Visual Perception for Virtual Agents

Environment Model and Perception's Architecture for Multi-Agent Based Simulation in Virtual Environments: Application to a Pedestrian Simulation

Nicolas GAUD — Stéphane GALLAND — Abder KOUKAM

Multiagent Systems and Applications Group,
Systems and Transportation Laboratory (SeT),
University of Technology of Belfort-Montbéliard (UTBM),
F-90000 Belfort, France

{*nicolas.gaud,stephane.galland,abder.koukam*}*@utbm.fr*

ABSTRACT. *This document presents a environment model for a multiagent system in Virtual Environment. This environment was originally designed to provide a fast visual perception for each agent of the simulation. Our objective was to develop a specific synthetic vision system for autonomous agent in virtual reality. After a short review of the state of art, we focus on some aspect of synthetic vision and virtual environment modeling. Then we present our model and some experimental results obtained on a simple application of crowd simulation. This environment is similar to an interface between a virtual reality platform and a classical multiagent-based simulation platform.*

RÉSUMÉ. *Cet article présente un modèle d'environnement dédié à la simulation multi-agent en environnement virtuel. Cet environnement fut à l'origine conçu afin de guarantir une perception visuelle compatible avec les contraintes temps réel pour tous les agents impliqués dans la simulation. Ce modèle est inspiré des méthodes de vision synthétiques et de modélisation d'environnement virtuel. Après un rapide état de l'art de ces deux domaines, nous présenterons notre modèle d'environnement et quelques résultats expérimentaux obtenus sur un exemple simple d'application de simulation de foules. Cet environnement peut être considéré comme un module d'interface permettant d'immerger une plateforme multiagent classique dans un environnement virtuel quelconque.*

KEYWORDS: *Multiagent Systems, Virtual Reality, Visual Perception, Environment Model*

MOTS-CLÉS : *Systèmes multi-agents, Réalité virtuelle, Perception visuelle, Modèle d'environnement*

1re soumission à *VIA'2004*, le January 22, 2013.

1. Introduction

Virtual worlds offer today a new and efficient environment for the multiagent-based simulation (MABS).

But immersing a multiagent system (MAS) inside a virtual environment (VE) is not an easy task. This type of environment imposes real time constraints. Moreover to assure high-level behaviors for the agents, the VE must integrate several semantical and symbolical informations in addition to the geometry. The perception, especially the vision, is one of the main information channels between the virtual agents and their VE. Indeed, to assure the immersion of a MAS in a VE, each agent should perceive its environment and reacts under real time constraints.

MABS could involve a great number of agents and each of them should perceive their environment. So the conception of an environment model allowing a fast visual perception for each agent of the simulation is the main contribution of this paper. This model is inspired by the synthetic vision models and the spatial data-structures used in 3d rendering. Semantical and symbolical informations are introduced into our model to allow high-level behaviors for the virtual agents. In another point of view, our environment architecture could be considered as an interface between a multiagent system and an immersive virtual reality environment.

This paper is organized as follow. The next section reviews several background research works. The section 3 introduces our environment model. An implementation on a simple crowd simulation example is also presented. It is followed by the experimental results which are discussed. Finally our future works in that field to achieve further optimizations are exposed.

2. Related works

The problematic of visual perception and pedestrian simulation was already focused by several works, and especially in motion synthesis and real-time computer animation. This section reviews the virtual environment definitions and the synthetic vision methods. The last part is dedicated to an introduction of the multiagent's simulation concepts.

2.1. *Virtual Environment*

To evolve in a virtual world and exhibit a complex behavior, an agent requires several semantical, symbolical and topological data about its environment. KALLMANN and THALMANN introduced smart objects in which all interesting features concerning the objects themself and how-to-interact with them are stored [KAL 98]. FARENC et al. developed a model of Informed Environment dedicated to urban life simulation [FAR 99]. Their environment is decomposed as a set of environmental entities called ENV. An ENV represents a surface or a volume of the scene, and consists of semantical information, a list of objects located inside the area and a list of associated actions or behaviors. A 3d-scene is successively decomposed into sub-areas to create an inclusion tree. Then the visual perception is simulated to extracting information from a database: each agent of a given ENV is aware of the list of objects being there. Donikian have developed a virtual environment modeling system (VUEMS) dedicated to driving simulation [DON 97]. Thomas adapts it for pedestrian's simulation in urban environment [THO 99]. He use a similar approach to Farenc et al.'s model. This multilevel database integrates three different level of environment : geometrical, topological and semantical. The perception in this type of environment is also simulated by a direct extraction from the database.

The following section will focus on the synthetic vision methods and on how it could be adapted to virtual reality constraints.

2.2. *Synthetic Vision*

The principles of the synthetic vision is to simulate the biological perceptive organs. All the proposed methods have a common aim: allowing a visual perception for autonomous entities inside a virtual environment. They are mostly inspired from the 3d-rendering techniques. Renault et al. introduced a $2\frac{3}{4}$ vision system [REN 90]: all objects are projected on a 2d bitmap according to the viewer's point of view, the distance from the eye to all the points of the objects are extracted from the graphical card's Z-Buffer [1] ($\frac{1}{2}$d), the objects' identifiers are stored inside the back-buffer of the graphical card ($\frac{1}{4}$d). Tu and Terzopoulos implemented a synthetic vision system for their artificial fishes using ray-casting [TU 94]. Terzopoulos and Rabbie developed a binocular vision system in which each eye was implemented as a four coaxial camera [TER 95]. The object's recognition is assured with a database of objects' models and an algorithm of pattern matching inspired from the Swain method [SWA 91] based on color histograms. Noser et al. has introduced a synthetic vision model where actors perceive there environments from small false-coloring images rendered from a point of view by the computer hardware [NOS 95a]. Kuffner and Latombe used a similar approach but with a different representation for the visual memory [KUF 99]. Instead of using an octree [2], they preferred a list of object identifiers. They so avoid a redundancy of geometrical informations between the environment and the visual memory. Noser and Thalmann admitted that their vision system could not be done in real time without some adaptations [NOS 95b]. They proposed a virtual world representation for which the synthetic vision was reduced to a simple visibil-

1. also called depth-buffer
2. spatial data structure which divides the space in 8 voxels

ity test. Then they extract position and semantical information of the visible objects directly from the 3d environment. WEN et al. presented an adaptation of [REN 90] in which an octree was used to assure the hierarchical scene decomposition [WEN 02]. The synthetic vision was reduced to a simple intersection test between the agent's view frustum and the object's AABB [3]. They used then a local area Z-buffering to determine the exact object visibility. The synthetic vision methods presented in this section are *incompatible* with the simulation involving a great number of agents: they spend too much computational time [MUS 97]. So we will try to avoid this incompatibility using a synthetic vision method associated to an adapted environment conception.

2.3. *Multi-Agent System for Virtual Human Simulation*

Multiagent-based simulation differs from other kinds of computer-based simulation in that the simulated entities are designed and implemented in terms of agents. Our objective is to be able to immerse a MAS in a virtual world especially in a virtual urban environment. From this point the notion of perception – interface between an agent and its environment – appears as a key problem. We adopt a specific agent's definition centered on its interfaces. An agent is a physical or virtual entity which perceives its environment (but inside a limited area) via its sensors and acts via its effectors [4]. It verifies the following properties :

– autonomy : act without human intervention and possesses resources of its own;

– communicative : communicate directly with other agents;

– responsiveness : answer to outside events;

– behavior : own one or more objectives whose behaviors tend to satisfy them;

– situation : perceive partially the environment via sensors, eventually builds a partial representation (i.e memory), and change its configuration by acting above locally via effectors.

In this paper, we focus only on micro-simulation, in which agents correspond to individual entities. It could be contrasted to macro-simulation mostly inspired by mathematical models where a population was considered as a whole. In micro-simulation, each entity is an agent and the organizations emerge from the interactions between the individuals. Due to their intrinsic properties, the agents are naturally adapted to model the human behaviors. The case study led at section 5 is oriented to pedestrian's and crowd simulation but it is not the main subject of this paper. Many simulation platforms, which tends to sim-

3. Axis Aligned Bounding-Boxes
4. Definition inspired from [BOU 01] and [FER 95]

ulate pedestrian traffic and behaviors with MAS or cellular automata, already exists [JIA 99, KAR 02, DIJ 00]. Traditionally the pedestrian traffic simulation is done in macroscopic level, considering the aggregation of the pedestrian movements into a flow. At microscopic level each pedestrian is considered as an individual situated [5] agent in interaction with its neighbors.

3. Environment Model

This section will describe our environment model. We will detail each of different levels of abstraction, making thus the interface between the 3d world and the multiagent platform.

3.1. *Global description*

The proposed model is similar to [WEN 02] but was adapted to a 2d-case with the projection of the object's bounding-boxes. The environmental entities have been classified into two categories:

– immovable (or static): building, road, environment's agent...

– mobile (or dynamic): pedestrians, vehicles...

The major aim of our environment model is to assure the visual perception of a great number of autonomous agents [6] evolving in a Virtual Environment. A set of components is proposed to supply this target. As illustrated by the figure 1, we propose a component called "metric environment" which acts as a layer between a multiagent platform (i.e. MadKit) and a Virtual Environment (i.e. Virtools). In this paper we don't present the integrality of our environment model, only the low-level which assure a fast visual perception is described.

3.2. *Environment Levels*

Our works define the VE for a multiagent system as an architecture composed of several layers. Each of them represents an abstraction level.

– *3d database*: this level is considered as a database of all the objects into the scene. In most of the case, this database is only used by the rendering engine during the simulation.

– *Metric Environment*: this is the lowest level into the MAS and the highest level into the 3d. Indeed it structures the 3d information into dedicated spatial data trees which could be easily used by the agents, or called by the higher

5. situated in space and sometimes in time too [FRA 00]
6. the quantity depends of the complexity of the agents

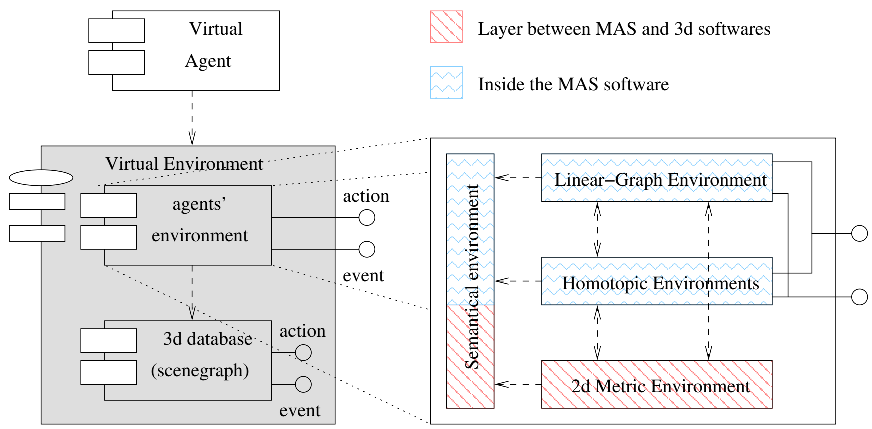

Figure 1: Virtual Environment Components

environment layers. The metric environment could be mostly generated prior to the simulation starting.

– *Linear-Graph Environment*: this is an example of high level environment. It defines the space as a set of edges and nodes. Each of them are associated to an surface (i.e. : an edge could be associated with a road, a node with a crossroad).

– *Homotopic Environments*: there environments act as layers between a lower and an higher environments.

– *Semantical Environment*: this is a transversal level in which all the object's semantics are specified. It could be based on a simple object-oriented typing or on an urban ontology. Each object from the other environments point to one or more concepts from this level.

4. Metric Environment and Visual Perception

The base of a classical synthetic vision is the rendering of a small false-coloring image (128×128) for each agent's point view. But it remains relatively expensive if the number of agents increases and it becomes difficult to satisfy the required real-time constraints. On the other hand, if the synthetic vision was reduced to a simple visibility test [WEN 02], it is unnecessary to use the completeness of the visibility pipeline: only the frustum and occlusion culling[7] are still useful. Moreover the agents which exhibit high level behaviors do not need a lot of geometrical informations: only the bounding boxes of the

7. selection of the not-completely occluded objects located inside the frustum

3d objects are really required. This section described how the perception is implemented face to face structure of *Metric Environment*.

4.1. *Scene Decomposition*

For small environment we have assimilated the building to static entities and we store them in a modified BSP [8] integrating an Icoseptree heuristic. For large environment a hierarchical scene decomposition similar to Farenc's environment model is used [FAR 99] but with a focus especially on the visibility problem. Thus the spatial decomposition is mainly influenced by visibility constraints. The scene decomposition is based on the Cell and Portal algorithms [LUE 95, TEL 92]. Each cell has its adjacency graph (similar to the Farenc's connexity graph) and two instances of the spatial data structures explained above.

4.2. *Spatial Data Structures*

Each environmental entities is stored inside an adapted spatial data structure inspired from the dynamic AABB tree presented by Shagam [SHA 03]. An dynamic AABB is a tree in which each node consists of an AABB, a list of children and a list of objects. The AABB of each node encloses all of the node's objects and all of the child AABB. The tree is built by successively splitting the top-level in half and putting the objects into the child in which it falls. Shagam tested four nesting heuristics which precise how nodes are split. Icoseptree is the one that we chose. The icoseptree heuristic is a blend of K-D tree and octree behavior. Each axis is divided into three sub-regions, two of which are for objects entirely on one of the sides, and the third for objects whose the bounding spaces intersect the splitting plan. The three regions on the two dimensions combine to provide a quadtree equivalent with eight regions. Each one is associated with a child node for objects to be inserted into.

4.3. *Perception Method*

The vision process is reduced to a simple 2d frustum and occlusion culling. The frustum surface is successively tested against the AABB of each node to determine fully and partially visible objects. Then we use a simple occlusion culling algorithm (Z-buffer equivalent) to assure the exact object visibility. The semantic of each object is extracted from the semantical environment and returned to the requiring agent.

---

8. Binary Space Partition Tree

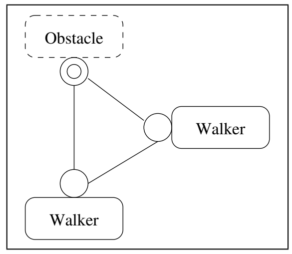

Figure 2: Pedestrian organization

5. Crowd Application

Our environment model and the corresponding perception component are typically adapted to the crowds' movement problems. The purpose of this study case is to run a simulation with many autonomous agents inside a constrained spatial environment to test and validate the efficiency of our approach.

An organizational framework was used to describe multiagent organization [HIL 02]. The *pedestrian organization* is composed of two roles: *obstacle* and *walker* (see figure 2, page 8). The environmental role *obstacle* interacts only with *walker* to avoid collision with them. The role *walker* aims to reach a point inside the spatial environment and tries to avoid any collision with the other components of the system. Each walker is attracted by a spatial location (called target) and tries to reach it using a potential field heuristic [REY 87]. This technique was chosen because it is rather simple to be quickly implemented and not strongly influence the measures on our perception component.

This model is designed as a simple case study for our perceptive approach. Other works focus particularly on the movements of single people or robots in constrained environments [SIM 00, LUC 02, CHA 02], or the simulation of a crowd (i.e. [MUS 97]).

5.1. *Walker's behavioral model*

The *walker*'s behavioral model is defined using statecharts (see figure 3, page 9) . Due to the size of this paper, only the major point of this model is presented. He/she has two parallel behaviors: the perception of his/her nearest environment, and the computation and the applications of his/her movements. The walkers' perception is processed only at the starting of the system or when a walker has changed his/her location. The perception frustum is the half-circle at the front of the walker. The second behavior is a cyclic sequence of three states. The first permits to compute the best movement direction accordingly to a simple potential-field theory. The second moves the walker by one step along

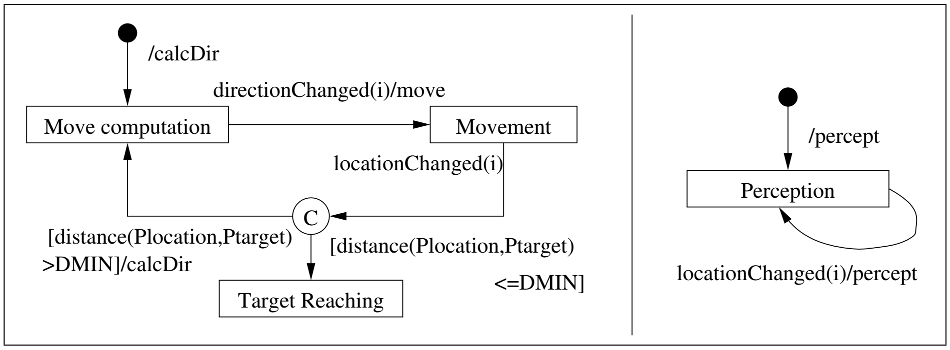

Figure 3: Walker's behavioral model

the directional vector. If the target point was reached ($distance \leq DMIN$), the walker stops. If not, he/she enters into another cycle.

5.2. *Implementation and Results*

The environment model have been implemented with Java 1.4.2 using Java3D 1.3.1 and tested on a PC under Linux 2.6.3 with an Intel Pentium IV processor (2.4 GHz), 512 Mo RAM and graphics accelerator using a Geforce 3 TI 200 chipset. The MAS has been developed with MadKit 3.1b5 and the pedestrian implementation is based on the *AbstractAgent* class. The table 5.2 presents the computational time of the perception process for different simulations from 50 to 50.000 agents. The figure 4 presents the evolution of this time of perception according to the number of agents.

5.3. *Discussion*

The spatial data structures used to implement our environment model (BSP, Quadtree...) were originally designed for 3d-rendering. This type of structures could store a great number of objects ($n > 10^5$) [SHA 03]. In our case the used structures are relatively oversized with regard to the number of agent implying in the simulation. We simulate between 50 and 50000 agents, so we use these structures in most of 50 percent of their real potential. Moreover the figure 4 proves it because the increase of the number of agents influences only few on the time of agent's perception. Very few multiagent platform support the simulations implying a great number of agents exhibiting high level behaviors. A pedestrian could not be reduced to a simple ant or termite. Our model assures a fast visual perception ($\mu < 1ms$) and confirms that we could developed real-time simulation in VE using a synthetic vision process. We use icoseptree as

#agents	Average of visible objects per frustum	Perception time (ms)		Mean (ms)
		Min	Max	
50	6.41	0.69	1.24	0.97
100	8.08	0.53	1.18	0.86
200	9.74	0.51	1.13	0.82
400	16.47	0.55	1.09	0.82
600	17.17	0.65	1.17	0.91
800	16.21	0.47	1.04	0.76
1000	17.37	0.6	1.12	0.86
5000	20.53	0.59	1.08	0.84
20000	21.67	0.59	1.06	0.83
50000	21.87	0.64	1.1	0.87
Mean	**15.55**	**0.58**	**1.12**	μ =**0.85**
Std deviation		**0.07**	**0.06**	**0.06**

Table 1: The computational time of the perception process

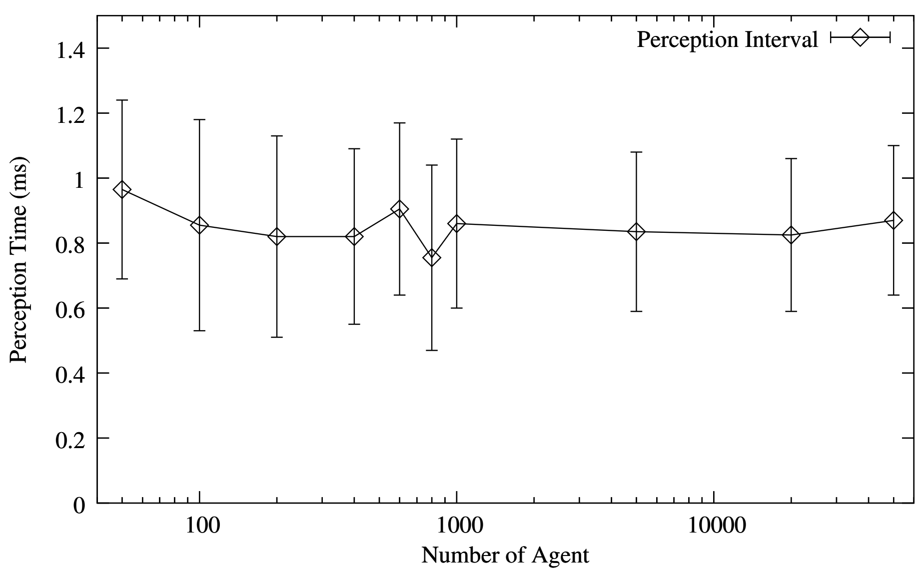

Figure 4: Evolution of the time of perception according to the number of agents.

nesting heuristic for our spatial data structures because it provide a good compromise between the K-D tree and the octree behaviors. It provides also good performances for practically every type of topological environments, and scales logarithmically with the input size. Further optimizations could be performed in studying the optimal split threshold.

6. Conclusion and Future Works

This paper briefly introduces our works on immersion of a multiagent system inside a virtual environment. The first stage to reach is to assure an effective perception for all the agents involved in the simulation. To achieve this goal we propose a vision-based environment model and the corresponding fast visual perception algorithms. This paper illustrates also that the perception constitutes a key concept for the immersion of a MAS inside a VE. From our proposal, we plan to develop pedestrian's and vehicle's simulations based on groups of agents defined from common goals and on the agents' mutual perceptions. Future works will be also carrying out on modeling environment dedicated to multiagent-based simulation in virtual environments and integrating different levels of simulation (micro-, meso-, macro-simulation). By this way we will simulate large scale virtual environment with a great number of agents.

7. References

[BOU 01] Bouzid M., "Contribution à la modélisation de l'interaction agent-environnement : Modélisation stochastique et simulation parallèle", Doctorat, University Henri Poincarré, Nancy I, 2001.

[CHA 02] Chapelle J., Simonin O., Ferber J., "How Situated Agents can Learn to Cooperate by Monitoring their Neighbors' Satisfaction", *15th European Conference on Artificial Intelligence (ECAI'02)*, Lyon, Jul. 2002, p. 68-72.

[DIJ 00] Dijkstra J. T. H., A.J. J., "A Multi-Agent Cellular Automata System for Visualising Simulated Pedestrian Activity", *Proceedings on the 4th International Conference on Cellular Automata for research and Industry*, Eindhoven, Netherlands, 2000, p. 29-36.

[DON 97] Donikian S., "VUEMS: a Virtual Urban Environment Modeling System", *Computer Graphics International?97*, Hasselt-Diepenbeek, Belgique, June 1997, IEEE Computer Society Press.

[FAR 99] Farenc N., Boulic R., Thalmann D., "An informed environment dedicated to the simulation of virtual humans in urban context", *Eurographics'99*, vol. 18, Milan, September 1999.

[FER 95] Ferber J., *Les systèmes multi-agents: Vers une intelligence collective*, InterEditions, Paris, 1995.

[FRA 00] Franck F., "Simulating the Collision Avoidance Behavior of Pedestrians", 2000.

[HIL 02] Hilaire V., Koukam A., Gruer P., "A Mechanism for Dynamic Role Playing", *Agent Technologies, Infrastructures, Tools and Applications for E-services*, num. 2592 Lecture Notes in Artificial Intelligence, Springer Verlag, 2002.

[JIA 99] Jiang B., "SimPed: Simulating Pedestrian Flows in a Virtual Urban Environment", *Journal of Geographic Information and Decision Analysis*, vol. 3, num. 1, 1999, p. 21-30.

[KAL 98] Kallmann M., Thalmann D., "Modeling objects for interaction tasks", *Eurographics Workshop on Aniamtion and Simulation*, Lisbon, Portugal, September 1998.

[KAR 02] Kardi T., "Microscopic Pedestrian Flow Characteristics : Development of an image Processing Data Collection and Simulation Model", PhD thesis, Tohoku University, Japan, 2002.

[KUF 99] Kuffner J., Latombe J., "Fast Synthetic Vision, Memory, and Learning Models for Virtual Humans", *IEEE International Conference on Computer Animation*, Geneva, Switzerland, May 1999, p. 26-29.

[LUC 02] Lucidarme P., Simonin O., Liégeois A., "Implementation and Evaluation of a Satisfaction/Altruism Based Architecture for Multi-Robot Systems", *IEEE Int. Conf. on Robotics and Automation (ICRA'02)*, Washington D.C., May 2002, p. 1007-1012.

[LUE 95] Luebke D. P., Georges C., "Portals and Mirrors : Simple, Fast Evaluation of Potentially Visible Sets", *Proceedings of the 1995 Symposium on Interactive 3-D Graphics, ACM Press, NY*, , 1995, http://www.cs.virginia.edu/ luebke/.

[MUS 97] Musse S., Thalmann D., "A model of human crowd behavior : group inter-relationship and collision detection analysis", *Computer Animation and Simulation'97*, Wien, 1997, Proc. Eurographics workshop, p. 39-51.

[NOS 95a] Noser H., Renault O., Thalmann D., Magnenat-Thalmann N., "Navigation for digital actors based on synthetic vision, memory and learning", *Computer and Graphics*, vol. 19, num. 1, 1995, p. 7-19, Pergammon Press.

[NOS 95b] Noser H., Thalmann D., "Synthetic vision and audition for digital actors", 1995, Hansrudi Noser and Daniel Thalmann, Synthetic vision and audition for digital actors.

[REN 90] Renault O., Magnenat-Thalmann N., Thalmann D., "A Vision-based Approach to Behavioural Animation", *Journal of Visualization and Computer Animation*, vol. 1, num. 1, 1990, p. 18-21.

[REY 87] Reynolds C., "Flocks, Herd and Schools : a Distributed Behavioral Model", *Computer Graphics*, vol. 21, num. 4, 1987, p. 15-24.

[SHA 03] Shagam J., "Dynamic spatial partitioning for real-time visibility determination", Informatique, New Mexico State University, Department of computer science, April 2003.

[SIM 00] Simonin O., Ferber J., "Modeling Self Satisfaction and Altruism to handle Action Selection and Reactive Cooperation", *The Sixth International Conference on the Simulation of Adaptative Behavior*, vol. 2, 2000, p. 314-323.

[SWA 91] Swain M., Ballard D., "Color indexing", *Journal of Computer Vision*, vol. 7, 1991, p. 11-32.

[TEL 92] TELLER S. J., "Visibility computations in densely occluded polyhedral environments", Informatique, University of California at Berkeley, 1992.

[TER 95] TERZOPOULOS D., RABBIE T., "Animat Vision : Active Vision in Artificial Animals", *Fifth international conference on Computer Vision (ICCV'95)*, June 1995, p. 801-808.

[THO 99] THOMAS G., "Environnements virtuels urbains : modélisation des informations nécessaires à la simulation d'humanoïdes", Informatique, Université Rennes I, IRISA, SIAMES, December 1999.

[TU 94] TU X., TERZOPOULOS D., "Artificial Fishes: Physics, Locomotion, Perception, Behavior", *Proc. of SIGGRAPH'94, Computer Graphics Proceedings*, vol. 28, July 1994, p. 43-50.

[WEN 02] WEN Z., MEHDI Q., GOUGH N., "A New Animation Approach for visualizing Intelligent Agent Behaviours in Virtual Environment", *Proc. of the Sixth International Conference on Information Visualiation (IV'02)*, July 2002, p. 93-98.