanket. Fig. 9 shows the results for automatic indexation, in which a collection of annotations for high-level events have been successfully generated for sequences recorded in outdoor and indoor surveilled scenarios, respectively. The collection of video annotations describe interactions among the involved entities, and also interactions and interpretations of complex occurrences.

Examples of content-based video retrieval are presented in Table 5, which retrieve episodes of sequences containing certain events or entities. More complex queries are possible, e.g. querying for chases after thefts, objects owned by different persons, or scenes in which a number of agents were seen at a certain location. As for the NL queries, acceptable propositions also restrict to the domain imposed by

[2] The sequences used in these experiments can be found at http://iselab.cvc.uab.es/tools-and-resources.

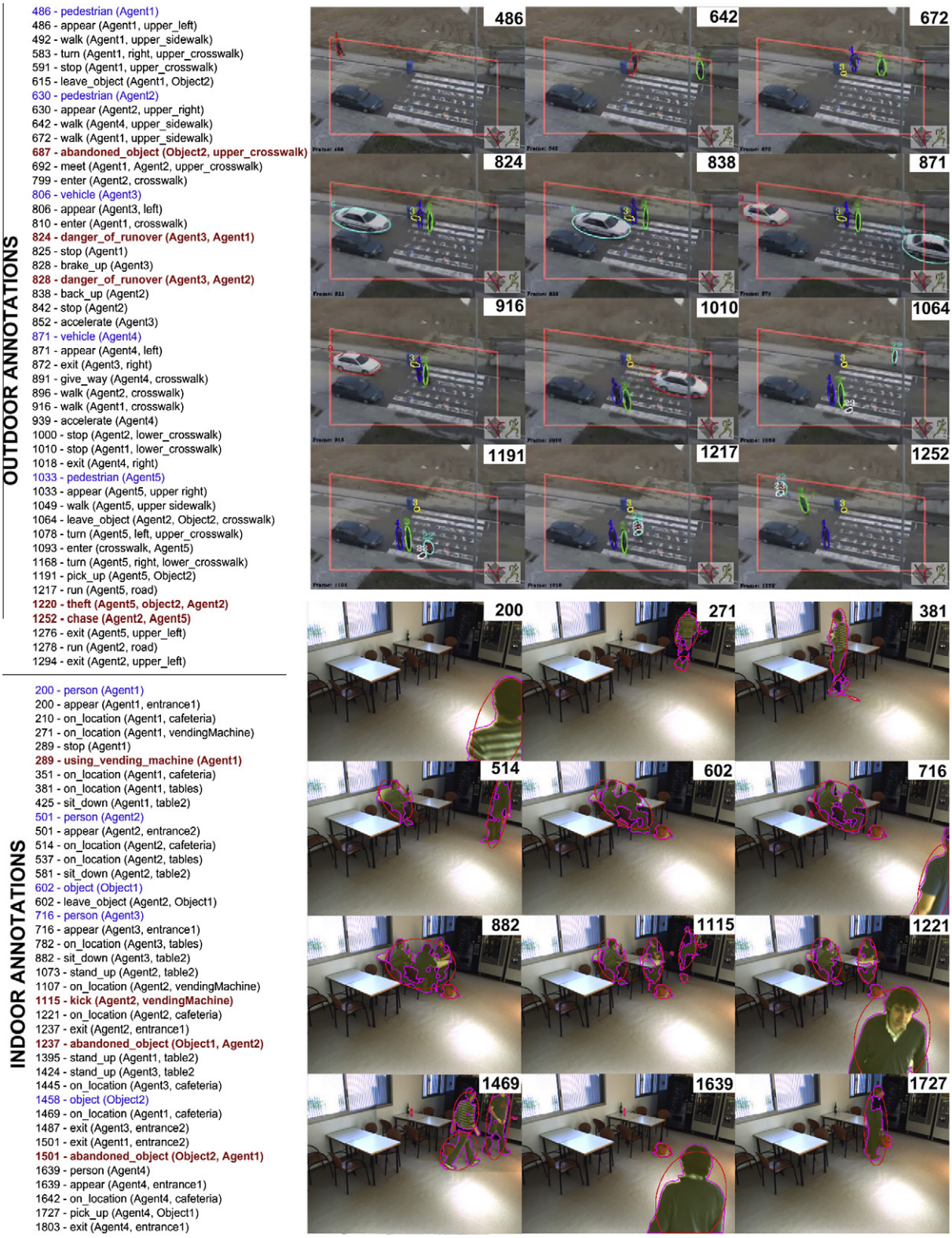


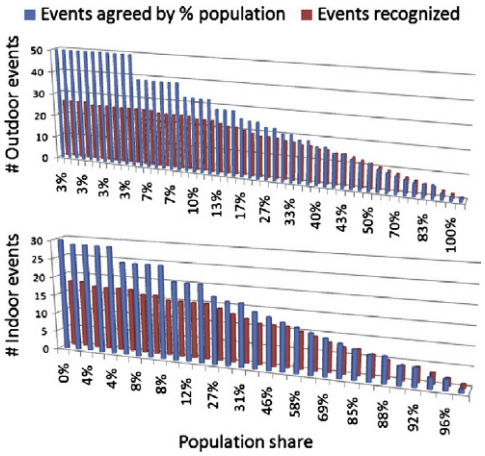
Fig. 9. The facts produced by the system (left) when processing the indoor and outdoor scenes (right) account for the main events and behaviors pointed out by end-users in other scenes of the domain.

the ontology. This way, users are enabled to ask for any modeled event involving any of the entities, which is related to any semantic zone in the scenario, and happens at any point or interval of time. These are some examples of the most repeated types of user queries that have been accepted by the NL module:

- *Show me pedestrians meeting between frames 300 and 1200.*
- *How many people has picked up bags?*
- *Have you seen any pedestrian running by the road after a theft?*
- *List all vehicles before frame 600.*

Table 5
Examples of retrieval of episodic events when querying for a given entity.

	Interval	Event	Arguments
Entity ID: **Agent5** Interval: 1200–1250 Sequence: Outdoor-1	1186–1202	pick_up	is_agent (**Agent5**) has_object_interaction_with (Object2)
	1186–1276	carry_object	is_agent (**Agent5**) has_object_interaction_with (Object2)
	1211–1219	run	is_agent (**Agent5**) has_location_interaction_with (Road)
	1220–1240	theft	is_agent (**Agent5**) has_patient_interaction_with (Agent1) has_object_interaction_with (Object2) has_property (Malicious)
	1241–1275	chase	is_agent (Agent1) has_patient_interaction (**Agent5**)
Entity ID: **Object1** Interval: 550–1250 Sequence: Indoor-2	501–601	carry_object	is_agent (Agent2) has_object_interaction (**Object1**)
	602–1236	leave_object	is_agent (Agent2) has_object_interaction (**Object1**) has_location_interaction (Hall)
	1237–1712	abandoned_object	is_patient (Agent2) has_object_interaction (**Object1**) has_property (Malicious)



Outdoor indexing		Indoor indexing	
Share	Detected/Agreed	Share	Detected/Agreed
3%	26/50 (52%)	3%	18/29 (62%)
10%	24/33 (73%)	11%	15/21 (71%)
20%	19/25 (76%)	25%	14/18 (78%)
33%	17/21 (81%)	31%	13/17 (76%)
40%	15/17 (88%)	42%	11/15 (73%)
50%	12/12 (100%)	50%	12/13 (85%)
60%	10/10 (100%)	61%	9/11 (82%)
70%	8/8 (100%)	69%	8/10 (80%)
80%	7/7 (100%)	80%	7/9 (78%)
90%	3/3 (100%)	92%	4/5 (80%)
100%	2/2 (100%)	100%	1/1 (100%)

Fig. 10. Correctly indexed events. Left graphic: horizontal axis shows the percentage of people agreeing with a set of events; vertical axis reports the total of events in this set, and the number out from them that were recognized. Right table: numeric details.

Similar concepts are automatically linked using the metrics over WordNet, such as *pedestrians–people*. In the experiments, subjects usually restricted to simpler queries. The difficult queries were usually too generic or stepped out of the domain, with sentences such as "*How is this person dressing?*" or "*Does it rain?*", in which case the concepts found could not be linked to the factual database. Out of the total number of queries asked that belonged to the domain, a 91% of them led to proper understanding by the system. Most of the non-understood questions were those starting with *why* or *how*, types that usually result less objective to answer.

These results have been compared to the validation data set provided by the second group of subjects. Fig. 10 shows the number of events agreed by a certain percentage of the population, and the events out of that set correctly identified by the system. Fig. 11 presents the percentage of events correctly recognized. As we can see, for sets of events agreed by above 50% of the population, the system recognizes all of them in the outdoor scenario and 85% of them in the indoor one. On the other hand, if we consider the set of events identified by more than 90% of the subjects, a recognition rate of more than 90% is achieved in both scenarios. The reason of the different performance between indoor and outdoor scenes is that although indoor image sequences permit a reduced viewpoint and incorporate less events, the events detected show a higher semantics, such as body gestures, facial expressions, and subtler interactions between agents, which require more knowledge than that one obtained solely from trajectory data.

Some examples of non-recognized annotations are *ignore_object*, *be_upset*, *be_hesitant*, *talk*, *realize_about_someone*, or *shake_hands*, among others, which mostly happened in indoor sequences. All undetected events were shared by less than 20% of the population, given the subjectivity of the interpretation, except for *talk* and *shake_hands*. In these two cases, the semantic framework facilitates retrieving non-modeled events by searching for similar concepts, e.g. *meet* or *interact*.

7. Conclusions and future work

State-of-the-art on surveillance video analysis is heading to the automatic exploitation of semantic context, in order to extract event patterns that permit us a better comprehension of image sequences. Nevertheless, few works assess the suitability and coverage of the selection of semantic events to model, and most of them are restricted to very specific scenarios, thus questioning the generalization capability of the methods used. In addition, these events should also be suited for end-user interfacing of video contents, something difficult to achieve by using bottom-up procedures.

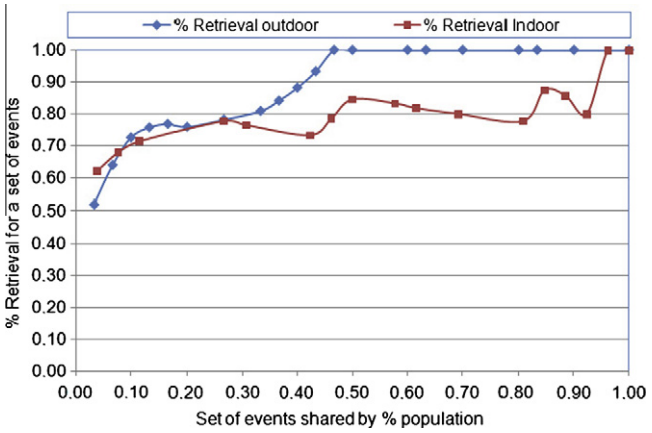


Fig. 11. Percentage of retrieval. Failures in indoor sequences are mainly due to unhandled recognition of expressions and gestures by the vision algorithms. Highlighted minima correspond to *be_upset*, *shake_hands*, and *talk* (left to right).

Our methodology contributes to these three challenges. First, it copes with the ambiguous and sometimes incorrect interpretations done by experts while building conceptual models. The ontology and the rest of the knowledge bases are modeled in a top-down manner from users' textual evidence, constituting a separate identifiable part of the design. The technique chooses the most suited event concepts from different scenarios, merging them into single models (ontology,SGT), and thus enabling generalization to different scenarios in the surveillance domain. And finally, since the ontology has been built from linguistic corpora, it provides straightforward connection to NL interfaces like those shown for video description and retrieval, allowing end-users to access meaningful video content flexibly by means of NL descriptions and dialogue-based instructions.

The resulting models can be independently maintained and increased, for being part of an expert system. Furthermore, this modular framework allows multimodality, as long as any new information from additional modules comes in form of atomic facts; in that case, it is easily integrated into the situation analysis.

Next steps will test the proposed framework to the challenging domain of movie and media analysis. To this end, current behavioral models will be enhanced by implementing a module for effective facial expression recognition, thus enabling the detection of most of the behaviors that could not be recognized in the low resolution surveillance videos. To consolidate the approach, steps will be taken toward (i) automatically learning the semantic context from visual features and (ii) holding multiple hypotheses as probable interpretations through the SGT traversal.

References

Albanese, M. et al. (2008). A constrained probabilistic petri net framework for human activity detection in video. *IEEE Transactions on Multimedia, 10*(6), 982–996.

Bille, P. (2005). A survey on tree edit distance and related problems. *Theoretical Computer Science, 337*(1–3), 217–239.

Borzin, A., Rivlin, E., & Rudzsky, M. (2007). Surveillance event interpretation using generalized stochastic petri nets. In *Eighth international workshop on image analysis for multimedia interactive services (WIAMIS'07), Santorini, Greece* (p. 4).

Conci, A., & Castro, E. M. M. M. (2002). Image mining by content. *Expert Systems with Applications, 23*(4), 377–383.

Fellbaum, C. et al. (1998). *WordNet: An electronic lexical database*. Cambridge: MIT Press.

Fernández, C., Baiget, P., Roca, F. X., & Gonzàlez, J. (2008). Interpretation of complex situations in a cognitive surveillance framework. *Signal Processing: Image Communication, 23*(7), 554–569.

Fernandez-Caballero, A., Gomez, F. J., & Lopez-Lopez, J. (2008). Road-traffic monitoring by knowledge-driven static and dynamic image analysis. *Expert Systems with Applications, 35*(3), 701–719.

Foresti, G. L., Marcenaro, L., & Regazzoni, C. S. (2002). Automatic detection and indexing of video-event shots for surveillance applications. *IEEE Transactions on Multimedia, 4*(4), 459–471.

Fusier, F. et al. (2007). Video understanding for complex activity recognition. *Machine Vision and Applications, 18*(3), 167–188.

Gonzàlez, J., Rowe, D., Varona, J., & Roca, F. X. (2009). Understanding dynamic scenes based on human sequence evaluation. *Image and Vision Computing, 27*(10), 1433–1444.

Guarino, N. (1995). Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human–Computer Studies, 43*(5–6), 625–640.

Laxton, B., Lim, J., & Kriegman, D. (2007). Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In *CVPR'07* (pp. 1–8). Minneapolis, USA: IEEE.

Le, T. L., Boucher, A., Thonnat, M., & Bremond, F. (2008). A framework for surveillance video indexing and retrieval. In *International workshop on content-based multimedia indexing, London, UK* (pp. 338–345).

Lee, M. H., Yoo, H. W., & Jang, D. S. (2006). Video scene change detection using neural network: Improved ART2. *Expert Systems with Applications, 31*(1), 13–25.

Nagel, H.-H., & Gerber, R. (2008). Representation of occurrences for road vehicle traffic. *AI-Magazine, 172*(4–5), 351–391.

Niebles, J. C., Wang, H., & Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision, 79*(3), 299–318.

Reiter, E., & Dale, R. (2000). *Building natural language generation systems*. Cambridge/UK: Cambridge University Press.

Sanchez, A. M., Patricio, M. A., Garcia, J., & Molina, J. M. (2009). A context model and reasoning system to improve object tracking in complex scenarios. *Expert Systems with Applications, 36*(8), 10995–11005.

Smeulders, A. W. M. et al. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(12), 1349–1380.

Torralba, A., Fergus, R., & Freeman, W. (2008). 80 million tiny images: A large dataset for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(11), 1958–1970.

Vallejo, D., Albusac, J., Jimenez, L., Gonzalez, C., & Moreno, J. (2009). A cognitive surveillance system for detecting incorrect traffic behaviors. *Expert Systems with Applications, 36*(7), 10503–10511.

Vezzani, R., & Cucchiara, R. (2008). ViSOR: VIdeo surveillance on-line repository for annotation retrieval. In *IEEE international conference on multimedia and expo 2008, Hannover, Germany* (Vol. 1(1), pp. 1281–1284).

Xiang, T., & Gong, S. (2006). Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision, 67*(1), 21–51.

Xiong, Z. et al. (2006). Semantic retrieval of video-review of research on video retrieval in meetings, movies and broadcast news, and sports. *IEEE Signal Processing Magazine, 23*(2), 18–27.

Yoo, H. W., Park, H. S., & Jang, D. S. (2005). Expert system for color image retrieval. *Expert Systems with Applications, 28*(2), 347–357.

Zhang, Z., Huang, K., & Tan, T. (2008). Multi-thread parsing for recognizing complex events in videos. In P. Torr & A. Zisserman (Eds.), *10th ECCV, Part III* (pp. 738–751).

Zhu, S., & Liu, Y. (2009). Automatic scene detection for advanced story retrieval. *Expert Systems with Applications, 36*(3), 5976–5986.