Activity Recognition from Video Sequences using Declarative Models

Nathanaël A. Rota and **Monique Thonnat** [1]

Abstract. We propose here a new approach for video sequence interpretation based on declarative models of activities. The aim of the video sequence interpretation is to recognize incrementaly certain situations, like states of the scene, events and scenarios, in a video stream, in order to understand what happens in the scene. The input of the activity recognition is an *a priori* model of the scene and human tracked in it. The activity recognition is composed of two subproblems. First, end-users have to declare all the activities in a configuration phase. Secondly, the declared models must be automatically recognized. To solve the first problem, we propose a homogeneous declarative formalism to describe all the activities (states of the scene, events and scenarios). The activities are described by the conditions between the objects of the scene. To solve the second problem, we translate it into a constraints satisfaction problem. Then, we use a classical CSP algorithm to recognize the activities in video sequences. Finally, we present some results to show the robustness of the approach.

1 Introduction

This paper presents recent work in video understanding in the context of visual surveillance applications [2]. The aim is to incrementaly recognize certain situations, like states of the scene, events and scenarios, in a video stream, in order to understand what happens in the scene. The input of the activity recognition is an *a priori* model of the scene and human tracked in it. The activity recognition is composed of two subproblems. First, end-users have to declare all the activities in a configuration phase. Secondly, the declared models must be automatically recognized. The classic approach consists in two levels of reasoning. The first level is the event level representing significant changes in the state of the scene and the second level is the scenario level, the aim of which is to recognize predefined long term situations modeled with combinations of events. We propose here a new approach based on a homogeneous formalism of the heterogenous concepts involved in this problem. That is to say that there is only one formalism to declare any kind of concepts (events and scenario) associated to only one recognition algorithm. In the section 2 we will see that the major part of work done in this domain stays on two-level-understanding. Then, in section 3, we introduce our approach. In section 4, we propose a formalism for the description of the concepts involved in activity recognition. In section 5, we give more details on the method of activity recognition. Finally, we present some resultsto show the robustness of the approach.

2 State of the Art

Cohen, Bremond Medioni and Nevatia, in DARPA's VSAM focus on event recognition involving vehicles and humans [15] and [7]. The particularity of this work is that videos are filmed by non-fixed camera. They used models of maps of the environment to place aerial images in an *a priori* known map. They used a property net to compute events and states, which controls the evolution of predefined automaton describing situations. Herzog proposes a system able to dynamically describe scene with humans. The originality of his work is the application environment: a soccer stadium [1] and [9] and the inference method based on time interval logic, to describe temporal sequence of events, which are computed and typed separatly. Intille and Bobick, in a similar environment, focus on analysis of American football scenes. Their aim is the recognition of particular strategies in the complex players' interactions [11], [10] and [12]. The main point is that those activities are not just human behaviors but human group behavior. Shah is interested by dynamic description of human behaviors in office environments [2] and [8]. Even if the problem is the recognition of long duration activities, the authors insist on the importance of the recognition of 'key instants' which are the conditions of changing states in an automaton representing the global behaviors. Kittler et al. [14] and Christensen et al. [6] in VAP project propose to analyse scene evolution for activities like breakfast table setting. Scene states are described by multiples cues (colours, motion, shape) and typical sequences of events are modelled by a grammars. Thonnat and Rota [18] propose a method based on both n-ary tree to declare events and temporal logics to declare scenario in the context of visual surveillance. Tessier , in the PERCEPTION project, proposes an original method to describe behavior. Petri nets are used to represent dynamic evolutions of a car park scene with humans and vehicles [17] and [5]. Buxton and Gong gave an important contribution to the domain with the VIEWS project [4] and [3]. The system was able to deal with humans and vehicles on roads, streets or in car parks. A high level representation based on Bayesian networks was computed. This work points out the necessity to deal with uncertainty and to use contextual information to enhance detection and tracking results. In the same vein, Ivanov and Grimson work on detection of human and vehicle behaviors in car park. The interest of this research is in the event's combination method [13]. A behavior is represented by a set of rules based with a stochastic context-free grammar, which allows certain combinations of simple constant predicates. The main point is that there is no global formalism for interpretation. All work describe systems using a couple of formalism. The first to compute more or less complex logical predicates and the second to manage the temporal problem with Automaton, Graphs, Petri Nets or Grammars.

[1] Projet ORION INRIA Sophia, 06902 BP 93 Sophia Antipolis cedex France, email: firstname.lastname@sophia.inria.fr

[2] Work done under support from Dyade GIE INRIA Bull

3 The approach

The approach we propose is based on the managment of a set of elements called facts, which represent differents kind of concepts, we want to recognize and to store. This notion groups very different concepts like detected persons, predefined areas and equipments, states of the scene, interesting events or long term scenarios. A fact is a structured object defined by seven sets of attributes: $name$, $type$, $date$, $geometry$, $velocity$, $properties$ and $reference$. The attribute $name$ is a symbolic identifier of the fact. The attribute $type$ is a symbolic value representing the category of the fact. This value can be $person$, $area$, $equipment$, $state$, $event$ or $scenario$. The attribute $date$ is a numerical value representing a time associated with the fact. The attribute $geometry$ is a list of points in 3D space describing the volume of the fact if it's necessary. The attribute $velocity$ is a 3D vector representing the estimated velocity of a fact if it makes sense.The attribute $properties$ is a list of symbolic values describing characteristics of the fact. The attribute $references$ is a list of pair $name/date$ usefull to associed a fact to another one.

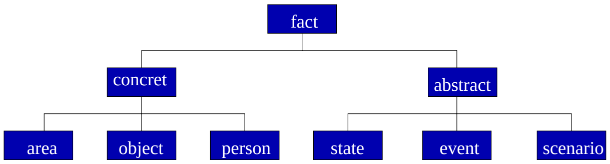

Figure 1. Hierarchy of facts

We organize those concepts in a hierarchy, as shown in figure 1. The set of all facts is divided into two sets: the concrete facts and the abstract facts. Concrete facts represent real-world objects which are given as input of the activity recognition system by a perception system. Abstract facts represent proprerties of the scene computed by our activity recognition system.

The types of concrete facts are $person$, $equipment$ or $area$. The fact $person$ is computed at each image frame by a Perception system. We consider that two facts $person$ come from the same real person if they have the same $name$. The $geometry$ and $velocity$ attributes of a fact $person$ can vary at each time. The facts $equipment$ and $area$ must be predefined in a configuration phase. They represent the knowledge of the scene. We consider here that $geometry$ and $velocity$ attributes of an $equipment$ or an $area$ are constant in time and the velocity attributes are null.

The types of abstract facts are $state$, $event$ or $scenario$. A $state$ represents a partial description of the scene at time t. An $event$ represents a significant change in the scene and a $scenario$ represents a long term situation. A $state$ is defined by a list of concrete facts and some spatial conditions. An $event$ is basically a pair or a triplet of $states$ related to the same concrete facts at different times. A $scenario$ is any combination of $state$ and $event$.

For example, f_1, f_2, f_3, f_4 and f_5 define respectivelly the person $person\ 2$, an equipment $ticket\ machine$, the state $person\ 2\ is\ far\ from$ from $ticket\ machine$, the event $person\ 2\ moves\ close\ to\ ticket\ machine\ 1$ and the scenario $Vandalism\ against\ ticket\ machine$ by $person\ 2$ on $ticket\ machine\ 1$.

$$\begin{array}{ll}
name(f_1) = person\ 2 & name(f_2) = ticket\ machine\ 1 \\
type(f_1) = person & type(f_2) = equipment \\
date(f_1) = 34 & date(f_2) = 34 \\
geometry(f_1) = & geometry(f_2) = \\
\quad ((10, 10, 0), & \quad ((20, 20, 0), \\
\qquad \vdots & \qquad \vdots \\
\quad (80, 80, 180)) & \quad (120, 120, 200)) \\
velocity(f_1) = (45, 12, 0) & velocity(f_2) = (0, 0, 0) \\
properties(f_1) = NO & properties(f_2) = fragile
\end{array}$$

$$\begin{array}{ll}
name(f_3) = is\ far\ from & name(f_4) = moves\ close\ to \\
type(f_3) = state & type(f_4) = event \\
date(f_3) = 34 & date(f_4) = 35 \\
references(f_3) = & references(f_4) = \\
\quad ((person\ 2, 34), & \quad ((person\ 2, 35), \\
\quad (ticket\ machine\ 1, 34)) & \quad (ticket\ machine\ 1, 35))
\end{array}$$

$$\begin{array}{l}
name(f_5) = Vandalism\ against\ ticket\ machine \\
type(f_5) = scenario \\
date(f_5) = 103 \\
properties(f_5) = Alarm \\
references(f_5) = \\
\quad ((person\ 2, 103), (ticket\ machine\ 1, 103))
\end{array}$$

Based on this approach, the problem we address in order to perform activity recognition is first to define the conditions of existence of each abstract fact. Then to recognize and to verify those conditions at each time. In the two following sections we present how to describe an abstract fact and then how to recognize it.

4 Description of abstract facts

We address in this section the problem of the description of the abstract facts representing a certain situation. The main idea is to define a formalism with needed variables representing needed facts and forbidden variables representing forbidden facts and to declare possible values for each of them. We describe a fact f with three sets: the variable set, the condition set and the production set. The variable set is a set of variables representing a fact required by the formalism of f. Those variables are typed with a binary value: needed or forbbiden. A needed variable represents a fact which must occur and a forbiden variable represents a fact which must not occur for the regular recognition of f. The condition set is a set of predicates involving the attributes of the facts in the variable set. Finally, the production set is a set of functions which enable us to compute the attributes.

$$\begin{array}{l}
\textbf{Variables} \\
x_1 : +, \ldots, x_k : +, x_{k+1} : -, \ldots, x_n : - \\
\textbf{Conditions} \\
\forall j \in \{1, \ldots, p\}\ c_j(x_1, \ldots, x_n) \ = \ true \\
\textbf{Production} \\
\forall l \in \{1, \ldots, m\}\ g_l(x_1, \ldots, x_k)
\end{array}$$

where $x_1, \ldots, x_n$ are variables involved in the **Conditions** and the **Production**(needed with a +, forbidden with a -), c_j are the predicates of conditions , g_l are the functions of production of the jth. attribute of f. We define the cardinality of a description by the number of variables in the variable set.

The figures 2, 3 and 4 are three examples of models of abstract fact: a state $is\ far\ from$, an event $moves\ close\ to$ and a scenario

Variables: $x_1 : +,\ x_2 : +$
Conditions:

$$\begin{cases} type(x_1) & = person \\ type(x_2) & = equipment \\ distance(x_1, x_2) & \geq \alpha_{is\ far\ from} \end{cases}$$

Production: x_3

$$\begin{cases} name(x_3) & = is\ far\ from \\ type(x_3) & = state \\ date(x_3) & = date(x_1) \\ reference(x_3) & = (name(x_1), name(x_2)) \end{cases}$$

Figure 2. Model of fact: *is far from*

Variables: $x_1 : +,\ x_2 : +,\ x_3 : -$
Conditions:

$$\begin{cases} type(x_1) = type(x_2) = state \\ type(x_3) = event \\ name(x_1) = is\ far\ from \\ name(x_2) = is\ close\ to \\ date(x_1) \leq date(x_3) \leq date(x_2) \\ date(x_2) = date(x_1) + \Delta \\ name(x_3) = moves\ close\ to \\ reference(x_1) = reference(x_2) \end{cases}$$

Production: x_4

$$\begin{cases} name(x_4) = moves\ close\ to \\ type(x_4) = event \\ date(x_4) = date(x_2) \\ reference(x_4) = reference(x_2) \end{cases}$$

Figure 3. Model of fact: *moves close to*

Variables: $x_1 : +,\ x_2 : +,\ x_3 : +,\ x_4 : -$
Conditions:

$$\begin{cases} type(x_1) = type(x_2) = type(x_3) = type(x_4) = event \\ name(x_1) = name(x_3) = moves\ close\ to \\ name(x_2) = name(x_4) = moves\ away\ from \\ date(x_1) \leq date(x_2) \leq date(x_3) \leq date(x_4) \\ date(x_2) \leq date(x_3) - \Delta_1 \\ date(x_3) \leq date(x_4) - \Delta_2 \\ reference(x_1) = ((ticket\ Machine, *), *) \\ reference(x_1) = reference(x_2) = reference(x_3) \\ reference(x_1) = reference(x_4) \end{cases}$$

Production: x_5

$$\begin{cases} name(x_5) = Vandalism\ against\ ticket\ Machine \\ type(x_5) = scenario \\ date(x_5) = date(x_4) \\ reference(x_5) = reference(x_3) \end{cases}$$

Figure 4. Model of fact: *Vandalism against ticket Machine*

Vandalism against ticket machine. For example, the model of *is far from* is defined with two needed variables x_1 and x_2 respectively of type *person* and *equipment* such that the euclidian distance between them is less or egal to a certain constant α.

With this method, we defined a list of 15 states: *is stopped*, *walks*, *runs*, *arrives*, *goes away*, *goes right side*, *goes left side*, *falled*, *standing*, *is close to*, *is far from*, *is inside*, *is outside*, *walk together* and *walk to*. The 15 models which define those states are similar to the model of *is far from* (cf. figure 2) except that the 7 first have only one needed variable of type *person* and the 8 last have two needed variables. In *walks*, *runs*, we put a condition on the norm of the velocity vector. In *arrives*, *goes away*, *goes right side*, *goes left side*, we put a condition on the angle between the velocity vector and a predefined direction. In *falled* and *standing*, we put a condition on the height of *person*. In *is close to*, *is far from*, *is inside*, *is outside*, we put a condition on the euclidian distance between the two involved concrete facts. In *walk together* and *walk to*, we put a condition on the angle between the velocity vector and another vector dependant on the second concrete fact.

We have defined furtive events and persistent events. The furtive events are defined with a pair of different states at different instants, such that both have the same references. The persistent events are defined with a triplet of states at different instants, such that all have the same references; as for furtive events the two first states are different but the name of the third state is the same as the name of the second state. The list of events we are able to recognize is: *falls down*, *stands up*, *turns right side*, *turns left side*, *turns back*, *stops*, *starts*, *moves close to*, *moves away from*, *enters*, *leaves*, *sits on*, *appears* and *disappears*.

The events *falls down* and *stands up* are based on *falled* and *standing*. The events *turns right side*, *turns left side*, *turns back* are based on *arrives*, *goes away*, *goes right side* and *goes left side*. The events *stops* and *starts* are based on *is stopped* and *walks*. *moves close to*, *moves away from* and *sits on* are based on *is close to* and *is far from*. *enters* and *leaves* are based on *is inside* and *is outside*. The events *appears* and *disappears* are directly defined on concrete *person* facts.

We have defined three categories of scenario like *Vandalism*, *Access forbbiden area*, *Holdup*. The models of those scenarios are defined with the help of human security experts. The *Vandalism* and *Access forbbiden area* scenarios were built during the european project AVS-PV [3] by operators of metro station of Nuremberg, Brussels and Charleroi. The *Holdup* scenarios have been recently defined with the help of bank security experts of FNCA [4].

5 Recognition of abstract facts

In this section we are interested in the second side of the problem; how to recognize incrementaly and efficiently predeclared models of fact.

We have for this a set of models $\{M_1, \ldots, M_m\}$ and a set of facts F_t. The models came from the modelling of the user's knowledge and the set F_t is formed with the concrete facts until t and the abstract facts until $t-1$. The aim is, for each model, to analyse F_t to know if some new facts must be created.

For example, if we want to create at t a fact *Vandalism against ticket machine* shown in figure 4, we have to found in current F_t two facts named *moves close to* and

[3] Advanced Video Surveillance for Prevention of Vandalism
[4] Federation Nationale du Credit Agricole, France

one $moves\ away\ from$ corresponding to the three needed variables of this model, such that each of them verify the specified conditions and we must not found a second $moves\ away\ from$ corresponding to the forbidden element x_4, such that this one verify its conditions.

To understand what can be the difficulties of a such recognition, let's define the boolean problem $P_0(M, A, F)$ where F is a set of facts, A is an ordered subset $\{f_1, \dots, f_k\}$ of F and M a model defined by (cf 4). We say that $A = \{f_1, \dots, f_k\}$ is a solution of P_0 if and only if $\neg\exists x_{k+1} \in F, \dots, \neg\exists x_n \in F\ c_j(f_1, \dots, f_n) = true$ $\forall j \in \{1, \dots, p\}$.

Now, the problem of the recognition of a fact defined by a model M at t with a set of facts F_t is be formalismed as a problem $P(M, F_t)$ such that the solution of $P(M, F_t)$ is the set of all the ordered subsets $F_{i,t}$ of F_t such that $F_{i,t}$ is a solution of $P_0(M, F_{i,t}, F_t)$. The number of ordered subsets $F_{i,t}$ of F_t is exponential in fonction of the cardinality of M, then the problem P cannot be simply solved.

We propose in the following, a method limiting the combinatorial explosion of the resolution of P. For this we will note in the following the model M like $(E, I, C_E \cup C_I, F)$ where E is the set of needed facts, I is the set of forbidden facts, $C_E = \{c_1, \dots, c_p\}$ is the set of conditions involving only needed facts, $C_I = \{c_{p+1}, \dots, c_n\}$ is the set of conditions involving at least one forbidden fact and F is the production set.

Theorem 1 **if** *$P(M_1 = (E, \emptyset, C_E \cup \emptyset, F), F_t)$ has at least one solution* **and** *$P(M_2 = (E \cup I, \emptyset, C_E \cup C_I, F), F_t)$ has no solution* **then***$P(M_3 = (E, I, C_E \cup C_I, F), F_t)$ has at least one solution*

Proof:

By definition $P(M_1 = (E, \emptyset, C_E, F), F_t)$ has at least one solution
$\Rightarrow \exists f_1, \dots, \exists f_k, \backslash$
$\forall j \in \{1, \dots, p\}\ c_j(f_1, \dots, f_k) = true$ **(1)**
By definition $P(M_2 = (E \cup I, \emptyset, C_E \cup C_I, F), F_t)$ has no solution
$\Rightarrow\exists f_1, \dots, \exists f_k, \exists f_{k+1}, \dots, \exists f_n \backslash$
$\exists j \in \{1, \dots, n\}\ c_j(f_1, \dots, f_n) = false$ **(2)**
(1) and **(2)** $\Rightarrow \exists f_{k+1}, \dots, \exists f_n \backslash$
$\exists j \in \{p+1, \dots, n\}\ c_j(f_1, \dots, f_n) = false$
$\Rightarrow \neg\exists f_{k+1}, \dots, \neg\exists f_n \backslash$
$\forall j \in \{p+1, \dots, n\}\ c_j(f_1, \dots, f_n) = true$ **(3)**
(1) and **(3)** $\Rightarrow \exists f_1, \dots, \exists f_k, \neg\exists f_{k+1}, \dots, \neg\exists f_n, \backslash$
$\forall j \in \{1, \dots, n\}\ c_j(f_1, \dots, f_n) = true$
$\Rightarrow P(M_3 = (E, I, C_E \cup C_I, F), F_t)$ has at least one solution

Then to find the solution of any $P = (M = (E, I, C_E \cup C_I, F), F_t)$ is to find the solutions of $P_1 = (M_1, F_t)$ and $P_2 = (M_2, F_t)$ such that $M_1 = (E, \emptyset, C_E, F), F_t)$ and $M_2 = (E \cup I, \emptyset, C_E \cup C_I, F), F_t)$. If they both have no solution or both have solution then P has no solution, but if P_1 has at least one solution and P_2 has no solution then P has at least one solution. It means that if P_1 has only one solution P has the same, but if P_1 has more than one solution we must verify each of them.

Now, we have to solve P_1 and P_2. As M_1 and M_2 have no forbidden fact, these problems can be translated into a double contraints solving problem (CSP). We define $P_1^* = (V_{P_1^*}, D_{P_1^*}, K_{P_1^*})$ and $P_2^* = (V_{P_2^*}, D_{P_2^*}, K_{P_2^*})$ such that, $V_{P_1^*} = E$, $D_{P_1^*} = F_t$, $K_{P_1^*} = C_E$, $V_{P_2^*} = E \cup I$, $D_{P_2^*} = F_t$, $K_{P_2^*} = C_E \cup C_I$.

The elements of E and $E \cup I$ belong to F_t (cf. 4). It means that the elememts of E and $E \cup I$ should be seen as sets of variables with values in F_t. It means that the elememts of E and $E \cup I$ be seen as sets of variables and F_t as the domain of elements of E and $E \cup I$. Futhermore, the elements of C_E and $C_E \cup C_I$ are predicates involving the elements of E and I, so they can be seen as set of discrete constraints. So, $P_1^* = (V_{P_1^*}, D_{P_1^*}, K_{P_1^*})$ and $P_2^* = (V_{P_2^*}, D_{P_2^*}, K_{P_2^*})$ are standard constraints problem solving with discrete values.

This transformation from $P((E, I, C_E \cup C_I, F), F_t)$ to the pair of CSP $P_1^* = (V_{P_1^*}, D_{P_1^*}, K_{P_1^*})$ and $P_2^* = (V_{P_2^*}, D_{P_2^*}, K_{P_2^*})$ is a polynomial time transformation. We solve P_1^* and P_2^* with the arc concistency algorithm AC4 detailed on [16] .

6 Quantitative results

We present in this section quantitative results about the performance of our method in term of noise resistance on a selected set of events. The protocol used to test the noise resistance w.r.t. a certain model is to take an ideal set of facts F_t, which generates the represented fact and to add a gaussian noise to these facts F_t. From the ideal set F_t, we only corrupt the $geometry$ attributes of the involved $person$ and we recompute the $velocity$ attribute. We tested each fact 20 times per variance and for 30 different values of variance. The results are organized in the following table. The first column is the list of facts defined in 4; (f) represents a furtive event model and (p) represents a persistent event model. Column 2 (resp. 3, 4 and 5) represents the percentage of facts recognized at the right time with a max error corresponding to 2 % (resp, 5 %, 10 % and 20 %) of the average size of the scene.

Fact's Name	2%	5%	10%	20%
(f) Stops	95	66	44	28
(p) Stops	95	60	33	17
(f) Starts	88	61	39	26
(p) Starts	88	60	39	26
(f) Turns Left	86	57	47	27
(f) Turns Back	95	63	34	18
(f) Falls Down	100	99	90	72
(p) Falls Down	100	100	79	48
(f) Stands Up	65	40	25	15
(p) Stands Up	65	40	25	15
(f) Enters Area	90	70	63	52
(p) Enters Area	85	68	61	56
(f) Leaves Area	76	50	27	14
(p) Leaves Area	58	32	17	9
(f) Moves Close To (Eq.)	98	82	57	37
(p) Moves Close To (person)	90	72	49	26
(f) Moves Away From (Eq.)	83	66	48	30
(p) Moves Away From (Eq.)	83	66	57	40
(f) Moves Away From (person)	86	66	43	26
(p) Moves Away From (person)	88	82	48	25

7 Example of recognition

The following example (figures 5, 6, 7 and 8) is an example of recognition of the scenario $Vandalism\ against\ ticket\ machine$. Each figure is divided in two images, the left image is the real image taken from CCTV network of Nuremberg Metro Station, the right image is the symbolic reconstruction; The different types of facts are drawn with different colours: $area$ in dark grey, $equipment$ in light grey, $event$ or $scenario$ in white and finally $persons$ with a dark cylinder.

8 Conclusion

We have proposed in this paper a formalism for video understanding. This problem has two subproblems: the first problem is to describe

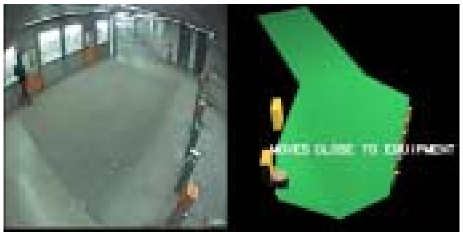

Figure 5. The same person moves close to the ticket machine: A fact $event$ named $moves\ close\ to\ equipment$ is created

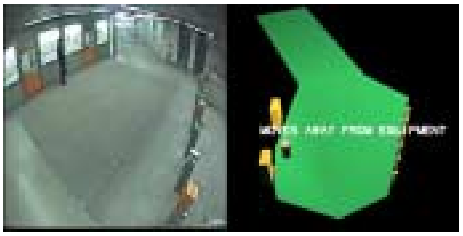

Figure 6. The person moves away from the ticket machine: A fact $event$ named $moves\ away\ from\ equipment$ is created

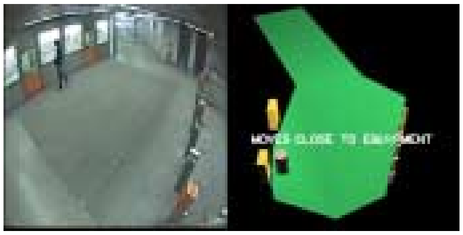

Figure 7. After a short period, the person comes back to the ticket machine: A fact $event$ named $moves\ close\ to\ equipment$ is created

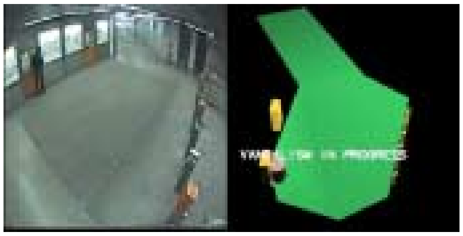

Figure 8. There is now two events $moves\ close\ to\ equipment$ and one $moves\ away\ from\ equipment$ and all the conditions between them are verify: A fact $Vandalism\ against\ ticket\ machine$ is created.

the models of concepts we want to recognize and the second is to recognize the concepts we described. Even if it's easier to describe models if concept are clearly differenciated, the recognition of those models can be less efficient if they are not based on the same formalism. That why, we have proposed in this paper a homogeneous formalism to describe the heterogenous concepts involved in activity recognition, like states of the scene, significant events or long terms scenario. Doing this, we can keep the semantic differenciation of the concepts to declare them, but the homogenity of the formalism enables us to propose a unique method to recognize any kind of concepts. We have tested this method on the recognition of differents models of facts in order to determinate the noise resistence of each of them.

REFERENCES

[1] E. André, G. Herzog, and T. Rist, 'On the simultaneous interpretation of real world image sequences and their natural language description: The system soccer', in *8th European Conference of Artificial Intelligence*, pp. 449 – 454, Munich, (1988).

[2] D. Ayers and M. Shah, 'Monitoring human behavior in an office environment', in *Computer Society Workshop on Interpretation of Visual Motion*, (1998).

[3] H. Buxton and S. Gong, 'Advanced visual surveillance using bayesian networks', in *Workshop on Context-based Vision*, Cambridge, (1995). IEEE.

[4] H. Buxton and S. Gong, 'Visual surveillance in dynamic and uncertain world', *Artificial Intelligence Journal*, **78**, 431 – 459, (1995).

[5] C. Castel, L. Chaudron, and C. Tessier, 'What is going on ? a high level interpretation of sequences of images', in *4th European Conference on Computer Vision, Workshop on Conceptual Descriptions from Images*, Cambridge UK, (April 1996).

[6] H.I. Christensen, J. Matas, and J. Kittler, 'Using grammars for scene interpretation', in *International Conference on Image Processing*, (1996).

[7] I. Cohen and G. Medioni, 'Detecting and tracking moving objects for video surveillance', in *Computer Vision and Pattern Recognition*, Fort Collins, Colorado, (June 1999).

[8] S. Dettmer, A. Seetharamaiah, L. Wang, and M. Shah, 'Model-based approch for recognizing human activities from video sequences', in *Workshop on Motion of Non-Rigid and Articulated Objects*, (June 1998).

[9] G. Herzog, 'Utilizing interval-based event representation for incrementatal high-level scene analysis', in *4th International Workshop on Semantics of Time, Space, and Movement and Spatio-Temporal Reasoning*, Chateau de Bonas, France, (1992).

[10] S. Intille and A. Bobick, 'Visual tracking using closed-world', Technical report, M.I.T Media Laboratory Perceptual Computing Section, Cambridge, MA 02139, (November 1994).

[11] S. Intille and A. Bobick, 'Closed world tracking', in *5th International Conference on Computer Vision*, Cambridge, (1995).

[12] S. Intille and A. F. Bobick, 'Visual recognition of multi-agent action using binary temporal relations', in *Computer Vision and Pattern Recognition*, Fort Collins, Colorado, (June 1999).

[13] Y. Ivanov, C. Stauffer, A. Bobick, and W.E. Grimson, 'Video surveillance of interactions', in *2nd International Workshop on Visual Surveillance*, pp. 82 – 89, Fort Collins, Colorado, (June 1999).

[14] J. Kittler, J. Matas, M. Bober, and L. Nguyen, 'Image interpretation: Exploiting multiple cues', in *International Conference on Image Processing and Applications*, Edinburgh, (June 1995).

[15] G. Medioni, I. Cohen, F. Brémond, S. Hongeng, and R. Nevatia, 'Event detection and analysis from video streams', in *DARPA Image Understanding Workshop*, Monterey, (November 1998).

[16] R. Mohr and T. C. Henderson, 'Arc and path consistency revised', *Artificial Intelligence*, **28**, 225 – 233, (1986).

[17] C. Tessier, 'Reconnaissance de scènes dynamiques à partir de données issues de capteurs: le projet perception', Technical report, Onera-Cert, 2 avenue Edouard-Belin BP 4025 31055 Toulouse Cedex France, (Août 1997).

[18] M. Thonnat and N. Rota, 'Image understanding for visual surveillance applications', in *Third International Workshop on Cooperative Distributed Vision*, Kyoto, Japan, (November 1999).