GENERATION OF 3D ANIMATIONS FROM SCENARIO MODELS

Jihène Bannour, Benoît Georis, François Brémond and Monique Thonnat
INRIA Sophia-Antipolis, ORION team
2004 route des Lucioles, BP 93
06902 Sophia Antipolis Cedex - France
firstname.surname@sophia.inria.fr

ABSTRACT

The objective of this work is to generate automatically 3D animations from scenario models defined by human experts for scenario recognition. These animations are useful for two reasons. First, they can help experts to validate their scenario models. Second, synthetic video sequences can be generated from these 3D animations and serve as reference data to evaluate video interpretation algorithms. An animation is generated in two stages. First, we transform a scenario model into a visualisation model. Second, we generate animation instances from visualisation models. The first results obtained are promising. We have automatically generated 3D animations for two applications: the monitoring of a bank agency and a metro platform.

KEY WORDS

3D animation, scenario models, evaluation of video interpretation algorithms, video interpretation, synthetic movies.

1 Introduction

For many years now, there has been a widespread interest in automatic video interpretation and several works focus on semantic human behaviour recognition [1], [2]. Our goal is the development of a platform able to recognize human behaviours predefined by experts of the application domain (e.g, a bank security director) [3]. Based on our experience, there are several difficulties for experts to provide an accurate definition of a scenario model at once. For instance, they can miss important information or they are not able to express their needs clearly. Most of the time, it is an iterative process, i.e they are able to refine their models once they see a demonstration of the interpretation system.

The automatic generation of 3D animations can help to improve the knowledge acquisition process. Experts will be able to visualise the result of their scenario modelling through 3D animations. They will validate their models, add or delete data which seem necessary or useless. The automatic generation of 3D animations is also a mean to evaluate video interpretation algorithms. Synthetic video sequences can be generated from these 3D animations in order to test and assess the reliability of video interpretation algorithms. Many videos can be created in order to highlight under several configurations a currently unsolved problem for a given video processing algorithm and thus to test robustness of this algorithm under these various configurations.

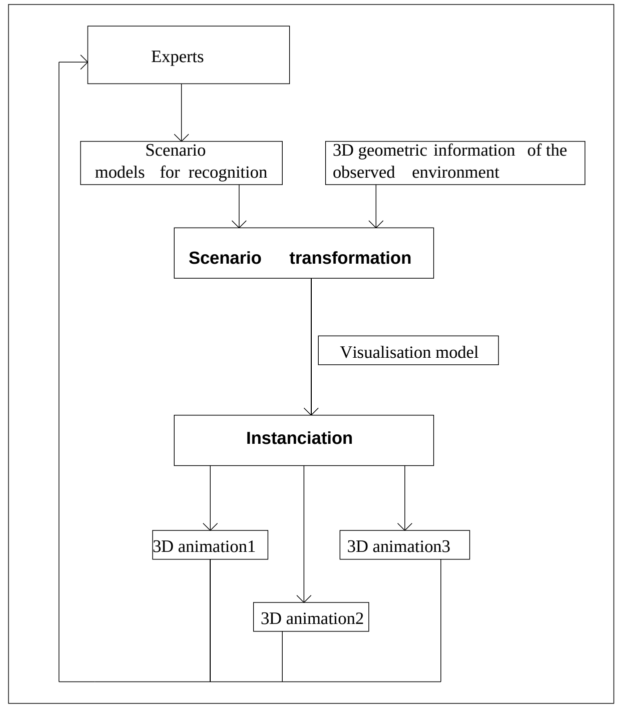

Figure 1. Global system

We suggest two stages to generate 3D animations from scenario models as shown in figure 1. The first stage transforms a scenario model (information given by experts like actions, states, constraints, environment) into a visualisation model needed to generate 3D animations. We propose in this paper a method allowing this transition which we call "scenario transformation". A second stage generates scenario instances from a visualisation model: we initialise the different parameters needed to generate 3D animations. This second stage is called instanciation and is responsible for parameter initialisation, trajectory generation and rendering issues.

The paper is organized as follows: section 2 presents a state of the art in scenario models and 3D scene visualisation. Section 3 describes the scenario representation formalism used by a human expert. Section 4 describes the method used to transform a scenario model defined by human experts for scenario recognition into a visualisation model needed to generate 3D animations. Section 5 presents the method used to generate scenario instances from a visualisation model. Section 6 presents results we have obtained in a bank agengy and discusses the expert feedback on these first obtained 3D animations. Finally, section 7 concludes and indicates future work.

2 Related Work

For several years, the problem of 3D animations has been approached. There are many works on 3D scene visualisation. For example at the faculty of Computer Science of Toronto university [4], reseachers generate 3D animations where fishes and a swimmer evolve in the sea. They have particularly modelled individual fish behaviours and their interactions in a group. Moreover, Craig.W.Reynolds has presented various solutions for autonomous characters in animation and games [5]. He has described several steering behaviours such as seek, flee, pursuit, evasion and obstacle avoidance. Thalmann et al [6] created behavioural animation characters who navigated down corridors and around obstacles. At the Computer Lab of the Swiss Technology Institute of Lausanne [7], [8] reaseachers have modelled individuals in a museum, in a street and in a supermarket. They have also modelled the reaction of people in a fire situation. Several works on 3D animation related to robotics have also been done. We can quote [9] which explains the method of potentiel field used to generate trajectories.
These works have obtained many results in the domain of 3D animations from a scene description.
The contribution of our work is the introduction of a visualisation model which plays the role of a bridge between scenarios defined by experts for scenario recognition and the generation of 3D animations. The content of this model is closely related to expert needs and to parameters needed to evaluate video interpretation algorithms.

3 Scenario Representation

This section is devoted to the description of scenario models given by human experts. We use a representation formalism which takes into account the application domain expert knowledge and a language to describe scenarios [10].
A scenario can be of different types and composed of states and events. A state is a spatio-temporal property defined at one time instant or on a time interval. An event is one or several change(s) of states at two successive time instants or on a time interval. Scenarios can be either primitive (single state change) or composite (combination of states and events). They are described by the following three parts:

- Physical objects: all real world objects present in the scene observed by cameras. A physical object can be a mobile (person, car) or a static object of the environment. A static object is defined by a priori knowledge and can either be a zone of interest or a piece of equipment. A zone is represented by its vertices and a piece of equipment is represented by its 3D bounding box.
- Components: list of states and events involved in the scenario.
- Constraints: relations between physical objects and/or components.

```
primitive event  move_close_to
physical_objects:
  (p: person) (eq : equipment)
components:
   (c1: primitive_state  far_from  (p, eq))
   (c2: primitive_state  close_to (p, eq))
constraints:
   (c1 before c2) (c1 duration <=2)
```

Figure 2. Example of a primitive event

```
primitive state  inside_zone
physical_objects:
  (p: person) (z: zone)
constraints:
  (p in z)
```

Figure 3. Example of a primitive state

An example of a primitive event and a primitive state are shown respectively in figure 2 and figure 3.

4 Scenario Transformation

This section describes the methodology used to transform scenario models defined by expert into visualisation models. We first explain the whole process and then we illustrate the method with two examples: a simple scenario model *(move_close_to)* and a more complex one *(vandalism)*.

4.1 Methodology to transform scenarios

The idea of scenario transformation is to transform the scenario model into a graph. The graph nodes correspond to components. A component could be a state, an event or a sub scenario. Each component is modelled by two nodes. One node corresponds to the starting state and the other one to the ending state. The two nodes of a component are connected by an oriented arc giving information about the duration in time of the related component. The value of this arc is initialised to]0 , oo[, as we first suppose that each component has a duration in time that goes from

0 to infinity. Once we have nodes as shown in figure 4, the study of constraints imposed by experts allows a reduction of the interval size and a connection of different component nodes by choosing the right value of the arc connecting them. In our scenario models constraints are represented with the following operators : before, during, start, finish, overlap, meet and before_meet (before or meet).
For instance, constraints: *(c1 meet c2)* and ($c1\_duration$ $<=$ 1) give the graph transformations presented in figure 4.

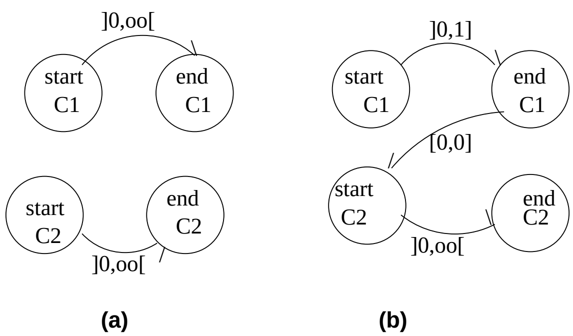

Figure 4. Example of graphical presentation : Before (a) and after (b) studying constraints *(c1 meet c2)* and ($c1\_duration <= 1$)

In a last step, we transform each arc into a linear equation between the two related nodes. For instance, with reference to figure 4 the equation relative to the arc connecting start c1 and end c1 is:
end c1 = start c1 + a; a belonging to]0,1].
Once we have gone through all arcs we thus obtain a system of linear equations related to time instants.
For key positions, the idea is also to transform scenario models into a system of linear equations related to components. Components are represented by the following operators : far from, close to, inside zone, outside zone. So if we note p1 the first physical object of the component, p2 the second one and pos_p1/pos_p2 the respective positions we have the following linear equations:

- far from (p1, p2) is equivalent to
  pos_p1 = pos_p2 + dist; $dist > Mindist$.
- close to (p1, p2) is equivalent to
  pos_p1 = pos_p2 + dist; $dist < Maxdist$
- inside zone (p1, p2) is equivalent to
  pos_p1 = center_zone + dist ; $dist < Maxdist$.
- outside zone (p1, p2) is equivalent to
  pos_p1 = boundary_zone + dist ; $dist > Mindist$.

We define our visualisation model as containing :

- a linear system related to time instants.
- a linear system related to positions.
- trajectories.
- 3D geometry information of the observed environment.
- rendering parameters such as contrast, illumination level, position and number of lights,...
- person speed.

Our visualisation model can be modified (augmented) according to the expert feedback.

4.2 Simple scenario example

We consider in this example the following scenario of a person who moves close to an equipment as shown in figure 2.
The corresponding equation system related to constraints is:

$$S1 = \begin{cases} \text{end\_c1 = start\_c1 + A; } A < 2 \\ \text{start\_c2 = end\_c1 + B ; } B > 0 \end{cases}$$

The corresponding equation system related to components is:

$$S2 = \begin{cases} \text{pos\_p\_c1 = pos\_eq + dist ; } dist > Mindist. \\ \text{pos\_p\_c2 = pos\_eq + dist ; } dist < Maxdist. \end{cases}$$

pos_eq is given by a priori knowledge containing the 3D geometry and semantic information of the observed environment.

4.3 Complex scenario example

In this example, an expert has modelled the behaviour of a vandalism act. A person approaches an equipment, then moves away from it and finally approaches it again (see figure 5). This back and forth movement corresponds to the fact that the vandal can be disturbed by another person when performing the vandalism act. The corresponding equation system related to constraints:

$$S = \begin{cases} \text{start\_e2 = end\_e1 + A ; } A >= 0 \\ \text{start\_e3 = end\_e2 + B ; } B >= 0. \\ \text{start\_e4 = end\_e3 + C ; } C >= 0. \\ \text{start\_e5 = end\_e4 + D ; } D >= 0. \end{cases}$$

The corresponding equation system related to components:

$$S = \begin{cases} \text{p\_e1 = pos\_eq + dist ; } dist < Maxdist. \\ \text{ip\_e2 = pos\_eq + dist ; } dist > Mindist. \\ \text{fp\_e2 = pos\_eq + dist ; } dist < Maxdist. \\ \text{ip\_e3 = pos\_eq + dist ; } dist < Maxdist. \\ \text{fp\_e3 = pos\_eq + dist ; } dist > Mindist. \\ \text{ip\_e4 = pos\_eq + dist ; } dist > Mindist. \\ \text{fp\_e4 = pos\_eq + dist ; } dist < Maxdist. \\ \text{p\_e5 = pos\_eq + dist ; } dist < Maxdist. \end{cases}$$

we note ip_ek and fp_ek the initial and final person positions related to the event ek.

```
scenario vandalism
  physical_objects:
     (p: person) (eq : equipment)
  components:
     (e1: primitive_event  move_close_to  (p, eq))
     (e2: primitive_event  stay_at (p, eq))
     (e3: primitive_event  move_away_from  (p, eq))
     (e4: primitive_event  move_close_to  (p, eq))
     (e5: primitive_event  stay_at (p, eq))
  constraints:
     (e1 before_meet e2) (e2 before_meet e3)
     (e3 before_meet e4) (e4 before_meet e5)
```

Figure 5. Vandalism scenario

5 Instanciation

This section describes how to generate scenario instances from the visualisation model we have defined. We have to choose key positions, time instants, a suitable trajectory and some rendering parameters.
The variability we have when choosing these parameters allows us the generation of several 3D animations for the same scenario model.

5.1 Key positions and time instants

Systems related to time instants and positions defined in the visualisation model have several solutions but for each animation we have to select only one solution. The idea is to choose for each equation a variable element value belonging to the fixed interval and to solve the system by a classical method such as Pivot Gauss method. Solutions are time instants and key positions. For instance, one transformation of the system S1 defined in sub section 4.2 is:

$$S1 = \begin{cases} \text{end\_c1 = start\_c1 + 1.} \\ \text{start\_c2 = end\_c1 + 10.} \end{cases}$$

5.2 Trajectory generation

In the previous sub section, we have shown how to extract time instants and key positions from linear systems defined in the visualisation model. In this sub section, we briefly describe the method used to generate trajectories taking as starting point this numerical information.
The method used is based on the potential field method. On one hand, obstacles (chair, table,...) which can be present in the 3D information of the observed environment apply a repulsive force on the person(s) of interest we have to animate. We can also have walls. In such a case, to avoid the local minimum problem and as person(s) sometimes has/(ve) to go along the wall, the idea is to change the person's direction once he/she approaches a wall as shown in figure 6. On the other hand, the arrival position or goal applies an attractive force which tends toward zero when the person approaches the goal. Moreover, in a great number of scenario models there are more than one person. To avoid collision problems between persons, each person applies a repulsive force with relatively low intensity to others.
Finally, we must deal with cases where a person must follow another one (a robbery scenario for example). Having the same starting and arrival positions does not lead necessarily to a follow-up event. We have thus added a constraint that (if we note P1 the person to follow and P2 the person who follows P1) P2 must have at any time instant t nearly the same position that P2 had at t-a ($a > 0$).
In conclusion, for the same key positions and time instants found in the previous sub section, we are able to generate several trajectories by changing the starting directions of persons acting in the scene or selecting different values for the parameters involved (e.g., repulsive force intensity).

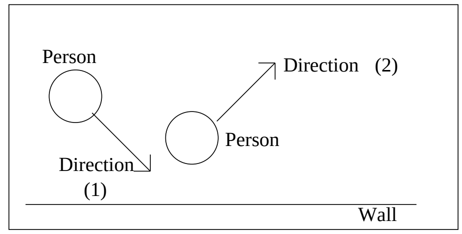

Figure 6. Person direction when approaching a wall

5.3 Rendering parameters

Another degree of freedom when generating several animations corresponding to the same scenario model is to modify rendering parameters such as contrast and lights. Rendering parameters have a great influence to obtain realistic animations and in consequence to generate complex synthetic video sequences. This complexity and variability is needed to assess performances and robustness of video interpretation algorithms. For instance, a synthetic video sequence composed of several people moving and crossing each other while having completely different colours will not represent a great difficulty for a tracking process. At the opposite, the tracking would probably fail on a synthetic video sequence composed of people of the same colour. Contrast has also a great influence on people detection, a weak contrast often causes detection problems. Shadows or illumination changes are among other examples.

Having numerous video sequences which can serve as ground truth data to evaluate video processing algorithm results is of prime importance. First, it eases the time consuming process of manual video sequence annotation to get ground truth data. Second, it enables experts to thoroughly understand current algorithm problems, to diagnose them and to develop new algorithms which are more and more

scenario

physical_objects:
((employee: Person), (robber: Person), (z1: Entrance),
(z2: Back_Counter), (z3: Infront_Counter), (z4: Safe))

components:
((c1: primitive_state inside_zone(employee, z2))
(c2: primitive_event changes_zone(robber, z1, z3))
(c3: primitive_event changes_zone(employee, z2, z4))
(c4: primitive_event changes_zone(robber, z3, z4)))

constraints:
((c2 during c1)
(c2 before c3)
(c1 before c3)
(c2 before c4)
(c4 during c3))

Figure 7. Bank scenario

robust.

6 Results and Discussion

In the bank agency, we obtained the 3D animations as shown in figure 8 taking as input the scenario model given in figure 7. It is a bank agency robbery. The two animation examples are different since we modified the trajectory of the two people acting in the scene by changing their initial direction but in the meantime we kept the same key positions and time instants (solution of the equation systems related to components and constraints, respectively).

For our visualisation model, an expert feedback on a scenario model of a robbery in a train has led to the addition of a speed parameter in the visualisation model. The expert has noticed when seeing 3D animations generated from the model which he/she has defined that the speed of the robber was too slow and thus too far from reality.
More results can be seen on the web page http://www-sop.inria.fr/orion/personnel/Benoit.Georis/english.html.

7 Conclusion and Future Work

In this paper, we have presented a methodology to generate 3D animations from scenario models defined by human experts for scenario recognition. We have insisted on the two most important stages needed to solve this problem: scenario transformation and instanciation. Experts, thanks to this method, are able to validate scenarios they have defined. This is an important result since it improves the knowledge acquisition process both in time and accuracy. Future works will focus more on rendering issues and animation realism. This will allow us a better evaluation of video interpretation algorithms. For instance, we will take into account the person's posture (raising of the hand, sitting, walking, running,...).

References

[1] V.Vu, F.Bremond, and M.Thonnat, "Video interpretation: human behaviour representation and on-line recognition," in *The Sixth International Conference onKnowledge-Based Intelligent Information and Engineering Systems*, (sophia antipolis, France), 2002.

[2] M.Arens and H.-H.Nagel, "Behavioral knowledge representation for the understanding and creation of video sequences," in *Proceedings of the 26th German Conference on Artificial Intelligence*, (Hamburg, Germany), September 2003.

[3] B.Georis, F.Bremond, M.Maziere, and M.Thonnat, "A video interpretation platform applied to bank agency monitoring," in *Intelligent Distributed Surveillance Systems Workshop*, (London, UK), February 23 2004.

[4] D. Terzopoulos, "Artificial life for computer graphics," in *Communication of the ACM, 42(8)*, pp. 32–42, 1999.

[5] C. W.Reynolds, "Steering behaviours for autonomous characters," in *Sony Computer Entertainment America*, (Foster City, California), June 1 1999.

[6] Thalmann, Daniel, Olivier, and M.-T. Nadia, "A vision-based approach to behavioral animation," in *Journal of Visualization and Computer Animation* (J. of Visualisation and C. Animation, eds.), pp. 18–21, 1990.

[7] L.Bezault, R.Boulic, and N.M.Thalmann, "An interactive tool for the design of human free walking trajectories," in *Computer animation*, 1992.

[8] R.Boulic, D.Thalmann, and N.M.Thalmann, "A global human walking model with real-time kinematic personification," in *Visual Computer*, 1990.

[9] H.Hadded, M.Khatib, S.Lacroix, and R.Chatila, "Reactive navigation in outdoor environments using potentiel field," in *Robotique and Artificial Intelligence*, (Toulouse , France), 1998.

[10] V.Vu, F.Bremond, and M.Thonnat, "Temporal constraints for video interpretation," in *The 15-th European Conference on Artificial Intelligence*, (sophia antipolis, France), 2003.

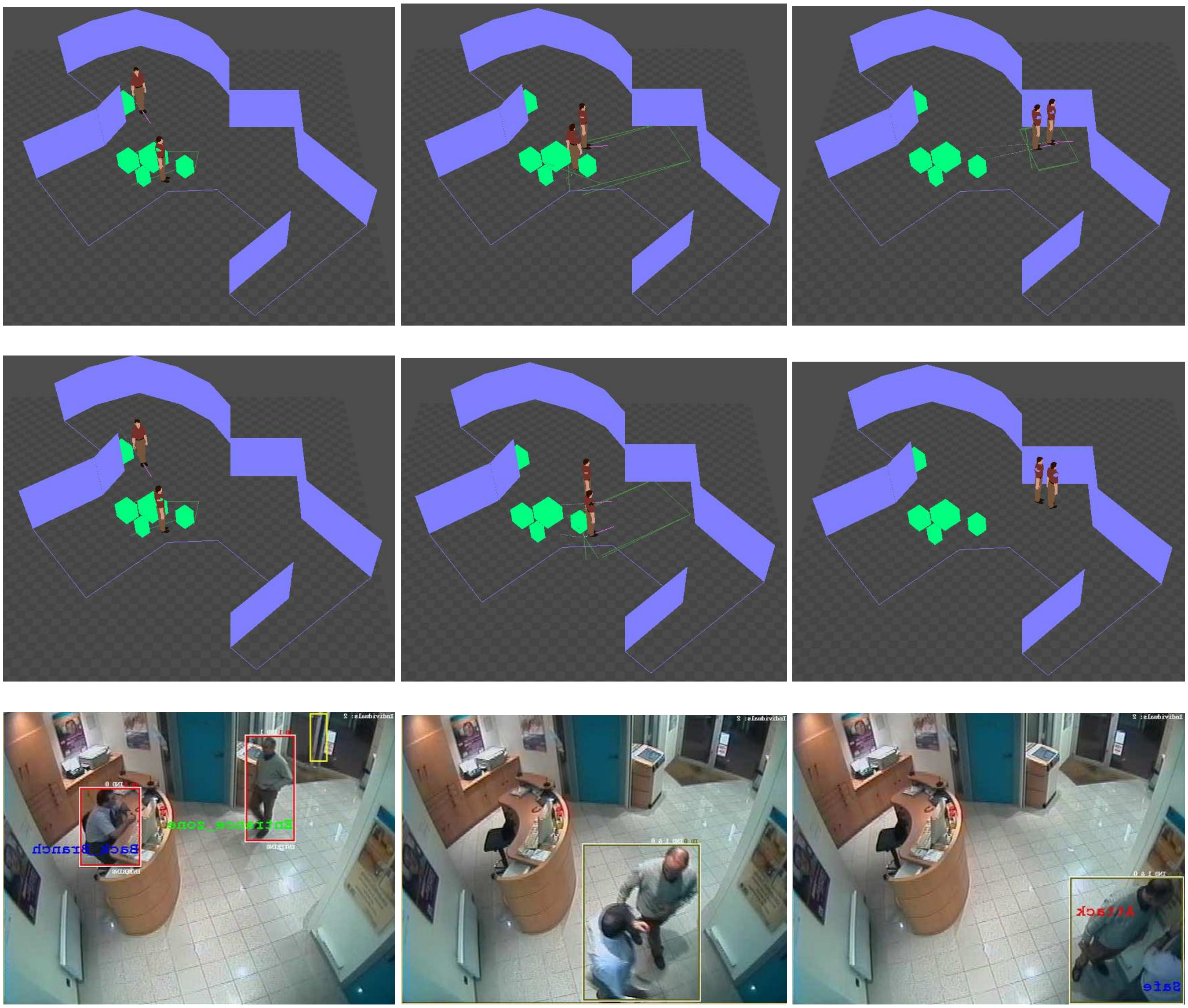

Figure 8. Two 3D animations from the same visualisation model and a corresponding video