

# Article

DOI: 10.1111/j.1468-0394.2010.00578.x

## Multi-agent system for knowledge-based event recognition and composition

Angel Rivas-Casado,<sup>1</sup> Rafael Martínez-Tomás<sup>1</sup> and Antonio Fernández-Caballero<sup>2</sup>

(1) Dpto. Inteligencia Artificial. Escuela Técnica Superior de Ingeniería Informática, Universidad Nacional de Educación a Distancia, Juan del Rosal 16, 28040 Madrid, Spain  
Email: arivas@dia.uned.es

(2) Dpto. Sistemas Informáticos, Escuela Técnica Superior de Ingeniería Informática, Universidad de Castilla-La Mancha, Albacete, Spain

**Abstract:** *This work presents a multi-agent system for knowledge-based high-level event composition, which interprets activities, behaviour and situations semantically in a scenario with multi-sensory monitoring. A perception agent (plurisensory agent and visual agent)-based structure is presented. The agents process the sensor information and identify (agent decision system) significant changes in the monitored signals, which they send as simple events to the composition agent that searches for and identifies pre-defined patterns as higher-level semantic composed events. The structure has a methodology and a set of tools that facilitate its development and application to different fields without having to start from scratch. This creates an environment to develop knowledge-based systems generally for event composition. The application task of our work is surveillance, and event composition/inference examples are shown which characterize an alarming situation in the scene and resolve identification and tracking problems of people in the scenario being monitored.*

**Keywords:** knowledge-based systems, high-level vision, image understanding, video surveillance, event-base system, intelligent agent, plurisensorial agent, Arduino

### 1. Introduction

This work presents a multi-agent system for knowledge-based high-level event composition, which interprets activities, behaviour and situations semantically in a scenario with multi-sensory monitoring. The system's specific aims would be (Bobick, 1997) to monitor the complex scenario, using all the sensors' capacities (Carmona, 2009) to interpret the information from these sources as a whole, and (Chleq & Thonnat, 1996) to request further information to confirm or reject the hypotheses raised in the process for diagnosing the situations.

The semantic interpretation of scenes must recognize situations, activities and interactions between the different actors. The physical signals captured by the sensors must be linked with the interpretation of their meaning. When a human observer interprets the meaning of a scene, he uses his knowledge of the world, the behaviour of the things that he knows, the laws of physics and the set of intentions governing the agents' activity. All this additional knowledge enables the observer to model the scene and use this model to predict, at least partially, what may happen. In other words, to launch a

hypothesis about the evolution of the events and activities that he is detecting. The aim is to aid the human operator so that he can take objective, consistent real-time decisions about the events detected.

Multi-sensory interpretation combines multiple sources of information from different sensors in order to generate a more exact and robust interpretation of the environment (Pavón *et al.*, 2007). The interpretation of behaviour and situations is currently based on using several sensors that offer a high number of false alarms. The availability of new types of sensors for monitoring tasks provides new challenges for multi-sensory data fusion that they solve or they diminish this problem. It is now possible to construct models of the environment and diagnose situations by analysing indefinite sequences of sensory information from different types of sensors. The efficient fusion of multi-sensory information – understood as the fusion of data from several sensors of the same type or, additionally, as the fusion of data from several different types of sensors – is a crucially important task in advanced surveillance. Thus, in recent years there has been a rapid increase in research and development into the combined use of multiple sensors, including vision, audio, heat (thermal), presence (volumetric), vibration sensors, etc. (Zhu & Huang, 2007). A multi-sensory, monitoring system usually requires three components of equal importance (Zhu & Huang, 2007): (1) sensors that capture the information for the monitoring system, process and interpret it; (2) fusion algorithms of data from each of the sensors; (3) architectures for constructing real-time systems.

Of all the sensors, the visual ones are obviously very important. High-level vision is defined as the interpretation of scenes beyond the mere recognition of objects (Fuentes & Velastin, 2004). Image understanding (image comprehension/interpretation) generally starts by analysing movement and then describes the scene with symbols (Tostsos *et al.*, 1980; Levine *et al.*, 1983). It is widely accepted that it is necessary to inject the expert's knowledge to complement the low-level information (Neumann & Novak, 1983; Neumann, 1984).

In a constructivist model, the event composition is done with spatio-temporal relations, with parallel processes that determine the time intervals of other events. Regarding the diagnosis of situations, in Chleq and Thonnat (1996) the hypothesis approach is included, which implies doing parallel explorations for alternative solutions. The confirmation of a hypothesis in a description level helps the previous levels. In Rota and Thonnat (2000), the use of declarative models for representation is described. Our work follows this line, constructing declaratory models from the skill of a human expert who knows how to identify critical situations, particularly in surveillance monitoring problems.

Surveillance is a perfect application field for both plurisensory monitoring and interpreting video-sequence scenes, where our group has accumulated considerable experience with AVISA project works (Folgado *et al.*, 2007; Martínez-Tomás *et al.*, 2008; Carmona, 2009). It is also a multi-disciplinary task affecting an increasing number of scenarios, services and clients. In particular, the use of agents in surveillance systems has some precedents in the bibliography (Abreu *et al.*, 2000; Remagnino *et al.*, 2004; Haesevoets *et al.*, 2007) and the multi-sensory surveillance works by Castanedo *et al.* (2008). In the cooperative sensor agent (CSA) architecture proposed in Molina *et al.* (2004) coalitions are formed between agents to perform surveillance tasks. The CSA is broken down into two levels: sensor layer and coalition layer. A coalition is formed when an agent (sensor) needs to cooperate with other agents that have capacities that it does not have.

The article is structured as follows: the system structure, its organization into vision agents (VAs) and plurisensory agents (PAs), and the high-level composition agent (CA) are in Section 2. The perception agents (VAs and PAs) identify significant changes in the sensors' magnitudes and transmit them as simple events to the CA. From these simple events, the CA composes composed events that imply higher-level semantic activities. The system is described and an application example is given that identifies the abandoning of objects as an alarming situation. Section 3

describes the CA as a knowledge-based system and the structure of its knowledge base. The methodology for creating, updating and modifying the base for different application fields is also described. The article finishes by describing the system's application for monitoring people in different visual fields with different cameras, therefore just visual information. It also describes the problem of access control where plurisensory information is already being used for precise identification and tracking.

## 2. Intelligent agents of the system

The term intelligent agent in artificial intelligence refers to any entity that can take decisions from its environment (Russell & Norvig, 2009). As shown in Figure 1, the system schema proposed consists of one or several PAs, one or several image interpretation agents (IIAs) and in theory, just one CA.

Both the PAs and the IIAs process the sensor information and identify (agent decision system) significant changes in the monitored signals, which the connection interface translates into simple events and sends to the CA via a network connection system. For an efficient system, the agents must be carefully selected and placed by an expert in the best positions of the environment to be monitored.

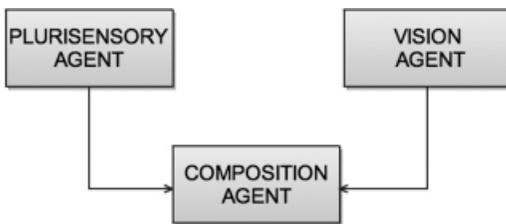
### 2.1. Composition agent

It is the high-level knowledge-based synthetic software agent that identifies activities or situations (composed events) as a composition from simple events that meet certain spatio-temporal restrictions. We can distinguish between two

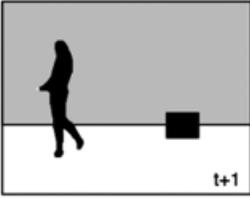
types of functionalities: the reception of simple events generated by the PA and IIA and the composition from these simple events. Defining the event patterns (composed events) that characterize the target activities is an analytical task of knowledge acquisition by a human expert who knows a behaviour pattern's simple actions perfectly. The actions correspond to significant high-level activities and have associated replies. For this, a methodology was created to develop the knowledge base. The generation of an event from the PA and VA implies the instantiation of the corresponding generic event, where values are given to its attributes. For example,  $At(h, x, y, t)$  is instantiated at  $At((h \text{ 'Peter'})(x \ 150)(y \ 120)(t \ 430))$  when Peter is identified as at position (150, 120) at instant 430. For the composition, the events are included as part of the facts base. The scenario ontology also includes facts describing its inherent characteristics, where the activities occur: doors, windows, passages, no entry areas, adjacent cameras, etc. Figure 2 schematically illustrates an example of an alarming situation. A person leaves an object on the ground and goes away. The first row includes simplified images in the scene instants. The second row contains the simple events that are generated from segmenting, monitoring and identifying each frame of the sequence. The third row shows the pattern for the composition (composition unit) of events occurring at each instant. It is a knowledge unit for a composition: a set of events that must meet specific spatio-temporal relations and the consequent events with higher-level semantics. In the event base there is a previous 'Walking' event that completes the pattern. The fourth row shows the higher-level event inferred by this way. 'Walking' belongs to a special type of event that can be called 'event-state' that occur over a time period  $t_1-t_2$ , unlike 'instant-event'.  $At$ , for example, occurs at a given instant  $t$ .

Thus, following the example of Figure 2 we can interpret the sequence as follows, step by step:

1. At instant  $t$ , human1 is detected on the scene. The identification and monitoring



**Figure 1:** Agent interconnection schema.

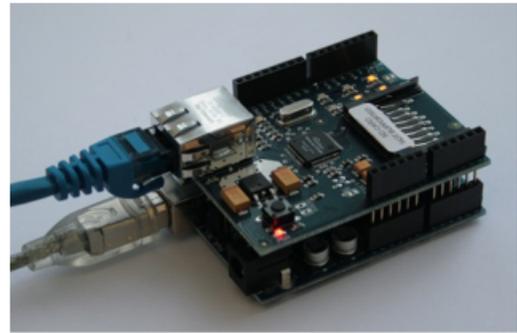
		
$At(\text{"human1"},x,y,t)$	$Appear(\text{"object1"},x",y",t+2)$ $At(\text{"human1"},x,y,t+2)$	$Disappear(\text{"human1"},t+1)$
Walking(h1,x1,y1,t1,x2,y2,t2) At(h2,x3,y3,t3) (h1 == h2) (t2 < t3 == t) => Walking(h1,x1,y2,t1,x3,y3,t2)	Appear(o1 ,x1 , y1 , t1) At(h1, x1, y1, t2) (x1 aprox x2) and (y1 aprox x2) (t1 == t2 == t+2) => Prealarm(h1,o1,t1)	Prealarm(h1, o1, t1) Disappear(h1, t2) (t1 < t2 == t+2) => ALARM(h1,o1)
Walking("human1",x1,y2,t1,x,y,t)	Prealarm("human1","object1",t+1)	ALARM("human1","object1")

**Figure 2:** Recognition that a person has abandoned an object.

agent does not recognize that the human is carrying an object, so the spatio-temporal location of the human is only represented with the event  $At$  in that instant  $t$ . With this  $At$  the event 'Walking2' is updated to the following instant.

- At instant  $t + 1$ , 'object1' is detected near to the position of human1. This situation activates a pre-alarm of possible abandonment of an object with the event 'Pre-alarm'. Since there are no other humans nearby, it is inferred that 'human1' has left the object. An association is created between the object and human, and the pre-alarm is activated.
- At instant  $t + 2$ , 'human1' is detected leaving the scene. This event and the active pre-alarm identify a situation of abandoning an object. The event 'Alarm' goes off.

We group composition units into packages that identify a specific situation. In turn, the packages are organized into composition levels, each package is assigned to a composition level. Each composition level sends the composed events that it has generated to their higher composition level. Packages in different composition levels are inter-dependent. Thus, if a package in a specific level is added to the knowledge base, all those packages in the lower composition levels, which are necessary for its functionality, will be

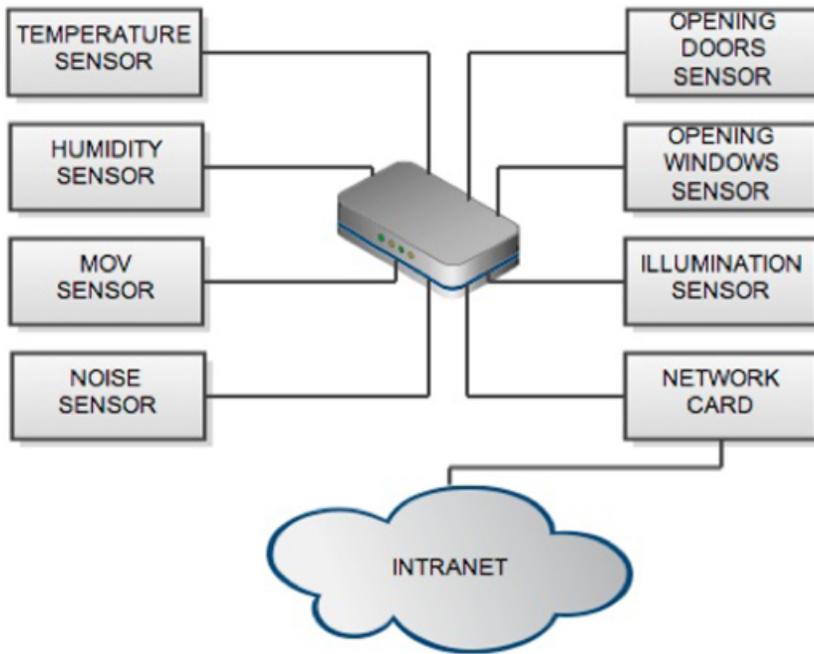


**Figure 3:** Multi-purpose multi-sensory system.

added. Our aim is to give the possibility to create libraries of packages rich enough to configure a system easily. Each library of packages has its own corresponding event ontology.

## 2.2. Plurisensory agent

The incorporated systems provide information about the environment where they are located. They can take a high number of readings per second to pinpoint significant alterations. Generally, they perform very simple operations. They can take, analyse and send samples from several sensors. Figure 3 shows a multi-sensory system consisting of a shield ethernet connected to an Arduino plate with an Atmel AVR 8bits



**Figure 4:** *Multi-sensory system schema.*

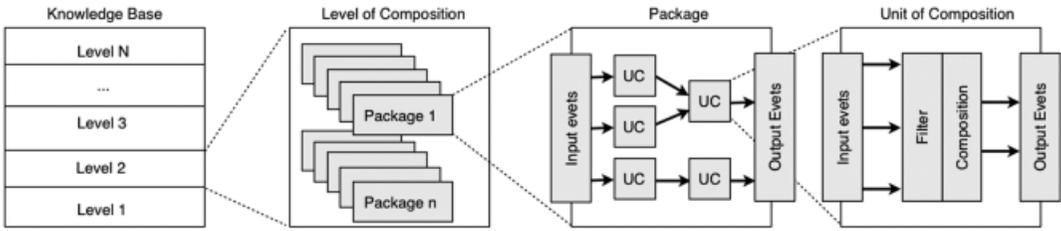
processor. This system can capture, analyse and send information from the sensors once the information has been filtered by the local area network (LAN).

At present, as shown in Figure 4, work is being done on integrating into the multi-purpose multi-sensory system different sensors that are particularly relevant for surveillance problems: movement sensors, proximity sensors, lighting sensors, temperature sensors, relative humidity sensors, open door and window detector, environmental noise sensors, radio frequency identification (RFID)-based identification systems. This multi-purpose multi-sensory system provides a PA with several advantages: its low cost and easy reproduction, it provides reliable, rapid information and finally, it can be used to monitor private rooms, where for legal reasons, video-surveillance systems cannot be installed. We are currently studying the possibility of incorporating IIAs into the system taking advantage of the information received from the PA improve the event segmentation and focusing processes (VA), like,

for example, detecting scene lighting changes or doors opening in the visual field.

### 2.3. Vision agent

This agent combines a system of perceiving images with vision software that requires high computational performance for near real-time tasks. Its main functions are those of a vision system: segmentation, identification and tracking of people and/or objects within a monitored visual field (Carmona, 2009). For the identification process, a block-based system is used (Folgado *et al.*, 2007) developed by our group, the same as the level structure to identify events (Martínez-Tomás *et al.*, 2008). The connection interface can identify a battery of pre-designed events, which are not specific to the application field, but are especially relevant for identifying (composing) the significant high-level events related to the scene events, as shown in Martínez Tomás and Rivas Casado (2009). Section 5 shows an example.



**Figure 5:** Breakdown of the knowledge base. Each knowledge level (abstraction level) consists of a number of packages that in turn have rules. Each rule implements, as composition unit, one or several high-level events pattern.

### 3. Knowledge base structure (KBS)

Figure 5 shows the internal structure of the knowledge base. First of all, we differentiate between levels that correspond to abstraction levels, from knowledge to identify more precise activities, more directly associated with mechanical actions or movements, to more abstract activities, which define behaviour or situations.

The packages are sets of rules (as mechanisms for minimal representation of composition units) that identify specific situations. The concept of package is really important in this schema since the system can add or eliminate these from a knowledge base. Thus there is package inter-dependency. If a package is eliminated from an upper level, all the lower-level packages providing it with information will be automatically eliminated. Obviously, when a package is eliminated, all its composition units are also eliminated. Each package has a work environment defined by the knowledge level to which it belongs, by the input events and the events that it generates. Each package contains one or more rules. The rules only have access to the events defining the package. Thus each rule is encapsulated within the best environment. Errors and possible crossing of information between different packages are also prevented. The advantage of this structure compared with a traditional KBS is that it is easy to maintain and reconfigure when changes occur in the environment. In each level, the rules are prioritized for their execution. The most important rules are the ones that process the information

from the previous level and the least important are those that generate the information that is transferred to the upper level. This process is performed for each knowledge level as a pipeline. The composed events that level  $N$  generates at instant  $T$  are processed in level  $N+1$  at instant  $T+1$ .

#### 3.1. Composition unit

In the prototypes that we have developed, the composition of events is represented on rules, but generically we can speak of composition unit. It is regarded as the unit representing the system's knowledge. It consists of three parts:

- *Antecedents*: It is the event pattern that must be met to be able to check the composition conditions.
- *Conditions*: It is a filter that must be passed to be able to execute the high-level event composition.
- *Actions*: Once all the tests have been passed, the corresponding changes are made in the event base.

New events can be created or one or other of the existing events can be eliminated. To create new events, the event data are taken that are the composition unit antecedents. Similarly, only events can be eliminated that belong to the antecedents. This mechanism is able to fuse simple events into more complex events. In the example below, we can see two events, a simple one and a composed one. The aim is to update the composed event information with the simple

event information:

$$At(h, x, y, t)$$

$$\text{Walking}(h, x1, y1, t1, x2, y2, t2)$$

The event  $At$  shows where a human  $h$  is at instant  $t$  in the position  $x, y$ . The event ‘Walking’ represents at what time and position he began to walk to what time and position he stopped walking. Now the example is presented with the instantiated events:

$$At((h\text{Peter})(x\ 200)(y\ 50)(t\ 500))$$

$$\text{Walking}((h\text{Peter})(x1\ 100)(y1\ 50)(t1\ 450)(x2\ 190)(y2\ 50)(t2\ 499))$$

In the first event it is observed that the human ‘Peter’ is at position  $x=200, y=50$  and  $t=500$ . The conditions would be that the humans were the same person and that instant  $t2$  of the event ‘Walking’ were less than instant  $t$  of the event  $At$ . With these data, the second event could be updated to

$$\text{Walking}((h\text{Peter})(x1\ 100)(y1\ 50)$$

$$(t1\ 450)(x2\ 200)(y2\ 50)(t2\ 500))$$

We now have the event updated to the instant 500, so we can affirm that ‘Peter’ has advanced 100 pixels to the right in 50 instants. After updating all the ‘Peter’ dependent events, the simple event could be erased before executing the following instant.

The knowledge base is structured as an abstraction pyramid (Figure 6). At the bottom is the knowledge level supported by the simple events, which arrive from the PAs and IIAs, and at the top of the pyramid are the upper knowledge levels. To do the event composition in level  $n$  (for  $n > 1$ ), a hypothesis is made, which searches the events verifying this hypothesis in level  $n - 1$ . If the hypothesis’s restrictions are met, the new composed event is generated in

level  $n$ . Configuration parameters are global variables that can be accessed by all the rules. They store a specific value, usually, numerical. They are used to be able to configure the knowledge base in different environments. A video camera’s resolution, for example, the ontology terms like ‘Near’ and ‘Far’ act as configuration parameters.

#### 4. Methodology and development tools

One of the AVISADOS project’s aims was to use it in different application fields to facilitate the configuration for new scenarios, always with the structure described in Sections 2 and 3. For this, a number of tools and similar methodology were developed, which means that each new application does not start from scratch. The

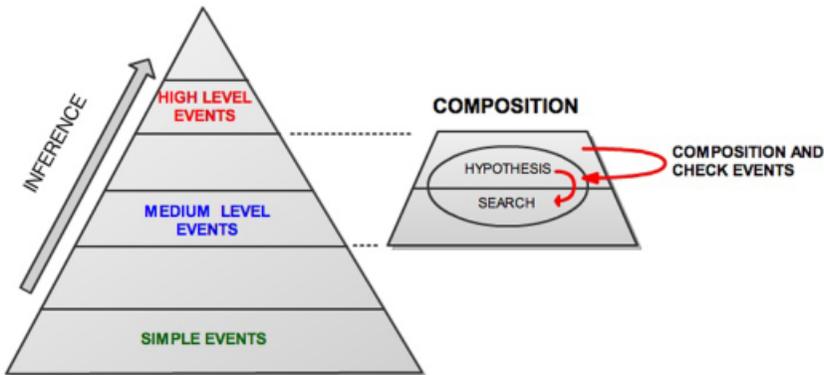
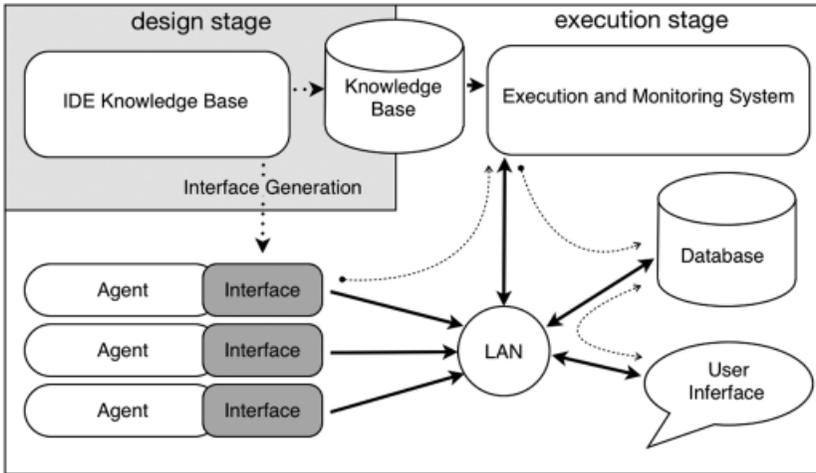


Figure 6: Inference pyramid.



**Figure 7:** *System schema.*

result is a system that generates KBSs in general and interpretation KBSs based on the agent structure in particular.

#### 4.1. Methodological and functional schema

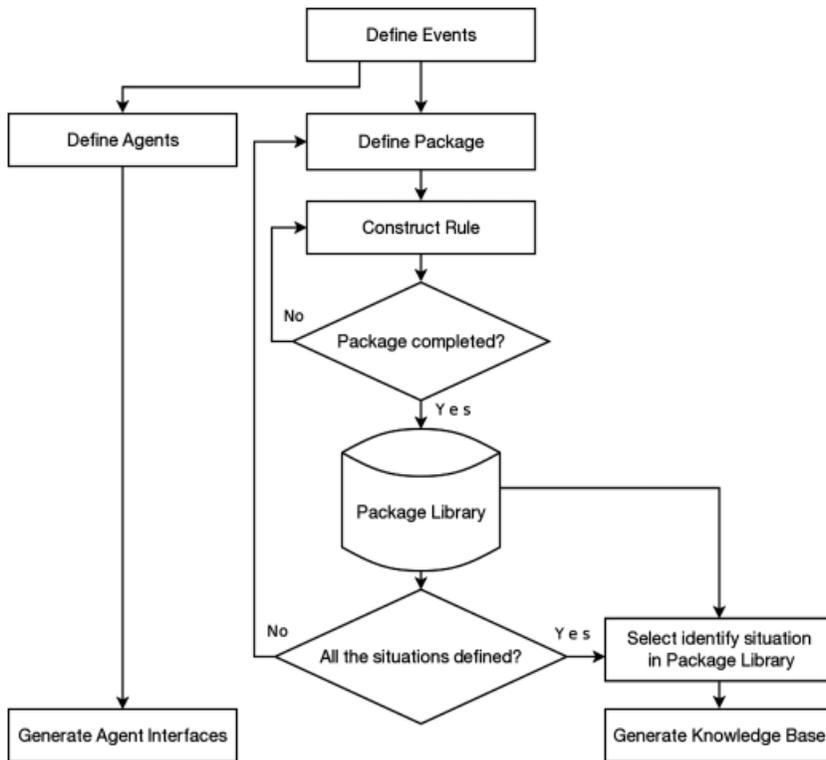
In a design stage, in Figure 7, the event ontology is configured. This will be used later to create agent interfaces and define the knowledge packages and units. Three stages can be identified in the execution stage. In the first, the agents process the environment information and identify the events corresponding to the actions detected. Then the agent interface sends the set of events to the execution and monitoring system (EMS) via a LAN. In the second stage, the events received are labelled with the corresponding instant in the EMS. Then the execution takes place. The inferred events are sent to the database to which the EMS is connected. The last stage is the analysis of the results by the user interface. All the agents are connected to their corresponding interface. This in turn is connected to the EMS. In the following sections, we focus in more detail on the system generating the agent interfaces and the knowledge base.

#### 4.2. Knowledge base IDE

The flowchart of Figure 8 shows the procedure for constructing a knowledge base and the agent interfaces. The development of a new knowledge

base begins by defining the event attributes. Then we implement the simple and composed event ontology that the CAs, PAs and VAs will be able to identify (define events) and that we use to develop the agent profiles (define agent) and to create the knowledge packages (define package). An agent profile contains the characteristic data of this perception agent ‘identifier, position in the scenario, function, description’ and the collection of simple events that it can identify. This information is necessary to create the connection interface (generate agent interfaces), from the perception agent, which sends the simple events perceived to the CA. The package library is a repository of packages of identifiable situations previously implemented. It is not necessary to finish of defining (define package and construct rules) all packages to generate a knowledge base. Through a process of selection, you can configure the knowledge to address different situations (select identify situation in package library and generate knowledge base). So, this process allows for reconfiguring a new system very quickly.

Figure 9 shows the user interface of knowledge base IDE. The list of simple and compound events developed is shown. The left-side bar shows the information about the project and the selected item. This provides expert assistance in designing the knowledge base. The search for elements developed is one of the utilities included in the tool. On the right of the window we can see



**Figure 8:** Flowchart for developing a knowledge (centre), reconfiguration system and generate knowledge base (right).

all elements that form the knowledge base. They are classified into six sections: agents interface, configuration parameters, parameters of the events, events, composition units and packages.

Each package has an associated knowledge level and some simple and/or composed events with which to work. The expert, following the activity identification patterns, generates the right composition units so that the package functions correctly. For this the rule manager is used. First of all, the package must be defined for the rule. Each composition unit has an associated package from which it inherits the set of events to which it may refer in the antecedents, conditions and actions.

### 4.3. Execution system

This tool executes the knowledge base created with the knowledge base IDE, stores all the

composed events generated in a record and connects all the system's agents. It consists of:

- *Network server:* It activates the socket so that the agents connect to the system. The connection is bidirectional to be able to request certain information from the agent.
- *List of connected agents:* A list is kept of all the agents that are connected to the system along with their description and related data.
- *Event synchronization system:* It generates time slices to receive and label events. Once the event has been received, it is labelled with the time associated with the time slice. Thus an asynchronous system becomes a synchronous one.
- *Composition motor:* Inference motor that evaluates the knowledge unit requirements and adds the inferred ones to the event base.
- *Statistic and monitoring system:* It analyses the number of events that arrive from each

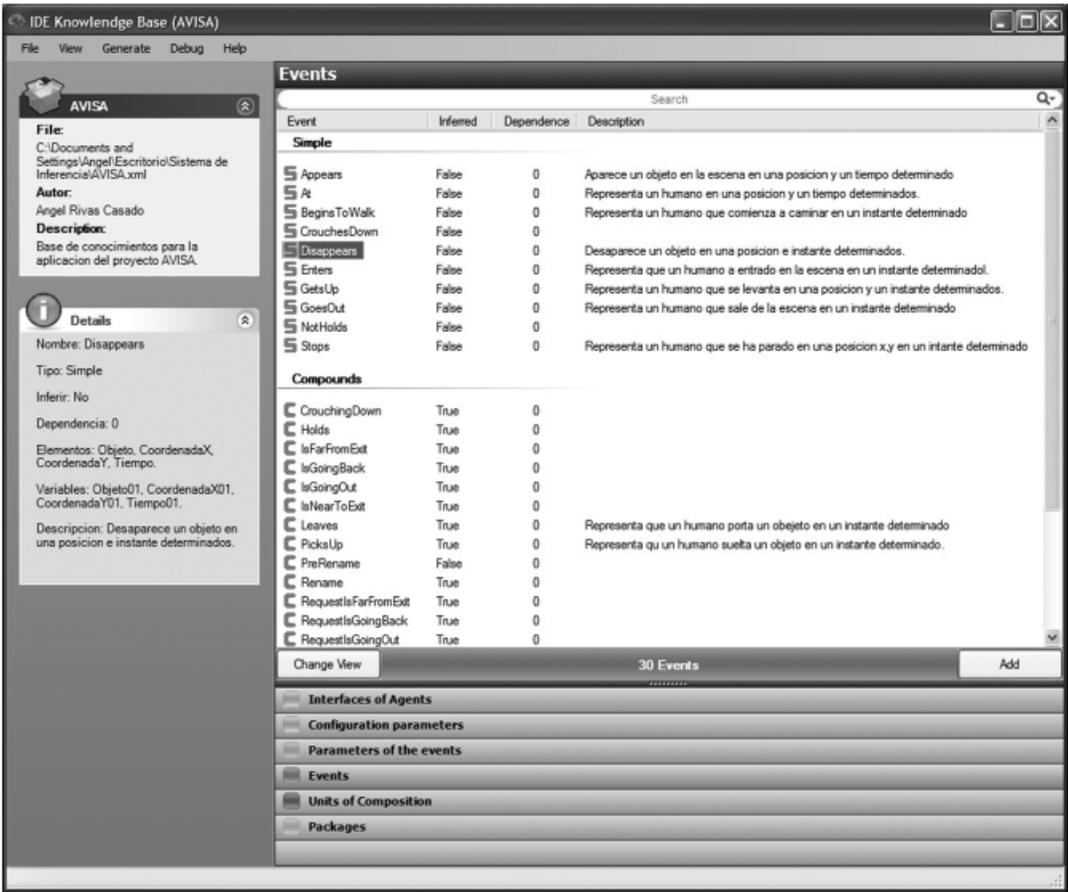


Figure 9: Knowledge base IDE.

agent, the composition motor execution time, the synchronization system buffers and the database connection response times. This information is really useful when calibrating and configuring all the system's execution parameters.

- *Database connection to store the information:* It sends all the events inferred at each instant to the selected database. The information can then be processed later. When these data are analysed, possible errors can be debugged from the knowledge base, which is really useful for refining the system.



Figure 10: Execution and monitoring system.

Figure 10 shows the EMS. This window shows the user the state overall system: server status,

		
$At(c2, h345, x1, y1, t)$	$At(c2, h345, x2, y2, t2)$ $At(c1, h213, x3, y3, t2)$	$At(c1, h213, x4, y4, t3)$
	$At(cam1, hum1, x1, y1, time1)$ $At(cam2, hum2, x2, y2, time2)$ $adjacent(cam1, cam2)$ $(x1 \text{ aprox } x2) \text{ and } (y1 \text{ aprox } y2)$ $(time1 == time2)$ $\Rightarrow$ $Similar(cam1, hum1, cam2, hum2)$	
	$Similar(c1, h345, c2, h213)$	

**Figure 11:** Scenario with overlapping cameras.

connection port, number of clients, state of the system timer, state of inference engine, state of the statistics and the state connection with the database. The red cylinder represents the execution time instant and the green cylinder, average execution time. The bottom bar shows number of the events that are currently on the basis of facts. The left bar buttons are used to change the status of different system functions monitoring and enforcement. The buttons on the top bar show individual from each of the elements system. The upper-left button, displays the system preferences: configuring the connection to the database, maximum number of customers, cycle time implementation, connection port, maximum size of the base facts and the selector of the knowledge base.

## 5. Resolving problems in identifying people and tracking

Section 2 showed identifying an abandoned object as an application to recognize an alarming situation. In this section, a solution is shown for identifying and tracking people, first with adjacent cameras and then with the support of

precise identification from the access control. Both of them are application examples of our event compositions to solving problems for the lower levels events.

### 5.1. Adjacent camera tracking

In the framework of tracking people and objects with artificial vision, the problem is to identify each one in two overlapping or adjacent images. Figure 11 shows a scenario where a person appears in the view angle of two separate cameras. Each of the cameras is calibrated according to the global coordinates. The system receives the positioning events of each person visible in each of the camera's images. If we add the knowledge of the 'Adjacent(c1, c2)' scene to this, we can generate a rule that resolves this problem. The example is shown below whereby the problem would be resolved:

$At(c1, h345, x1, y1)$

$At(c2, h213, x2, y2)$

$Adjacent(c1, c2)$

IF  $(x2 \text{ near } x1) \text{ and } (y2 \text{ near } x1)$



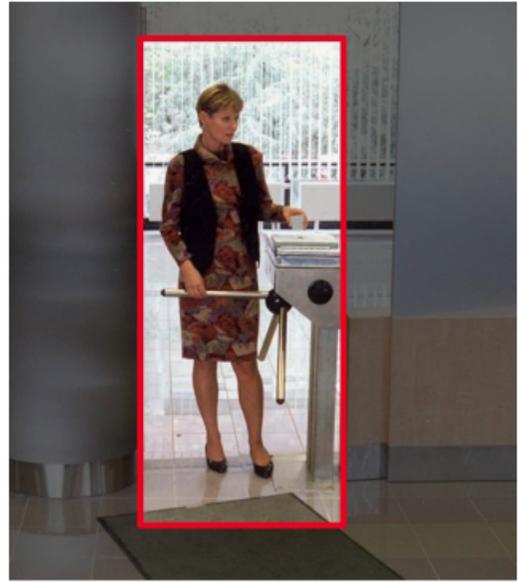
**Figure 12:** *Individual access control.*

Then  $(c1, h345) = (c2, h213)$  and  $(c2, h213) = (c1, h345)$ . Thus it is immediately inferred that the human visible in camera 1 is the same one visible in camera 2.

Figure 11 shows the evolution in time of events and by inference rule described above. In Instanto,  $t$   $h345$  shows a human in a position  $x1, y1$  in camera 2. At time  $t2$  is a  $h345$  human in a position  $x2, y2$  in the camera 2 and in turn a human  $h213$  at position  $x3, y3$  in the house 1. In the knowledge of the scene must be the camera 1 and camera 2 are adjacent. With this background runs the rule displayed in row 3, and it appears that the  $h345$  human camera 2 is the same as the  $h213$  human camera 1 by the event 'Similar'.

### 5.2. Precise identification from the access control

Figure 12 shows an access control RFID identification. Such controls provide, in normal circumstances, a robust system to identify people. One of the problems posed by the artificial vision is the reliable identification of humans within a scene. It is therefore envisaged the merger of the information from the RFID identification system with the identification and tracking information provided by artificial vision. Figure 13 shows an example of agent fusion using the scene knowledge level. It is an RFID card reader access control. A camera with a tracking system monitors people entering. The



**Figure 13:** *Video-surveillance access control.*

CA receives the person's identification information from the RFID card read by the PA. Moreover, the VA detects another person entering the security area and labels him/her with a self-generated ID. When these two items of information reach the CA, it searches for some correlation between the two. The system changes the self-generated ID for the RFID card ID. Then the person can be tracked via the video-camera inter-connection system.

## 6. Conclusions

This work presents a multi-agent system for composition based on high-level knowledge events, which interprets activities, behaviour and situations semantically in a scenario with multi-sensory monitoring. A perception agent (PA and visual agent)-based structure is presented. The agents process the sensor information and identify (agent decision system) significant changes in the monitored signals, which they send as simple events to the CA that searches for and identifies pre-defined patterns like higher-level semantic composed events. This

structure also has a description of the methodology to develop knowledge-based systems to compose events and a set of tools to facilitate its application. The methodology focuses on dividing knowledge, classifying it into types and encapsulating it into what we call knowledge packages, which in turn are organized into abstraction levels. Each knowledge package identifies specific situations or activities from lower-level abstraction events. These ideas were shown using the prototype developed for video surveillance. It consists of two tools: a development interface for a rule-based knowledge base, and an EMS. The first tool aims to configure the pattern of new alarming situations, either based on standard, usual activities (in the knowledge package repository) or directly from identifying and tracking events in the image sequences. These sequences can be labelled with the tool included or the image analysis module. The EMS offers an execution system to facilitate the tasks to analyse the results. Different application examples of the event composition from different perception agents have been shown, which identify alarming situations and resolve problems in identifying and tracking people. The system is particularly useful for surveillance but it could also be applied to other fields. Future works will aim to increase the system's inferential capacities. Obviously, it is necessary to bear in mind and process the sources of indetermination in each abstraction level, and particularly the identification and tracking ones.

## Acknowledgements

The authors are grateful to the CiCYT for financial aid on project TIN-2007-67586-C02-01.

## References

- ABREU, B., L. BOTELHO, A. CAVALLARO, D. DOUX-CHAMPS, T. EBRAHIMI, P. FIGUEIREDO, B. MACQ, B. MORY, L. NUNES, J. ORRI, M. TRIGUEIROS and A. VIOLANTE. (2000) A video-based multiagent traffic surveillance system, in *The IEEE Intelligent Vehicles Symposium* 45-462.
- BOBICK, A. (1997) Movement, activity, and action: the role of knowledge in the perception of motion, *Royal Society Workshop on Knowledge-based Vision in Man and Machine*, London, pp. 1257-1265.
- CARMONA, E.J. (2009) On the effect of feedback in multilevel representation spaces, *Neurocomputing*, **72**, 916-927.
- CASTANEDO, F., J. GARCÍA, M.A. PATRICIO and J.M. MOLINA (2008) A multiagent architecture to support active fusion in a visual sensor network, in *2nd ACM/IEE International Conference on Distributed Smart Cameras*, Stanford University, CA, USA, pp. 1-8.
- CHLEQ, N. and M. THONNAT (1996) Realtime image sequence interpretation for video-surveillance applications, *IEEE International Conference On Image Processing (ICIP'96)*, Laussane, pp. 800-804.
- FOLGADO, E., M. RINCÓN, E.J. CARMONA and M. BACHILLER (2007) A block-based model for monitoring of human activity, Technical Report AVISA-12-07.
- FUENTES, L.M. and S.A. VELASTIN (2004) Vigilancia avanzada: del tracking a la detección de sucesos, *IEEE América Latina*, **2**, 206-211.
- HAESEVOETS, R., B. VAN EYLEN, D. WEYNS, A. HELLEBOUGH and T. HOLVOET (2007) Context-driven dynamic organizations applied to coordinated monitoring of traffic jams, *Engineering Environment-Mediated Multiagent Systems*, Dresden, Germany, 1-5 October, 2007, D. Weyns, S. Brueckner and Y. Demazeau (eds.), pp. 126-143.
- LEVINE, M.D., P.B. NOBEL and Y.M. YOUSSEF (1983) A rule-based system for characterizing blood cell motion, in *Image Sequence Processing and Dynamic Scene Analysis*, T.S. Huang (ed.), Berlin, Germany: Springer-Verlag, 663-709.
- MARTÍNEZ-TOMÁS, R., M. RINCÓN-ZAMORANO, M. BACHILLER-MAYORAL and J. MIRA-MIRA (2008) On the correspondence between objects and events for the diagnosis of situations in visual surveillance tasks, *Pattern Recognition Letters*, **29**, 1117-1135.
- MARTINEZ TOMÁS, R. and A. RIVAS CASADO (2009) Knowledge and event-based system for video-surveillance tasks, in *Proceedings of the 3rd International Work-Conference on the Interplay Between Natural and Artificial Computation (IWINAC): Part I: Bioinspired Applications in Artificial and Natural Computation*, ISBN:978-3-642-02263-0, J. Mira, J. M. Ferrández, J.-R. A. Sanchez (eds.), Berlin: Springer-Verlag, 386-394.
- MOLINA, J.M., J. GARCÍA, F.J. JIMÉNEZ and J.R. CASAR (2004) Fuzzy reasoning in a multiagent system of surveillance sensors to manage cooperatively the sensor-to-task assignment problem, *Applied Artificial Intelligence*, **18**, 673-711.
- NEUMANN, B. (1984) Natural language description of time-varying scenes. Brericht no. 105, FBI-HH-B-105/84, Fachberic Informatik, University of Hamburg.

- NEUMANN, B. and H. NOVAK (1983) Events models for recognition and natural language description of events in real-world image sequences, *Proceedings of the Eighth IJCAI, Karlsruhe*, Morgan Kaufmann, San Mateo, California, pp. 724–726.
- PAVÓN, J., J.J. GÓMEZ-SANZ, A. FERNÁNDEZ-CABALLERO and J.J. VALENCIA-JIMÉNEZ (2007) Development of intelligent multi-sensor surveillance systems with agents, *Robotics and Autonomous Systems*, **55**, 892–903.
- REMAGNINO, P., A.I. SHIHAB and G.A. JONES (2004) Distributed intelligence for multicamera visual surveillance, *Pattern Recognition*, **37**, 675–689.
- ROTA, N.A. and M. THONNAT (2000) Video sequence interpretation for visual surveillance, in IEEE International Workshop on Visual Surveillance (VS'00), Dublin, Ireland, pp. 59–68.
- RUSSELL, S. and P. NORVIG (2009) *Artificial Intelligence: A Modern Approach*, 3rd edn, Pearson: Prentice Hall.
- TOSTSOS, J.K., J. MYLOPOULOS, H.D. CORVEY and S.W. ZUCKER (1980) A framework for visual motion understanding, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, 563–573.
- ZHU, Z. and T.S. HUANG (eds), (2007) *Multimodal Surveillance: Sensors, Algorithms and Systems*, Artech House Publishers.

## The authors

### Angel Rivas-Casado

Angel Rivas-Casado received his degree in technical engineer in computer science of systems from the University Polytechnical of Madrid, Spain, in 2009. At present, he attends a master of advanced artificial intelligence in the National University for Distance Education

(UNED) in Madrid, Spain. His research interests are in robotics, knowledge engineering, artificial vision, sensors fusion and intelligent agents.

### Rafael Martínez-Tomás

Rafael Martínez-Tomás received his degree in physics from the University of Valencia, Spain, in 1983, and received his PhD from the department of artificial intelligence of the National University for Distance Education (UNED) in Madrid, Spain, in 2000. Since 2001, he is an associate professor with the department of artificial intelligence at the UNED. His research interests are in knowledge engineering, knowledge-based systems, spatial-temporal logics, description logics and video-sequence interpretation.

### Antonio Fernández-Caballero

Antonio Fernández-Caballero received his degree in computer science from the Technical University of Madrid, Spain, in 1993, and received his PhD from the department of artificial intelligence of the National University for Distance Education, Spain, in 2001. Since 1995, he is an associate professor with the department of computer science at the University of Castilla-La Mancha, Spain. His research interests are in image processing, computer vision, neural networks and agent technology. A. Fernández-Caballero is member of the IAPR.