EUROGRAPHICS 2000 / M. Gross and F.R.A. Hopgood
(Guest Editors)

Volume 19 (2000), Number 3

Modelling virtual cities dedicated to behavioural animation

Gwenola Thomas † and Stéphane Donikian ‡

[gthomas, donikian]@irisa.fr
IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France

Abstract

In order to populate virtual cities, it is necessary to specify the behaviour of dynamic entities such as pedestrians or car drivers. Since a complete mental model based on vision and image processing cannot be constructed in real time using purely geometrical information, higher levels of information are needed in a model of the virtual environment. For example, the autonomous actors of a virtual world would exploit the knowledge of the environment topology to navigate through it. In this article, we present a model of virtual urban environments using structures and information suitable for behavioural animations. Thanks to this knowledge, autonomous virtual actors can behave like pedestrians or car drivers in a complex city environment. A city modeler has been designed, using this model of urban environment, and enables complex urban environments for behavioural animation to be automatically produced.

1. Introduction

Existing 3D Modelling Systems makes it possible to generate realistic models of virtual environments, including texture mapping and level of detail [15], while rapid progress in graphic cards allows extensive use of textures. Modelling a city with such tools is still a long, complex and costly task. A lot of work has been done to partially automate the rebuilding process. From photogrammetry, 2D Geographical Information Systems (GIS), and a Digital Elevation Model (DEM), it is possible to build the model of an object such as a building. The object's lower boundary is defined using 2D GIS, and the lower boundary's elevation is interpolated by a DEM. Skeletal points of the roof are measured by photogrammetry. Then the roof surface can be modeled by triangular network with the skeletal points. In order to enhance the visual reality, each object surface can be textured. For example, the CyberCity Modeler software can be used to automatically generate the topology of buildings from photogrammetrically measured points [14]. Recently, IWI company has developped a technique which fully automates the rebuilding process of a city (at the time of writing Rennes and part of Paris) from different input data such as GIS data, a Digital Terrain Model, a very detailed aerial orthophoto and a set of generic textures and objects. As they use generic textures, specific well known buildings still need to be built manually. For an enhanced realism of specific buildings it is possible to combine techniques such as videogrammetry and laser range scanner [25] or real-world video and digital maps [17].

All these techniques are very useful for the visual realism of virtual urban cities but they are not sufficient due to the lack of animation in the virtual environments. Being uninhabited, these virtual city model does not provide the visitor with real life feeling. In order to animate these scenes, we have to specify the behaviour of dynamic entities such as pedestrians, car drivers or public transportation systems. Behavioural animation consists of a high level closed control loop[10, 19, 5, 4], which enables autonomous entities to be simulated [1, 24]. Such actors are able to perceive their environment, to communicate with others[3] and to execute a number of actions, such as walking in the street or grasping an object, in accordance with the nature of the environment and with their intentions. If more complex behaviour than obstacle avoidance is to be reproduced, it is necessary to provide additional data such as mereotopological and semantic information (Mereology concerns the part-whole relationships, while topology concerns connection relationships). The pur-

---

† University of RENNES1
‡ CNRS

pose of this paper is to propose a city model *affordant* to autonomous actors. The term *affordance* which has been introduced by J.J. Gibson in [11] will be explained in the core of the article.

In this paper, we focus on three different part: first the nature of data required to simulate behavioural entities from a psychological point of view, then the structure of our city model and finally the city modeler that has been developped. Car driver and pedestrian behavioural models which are specified by using other specific tools [20] are not addressed in this paper.

2. Behavioural Animation Requirements

2.1. Introduction

Information needed to describe the behaviour of an entity, depends on the nature of this entity. No theory exists for determining either the necessary or sufficient structures needed to support particular capabilities and certainly not to support general intelligence. As direction and inspiration towards the development of such a theory, Newell [22] posits that one way to approach sufficiency is by modelling human cognition in computational layers or bands. He suggests that these computational layers emerge from the natural hierarchy of information processing. Lord [18] introduces several paradigms about the way the brain works and controls the remainder of the human body. He explains that human behaviour is naturally hierarchical, that cognitive functions of the brain are run in parallel. Moreover cognitive functions are different in nature: some are purely reactive, while others require more time. Newell asserts that these levels are linked by transferring information across hierarchical levels, and that each of them operate without having any detailed knowledge of the inner workings of processes at other levels. All that is required is a transfer function to transform the information produced by one level into a form that can be used by another. Particularly important is the notion that symbolic activities occur, locally based on problem spaces constructed on a moment-to-moment basis.

Goffman [12] describes the techniques that pedestrians employ in order to avoid bumping into one another and call them traffic codes. The first technique is called *externalization* which is a term to indicate that people are constantly making other pedestrians aware of intentions by over-all body gestures. Each pedestrian must make his/her direction and rate apparent to others occupying the interaction area. The second technique called *scanning* is a process by which pedestrians selectively gather *externalized* information from other pedestrians. During this process, an important aspect concerns the mereotopological structure of the interaction area, as the same externalization of an entity will not have the same meaning in a pedestrian crossing as on a sidewalk. During the scanning process, it is necessary to take into account both the externalization of other pedestrians and the structure of the surrounding environment.

Lane following is a preponderant activity both for pedestrians and car drivers. In this activity, the main input concerns the shape of the lane which means its skeleton or axial line and its width. Semantic information is also of interest. For example, a specific coding on a sidewalk will delimitate the part devoted to cyclists from the pedestrian area. Another important aspect is the tactical one: what is the best path to go from an initial location to a final one, depending on the nature of the motion and on characteristics of the environment ? This can be performed by the use of topological information on the scene which is different for car drivers and pedestrians as their routes are not constrained by the same rules.

2.2. Visual Perception

The simplest behaviour, for a pedestrian walking in a street, consists in minimizing possible interactions which means avoiding static and dynamic obstacles. But, even in this simple walking activity, the pedestrian needs to know the nature of the objects he/she will interact with. For example, a public phone is considered as an obstacle to avoid for most people, but some of them will be interested in its function and will use it. When crossing a street, one activity consists in reading the signals, which means that it is necessary to associate semantic information to geometric objects in the scene, and to update it during the simulation. In the realm of behavioural psychology, there have been a few studies on visual perception, mainly based on Gibson's theory of affordances[11]. The theory of affordances is based on what an object of the environment *affords* to an animal. Gibson claims that the direct perception of these affordances is possible. Affordances are relations between space, time and action, that work for the organism. What the world is to the organism depends on what the organism is doing and might do next. An affordance is an invariance in the sense that it is a recurrent, generalized relation between the environment and the agent's activity. For computer scientists, the Gibson's theory is really attractive because it will assign a semantic behavioural value to each object, i.e. what the human being is likely to do with a given object. M. Kallmann [16] introduces smart objects, in which all interesting features of the objects are defined during the modelling phase. He defines four classes of different interaction features: intrinsic object properties, interaction information, object behaviours and expected agent behaviours.

2.3. Conclusion

In accordance with Gibson's ecological theory, components of the virtual urban environment should be informed. N. Farenc[9] has suggested using an informed environment, dedicated to urban life simulation, which is based on a hierarchical breakdown of an urban scene into environmental entities providing geometrical information as well as semantic notions. The geometrical level is sufficient for a driving simulation in which vehicles are all driven by users in the loop. Since vehicles driven by autonomous entities are added in the simulation, other kinds of information become necessary[23, 2]. Information required to simulate the behaviour of a car driver is of different kinds: the road network, with its geometry (road shape), its rules (horizontal and vertical road-signs) and its symbolic information associated to geometrical objects (lane characteristics, inter-lane marking, names of streets, buildings, parks, squares or neighborhood). As far as we know there is no normalization of the design of elements like a round-about or a crossroads. Each element of the thoroughfares in a urban environment is unique, but it is possible to classify them in a small number of categories[6]. Concerning pedestrians, it is not possible to restrict the environment to a subcomponent of the city, like the thoroughfare for vehicles, as they are not restricted to the thoroughfare but, unlike the others (car or tram drivers for example), they can wander about everywhere in the city. It is necessary to build a complete mereotopological representation of the city, which will be described in the following section.

3. The City model

Our aim is to be able to offer virtual humans all the information required to perform realistic actions according to their intention. Due to the complexity of the urban structure we have to deal with both the efficiency of the access to information and the quantity of data to store. The use of a mereotopological structure makes it possible to have an efficient access to information. At the present time, we are interested in reproducing life in the streets and not in what is happening inside a building or inside a subway station. Interesting information for a building is its external shape, the location of its entrance, its functionality (administration, shop, private house, ...) and the value of input and output pedestrian flows depending on the time of day. Input and output flows can also be managed for subway entrances.

Two data structures are provided for the behavioural entities:

1. *Hierarchical structure*: the path planning behaviour of entities works at distinct levels of the city hierarchy. It is necessary to provide the dynamic entity with a hierarchical structure (Figure 1 shows the whole structure breakdown). The construction of the tree is based on inclusion relationships between polygons. These polygons are the ground projection of the heterogeneous or homogeneous zones composing the city. The root of the tree is the town (heterogeneous), the leaves are geometrical regions of homogeneous nature. At the intermediate levels of the tree are different kinds of heterogeneous spaces which can be constructed automatically or manually (for example the user of VUEMS draws a bounding area to group several buildings, parks and roads in order to build a block). In Figure 1, free regions are positioned across the border between heterogeneous and homogeneous spaces because they can belong to both types of space. For example a square is *heterogeneous* when it contains roads, walking areas and buildings.
2. *Topological structure*: To navigate from one street to another crossing an intersection, a simulated driver uses a graph representing the road network. This graph is a strongly connected planar graph. Nodes of the graph are the homogeneous spaces found at the lowest level of the town hierarchy (lanes, intersection regions). Edges link adjacent spaces. Those edges contain geometrical and symbolic information; this information is used by behavioural entities for travelling through the town. Because pedestrians can go anywhere the topological structure does not only contain the road network but also buildings, parks and places which are connected to it.

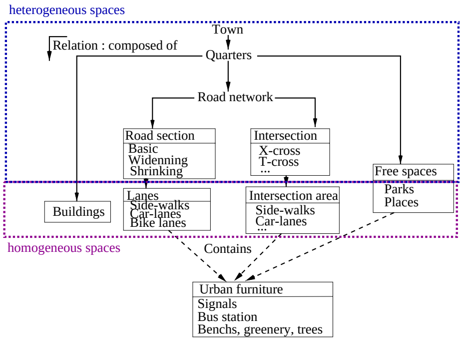


Figure 1: *the hierarchical structure of the town.*

The components of those structures are different kinds of regions. The behaviour of the dynamic entities is strongly linked to the nature of those regions which are now described. The way dynamic entities circulate depends on the areas they pass through (to wander in a park, to follow a side-walk, to cross a street on a zebra-crossing...). In those spaces, entities must also avoid static and dynamic obstacles. Three types of regions are distinguished:

1. *constrained regions*: in these areas there is a preferred direction of circulation (such as lane, corridor or sidewalk). For example, cars *follow* car-lanes; the geometrical and symbolic information (direct or indirect lane) of car-lanes force the driver to follow an certain orientation and a direction of driving;
2. *intersection regions*: where entities can change routes;
3. *free regions*: there are no constraints on circulation in those areas: the entity can freely wander (typically a square).

Each of those three regions has a specific geometry which implies specific algorithms of locomotion for dynamic entities. The geometry of a region is defined by its outline. This outline is composed of borders. These borders can be crossable or not (walls, barriers, etc.). We now describe the three types of regions.

3.1. Lanes

A lane is defined by four oriented borders (see stamp 4 in Figure 2) geometrically indexed on the skeleton of the road (oriented from construction):

1. two longitudinal borders: B_RIGHT and B_LEFT
2. two transversal borders: B_IN and B_OUT

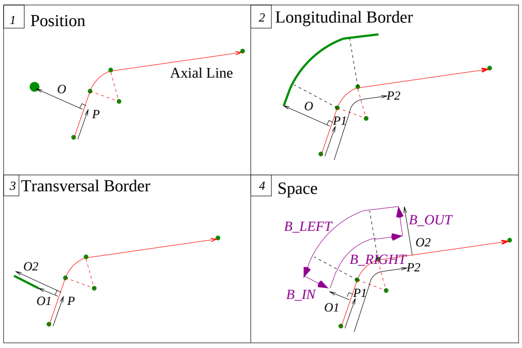


Figure 2: *From position to space.*

There are two ways of passing through a lane: to cross it or to follow it. Those actions consist in going from one border to another. In the four borders configuration of the lane, three actions express the relation between one border and the others: A_LEFT, A_RIGHT and A_STRAIGHT. Given an action to perform through the area and an entrance border, the trajectory is computed by finding the target border thanks to the routing matrix (see Figure 3).

	A_RIGHT	A_STRAIGHT	A_LEFT
B_IN	*B_RIGHT*	*B_OUT*	*B_LEFT*
B_RIGHT	*B_OUT*	*B_LEFT*	*B_RIGHT*
B_OUT	*B_LEFT*	*B_IN*	*B_RIGHT*
B_LEFT	*B_IN*	*B_RIGHT*	*B_OUT*

Figure 3: *Routing matrix.*

The geometry of the borders is relative to the skeleton of the road. Figure 2 shows how relative positioning is performed relatively to the skeleton. Lanes must be of constant width and continuous. To enforce these two constraints the skeleton is a list of segments and arcs. A relative position (first stamp in Figure 2) is defined by two parameters:

- **O**: the offset to the skeleton;
- **P**: parameters on the skeleton; there are three parameters: primitive number, type of primitive (arc or segment) and parametric coordinate (between 0 and 1) on this primitive.

Given an entity having a relative position (**O**, **P**); the lane configuration (see stamp 4 Figure 2) easily allows :

- to test inclusion: the entity is included in the space if its parameter is between $\mathbf{P}_1$ and $\mathbf{P}_2$, and its offset between $\mathbf{O}_1$ and $\mathbf{O}_2$.
- to follow a lane: the orientation is tangent to the skeleton at parameter **P**; offset is constant.
- to avoid obstacles: a list of objects is associated to lanes (see in Figure 1 the containing relationship at the bottom of the town hierarchy). Objects are positioned relatively to the skeleton; this allows the performance of efficient obstacle avoidance. Figure 4 illustrates obstacle avoidance for an entity.

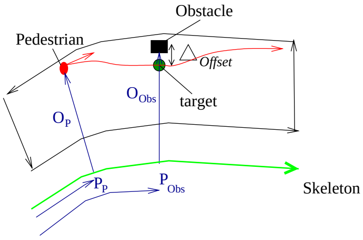


Figure 4: *Obstacle avoidance*

When an obstacle $(\mathbf{O}_{Obs}, \mathbf{P}_{Obs})$ is in the path of the entity $(\mathbf{O}_E, \mathbf{P}_E)$, a target position $(\mathbf{O}_{Obs} + \Delta_{Offset}, \mathbf{P}_{Obs})$ is computed to bypass the obstacle.

In reality, the situation is more complex than simply avoiding an obstacle. Studies of the dynamic interactions involved in the behaviour of entities moving through lanes have been carried out, but this paper does not intend to deal with these aspects. This is illustrated by an example including thirty pedestrians on a sidewalk in section 5. The work consists in specifying social rules of interaction in pedestrian behaviour. Those social rules depend on the regions crossed (side-walk, car-lane) and on the nature of the entities interacting (pedestrian/pedestrian, pedestrian/driver).

3.2. Intersections

Different configurations of roads intersections exist: X-cross, Y-cross, T-cross (cf [6] for a complete description of the road network). In Figure 5, two roads containing six lanes intersect. The result is thirty-six interconnected intersection areas (see the partitioning in stamp 2). The area type is defined by the combination (based on priority rules) of the kind of lanes intersecting; for example, when a car-lane (bitumen) intersects a cycle-lane (green-paint) the appearance of the intersection region is bitumen (see stamp 1 in Figure 5). As for lanes, intersection regions are defined by four oriented borders (see drawing at the bottom of Figure 5). The four borders are longitudinal (see Figure 2). **B**_IN and **B**_OUT are indexed on **S**keleton2, whereas **B**_RIGHT and **B**_LEFT are indexed on **S**keleton1. This structural similitude with lanes implies the same use of algorithms (routing matrix, inclusion tests...) for travel through those areas. At intersections, dynamic entities can change route. The path followed when changing route is a Bezier curve (see trajectory from **B**_IN to **B**_RIGHT in Figure 5).

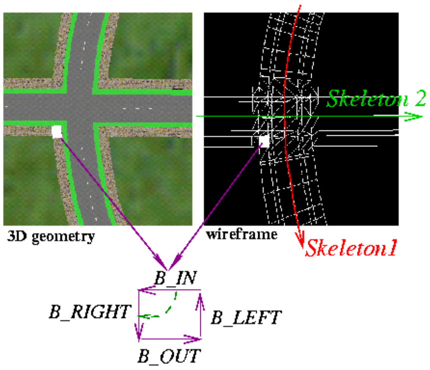
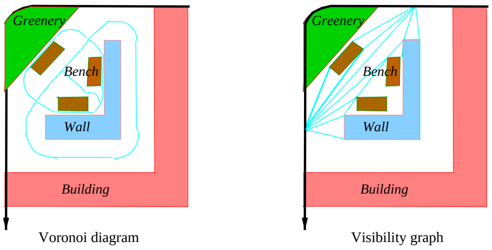


Figure 5: *X-Cross representation.*

3.3. Free regions

Free regions are 2.5D polygonal spaces. They are structured by objects characterized by their type (bench, lake...) and function (to sitting, to swimming...). The main idea is to associate a network to those regions which will help pedestrian circulation. The skeleton of those regions is computed in VUEMS (our city modeler) using computational geometry algorithms. We have chosen two breakdown models: the Voronoï diagram extended to a set of polygons[13] and the visibility graph (see example of a park in Figure 6).



Voronoi diagram

Visibility graph

Figure 6: *Free regions breakdown.*

A skeleton (similar to the road skeleton) is then available for the pedestrian; he/she walks on it or near it (offset distance). The extended Voronoï diagram respects optimal distance to objects. The kernel of the Voronoï cells is made up of objects which are characterized by their type and function. The Voronoï paths (list of segments and parabolas) give geometric sense to the action *walk around the statue*. The visibility graph provides the shortest path for the pedestrian.

3.4. Topology

Nodes of the topological structure are regions defined by their type (side-walk, car-lane..) and their function (lane, intersection area, free region). Edges (i.e. the borders) delimitate the geometry of those regions. Figure 7 illustrates the graph significance: the edge between two spaces $\mathbf{E}_1$ and $\mathbf{E}_2$ is a pair of borders (Border1, Border2).

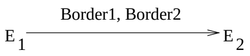


Figure 7: *Edge definition.*

Because they are adjacent, those borders link the spaces $\mathbf{E}_1$ and $\mathbf{E}_2$. One border can be split into several parts of different types. For example, guard-rails can be placed on a sidewalk to prevent pedestrians from crossing the street. Parts of borders are of three types:

- impassable: fence, hedge, wall, ...
- manoeuvrable: door, portal, ...
- passable: connexity of two areas without any obstacle.

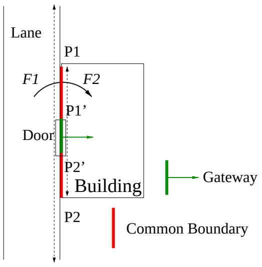


Figure 8: *Connexity and accessibility relations.*

Two areas are connected in the graph of accessibility if one or more part of their common border is manoeuvrable or passable. In figure 8, the lane and building objects are connected, and one gateway (the door of the building) specifies the area of accessibility between the two regions. Figure 9 illustrates the difference between the connexity and the accessibility graph on the example of a crossroads composed by four buildings (B1,B2,B3,B4), two parks (P1,P2) and the road network. For example, B2 and P1 are connected but there is no direct access between them, while the common border of P1 and V1 is passable, and there is an entrance of P2 on V3.

In the example of figure 10, the topological structure, with its associated geometrical and symbolical information, is used to go from one point to another in the town. The example is a road section composed of two side-walks ($\mathbf{S}W_1$ and $\mathbf{S}W_2$), a car lane ($\mathbf{C}L$) and a zebra-crossing ($\mathbf{Z}C$). Figure 10 shows the symbolic representation of this scene, its corresponding topological structure and the 3D geometrical scene. In this scene a pedestrian wants to go from A to B.

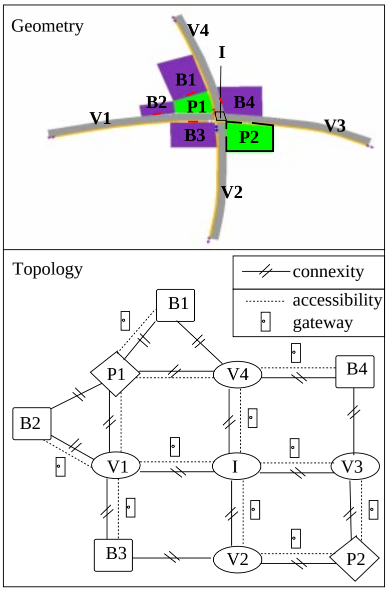


Figure 9: *Connexity and accessibility graph*

Two paths, the results of different graph traversals, can be generated:

1. Path 1: passes through spaces dedicated to pedestrian circulation;
2. Path 2: the shortest path.

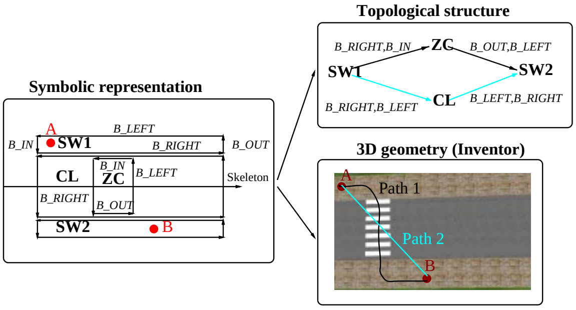


Figure 10: *Representations of a road section.*

The decision model of dynamic entities contains parameters representing *caution* and *laziness*. Depending on the value of those parameters one of the path is chosen. A variety of behaviour, even for this simple configuration, can then emerge, as the choice can be modified during the movement. This variety is a key element to obtain realistic simulations.

The topological structure can be used also for the vision process. Usually visual sensors used by autonomous actors are very simple (proximity sensor, linear sensor) or time consuming (Z-buffer algorithm). The topological structure allows a visibility graph to be created to accelerate the vision calculus.

4. The City Modeler: VUEMS

4.1. Introduction

VUEMS (Virtual Urban Environment Modeling System) is a modeler dedicated to the building of virtual cities[6]. Those virtual cities must be as realistic as possible. This realism can be obtained by using data from Geographical Information Systems (GIS) as input for the modeler. Most cities use this kind of information for urban planning. As a modeler, VUEMS is characterized by its inputs, its modeling tools and its outputs described in this section.

4.2. Inputs

The Modeling System uses, as inputs, different kinds of information supplied by City Technical Departments:

1. *Cartographic database*: this database (Ascodes file format used in various French cities) contains very precise information on the topography of the public domain of a city. This information is used by the technical departments for urban planning and project management. It concerns, for example, the topography of service networks like gas, electricity and water, but also the location of lamp-stands or trees. The database is structured in a set of different classes, which are themselves broken down into sub-families. Each element of the database is an object, representing a specific sub-family. For our current use, only superficial information is interesting. Some classes of objects are directly used to generate tridimensional objects, such as trees and lamp-stands, when other classes generate only bidimensional graphic information, visualized in specific layers of the viewing manager.
2. *Scanned map*: since the above mentioned database does not contain information such as horizontal and vertical road-signs, maps of roadways including this information are scanned into GIF files. Those scanned maps support the manual modeling of missing information.
3. *Traffic lights*: description of traffic lights, including their organization in different sequences and their synchronization locally at the crossroads and globally with other crossroads (depending on the time of day).

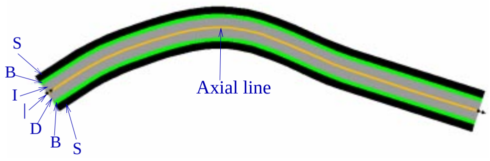


Figure 11: *Lane encoding in VUEMS.*

4.3. Modelling

By using a graphical user interface built upon the Ilog Views C++ library, the user creates complementary information to the GIS source. Three main families of objects are created:

1. *the road network*: a road is defined by a skeleton and an encoding (cf figure11). The road skeleton is modeled by a cubic spline curve; the radial encoding describes the lane width and type, and inter-lane marking. When roads intersect, the topological graph is updated and the resulting geometry is automatically computed (cf figure 12).

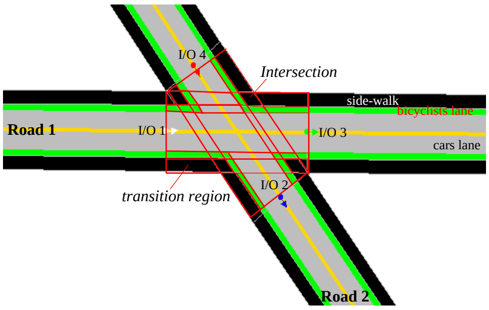


Figure 12: *Crossroads structure in VUEMS.*

2. *signals and urban furniture*: signals are associated to roads. The user creates a marker on the skeleton and associates an encoding to it (cf figure 13). This encoding gives the nature of the signal, its positioning (left or right side of the road) and its orientation. Urban furniture (lampstands, mailbox, etc.) is created by designating a 2D position in the city.
3. *buildings, parks and places*: they are 2D polygons with a list of attributes. For example, attributes of height, appearance, function and name are associated to a building.

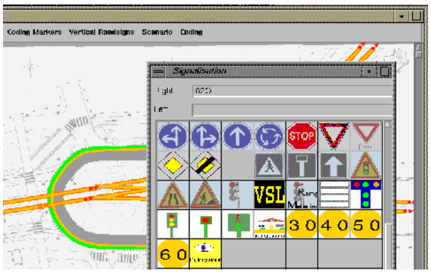


Figure 13: *Road-signs specification in VUEMS.*

4.4. Outputs

VUEMS produces two complementary outputs:

- the 3D geometric representation of the scene, described in Open Inventor format and including automatic texturing (cf figure 14).
- the Object Oriented database which contains all information which will be used by sensors and deliberative agents, in accordance with the city model presented in the preceding section.

Figure 14: *3D scene automatically generated by VUEMS.*

5. Examples

Currently, driving simulations are commonly limited to car and truck interactions on highways. Urban traffic has a higher degree of complexity, as it requires interactions on the same thoroughfare between not only cars, trucks, cyclists and pedestrians, but also public transportation systems such as busses and trams (cf figure 15)[8]. As our approach is modular, we have started to integrate all these transportation modes into GASP, our simulation platform[7]. Mechanical models of tram, trucks and cars are available, as well as a biomechanical model of pedestrians [21].

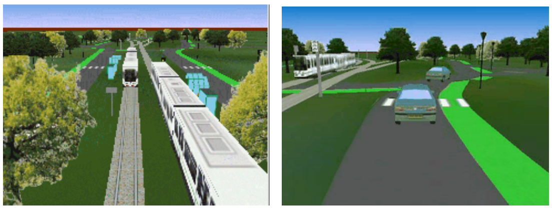


Figure 15: *Tram Simulation.*

Each simulation is composed of several modules working at their own frequency. The kernel of the simulation platform manages the data-flow and event based communication between modules and their synchronization. Modules are organized in a simulation tree and for example an autonomous pedestrian entity is composed of a visual sensor module, a behavioural module and a biomechanical module. The user can also control the motion of one pedestrian by using a joystick. The biomechanical module is then connected to a user interaction management module. The scene produced by VUEMS is loaded and is then available for use by all autonomous entities (see figure 16). Firstly, sensors can determine visible objects in their environment and then the behavioural module can access in the information on these visible objects.

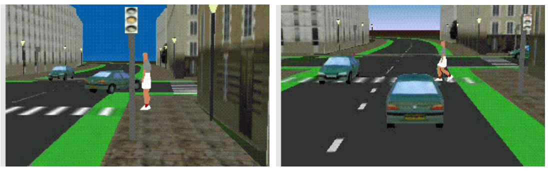


Figure 16: *Pedestrians and cars interaction.*

A test site has been used to study interaction between vehicles and pedestrians. This is a crossroads in Paris with heavy vehicles density and a great number of pedestrians, due to the underground subway station exits at the corners of the crossroads. Real data were collected, like pedestrian and vehicle flow density and percentages of vehicles changing direction at the crossing lane. For example, traffic flows in simulation were of 1650 vehicles per hour for vehicles going straightforward, and 2390 vehicles per hour entering the test site from the South. In this simulation, pedestrians were generated at each extremity of the zebra crossing and leave the simulation when they arrive on the opposite sidewalk (cf figure 17). This is due to the fact that we are only interested in the influence of pedestrians on the vehicle traffic flow. The south/north road is a three lane one way road. Because of pedestrians crossing the other street, the mean speed on the two extremity lanes slows down due to left and right turning manoeuvres.

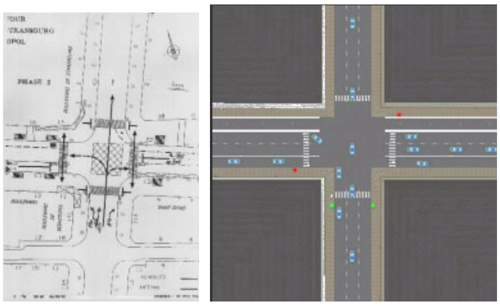


Figure 17: *Interaction between pedestrians and vehicles in a Test Site in Paris.*

Figure 18 illustrates the variety of trajectories taken by pedestrians on the same sidewalk. The name Left indicates that the pedestrian's walking direction is from the left to the right. Trajectory *Right1* is corresponding to a quiet pedestrian which is walking slowly and do not want to adapt its trajectory. Trajectory *Left3* is more angular, as the pedestrian has a higher desired speed, and has to adapt its trajectory to overtake or avoid other pedestrians. Figure 19 illustrates the activity of the pedestrian during the journey (0 stands for free walk, 1 for long distance trajectory adjustment, 2 for short distance trajectory adjustment, 3 for overtake). The five pictures in figure 20 illustrate the crossing of two pedestrian crowds on a sidewalk. The camera viewpoint corresponds to the subjective view of one of the pedestrians, which explain why pedestrians arriving from the opposite direction change their trajectory. The behavioural model of pedestrians includes social rules of interaction (minimize the interaction and choose in priority the left side to overtake). The ten pictures in figure 21 illustrate the difference in behaviour of two pedestrians who have to cross the same crossroads: the pedestrian wearing the black clothes decides to take a safe route by crossing the zebra crossing, while the other takes the shortest route. This example also illustrates obstacle avoidance.

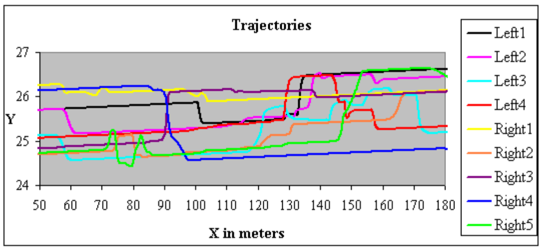


Figure 18: *Trajectories of pedestrians on a sidewalk.*

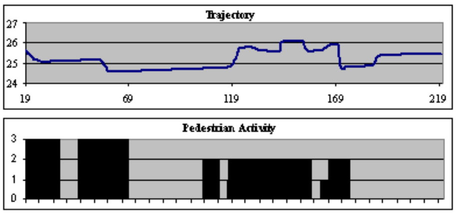


Figure 19: *Trajectory and activity of a pedestrian on a sidewalk.*

6. Conclusion and Future Work

In this paper we have presented a model of a virtual urban environment which enables us to perform some behavioural simulation, including different kinds of city actors (pedestrians, vehicles, public transportation systems). VUEMS, our city modeller, has also been presented. This modeller has already been used for different applications. First results concerning multi-modal interaction have been obtained, but more work should be performed on behavioural models. Our city model is currently 2D 1/2, as the ground floor is flat. We intend to use the Digital Terrain Model, but we have to take take into account the complexity both on the geometric database and on the impact on algorithms used for the behavioural model, as it will require real 3D vision and locomotion. Up to now we have focused our attention on the topological structure and on the nature of the interaction with areas. Another point of interest concerns cognitive modelling, which requires more information on the objects of the scene.

References

1. N. I. Badler, C. B. Phillips, and B. L. Webber. *Simulating Humans : Computer Graphics Animation and Control*. Oxford University Press, 1993. 1
2. S. Bayarri, M. Fernandez, M. Perez, and R. Rodriguez. Virtual reality for driving simulation. *Communications of the ACM*, 39(5):72–76, May 1996. 3
3. B.M. Blumberg and T.A. Galyean. Multi-level direction of autonomous creatures for real-time virtual environments. In *Siggraph*, pages 47–54, Los Angeles, California, U.S.A., August 1995. ACM. 1
4. D.C. Brogan, R.A. Metoyer, and J.K. Hodgins. Dynamically simulated characters in virtual environments. *IEEE Computer Graphics and Applications*, pages 58–69, 1998. 1
5. E. Cerezo, A. Pina, and F.J. Seron. Motion and behaviour modelling: state of art and new trends. *The Visual Computer*, 15:124–146, 1999. 1
6. S. Donikian. Vuems: a virtual urban environment modeling system. In *Computer Graphics International'97*, pages 84–92, Hasselt-Diepenbeek, Belgium, June 1997. IEEE Computer Society Press. 3, 4, 6
7. S. Donikian, A. Chauffaut, R. Kulpa, and T. Duval. Gasp: from modular programming to distributed execution. In *Computer Animation'98*, pages 79–87, Philadelphie, USA, June 1998. IEEE. 7
8. S. Donikian, G. Moreau, and G. Thomas. Multimodal driving simulation in realistic urban environments. In S. Tzafestas and G. Schmidt, editors, *Progress in System and Robot Analysis and Control Design*, pages 321–332. Lecture Notes in Control and Information Sciences (LNCIS 243), 1999. 7
9. N. Farenc, R. Boulic, and D. Thalmann. An informed environment dedicated to the simulation of virtual humans in urban context. In P. Brunet and R. Scopigno, editors, *EUROGRAPHICS'99*, pages 309–318. Blackwell, September 1999. 2
10. J. Funge, X. Tu, and D. Terzopoulos. Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. In *SIGGRAPH'99*, pages 29–38, Los Angeles, August 1999. 1
11. J.J Gibson. *The ecological approach to visual perception*. NJ: Lawrence Erlbaum Associates, Inc, Hillsdale, 1986. 2
12. E. Goffman. *Relations in public : microstudies of the public order*. New York : Basic Books, 1971. 2
13. C.M. Gold, P.R. Remmele, and T. Roos. *Algorithmic Foundations of Geographic Information Systems*, chapter Voronoï methods in GIS, pages 21–36. Lecture Notes in Computer Science 1340. Springer, 1997. 5
14. A. Gruen and X. Wang. Cc-modeler: a topology generator for 3-d city models. *ISPRS Journal of Photogrammetry & Remote Sensing*, 53(5):286–295, October 1998. 1
15. W. Jepson, R. Liggett, and S. Friedman. Virtual modeling of urban environments. *Presence Teleoperators and Virtual Environments*, 5(1):72–86, 1996. 1
16. M.E. Kallmann and D. Thalmann. A behavioral interface to simulate agent-object interactions in real-time. In *Computer Animation'99*, pages 138–146. IEEE Computer Society Press, 1999. 2
17. H. Kawasaki, T. Yatabe, K. Ikeuchi, and M. Sakauchi. Construction of a 3d city map from real-world video and digital maps. In R. Shibasaki and Z. Shi, editors, *International Workshop on Urban 3D/Multi-Media Mapping*, pages 165–168, The University of Tokyo, Japan, September 1999. 1
18. R. G. Lord and P. E. Levy. Moving from cognition to action : A control theory perspective. *Applied Psychology : an international review*, 43 (3):335–398, 1994. 2
19. P. Maes, T. Darrell, B. Blumberg, and A. Pentland. The alive system: Full-body interaction with autonomous agents. In *Computer Animation'95*, pages 11–18, Geneva, Switzerland, April 1995. IEEE. 1
20. G. Moreau and S. Donikian. From psychological and real-time interaction requirements to behavioural simulation. In *Eurographics Workshop on Computer Animation and Simulation*, Lisbonne, Portugal, September 1998. 2
21. F. Multon, L. France, MP. Cani-Gascuel, and G. Debunne. Computer animation of human walking : a survey. *Journal of Visualization and Computer Animation*, 10:39–54, 1999. 7
22. A. Newell. *Unified Theories of Cognition*. Harvard University Press, 1990. 2
23. Y.E. Papelis and S. Bahauddin. Logical modeling of roadway environment to support real-time simulation of autonomous traffic. In *SIVE95: the First Workshop on Simulation and Interaction in Virtual Environments*, pages 62–71, University of Iowa, Iowa City, U.S.A., July 1995. 3
24. D. Thalmann and H. Noser. Towards autonomous, perceptive, and intelligent virtual actors. In *Artificial Intelligence Today*, volume 1600 of *Lecture Notes in Artificial Intelligence*, pages 457–472. Springer, 1999. 1
25. H. Zhao and R. Shibasaki. A system for reconstructing urban 3d objects using ground-based range and ccd sensors. In R. Shibasaki and Z. Shi, editors, *International Workshop on Urban 3D/Multi-Media Mapping*, pages 175–182, The University of Tokyo, Japan, September 1999. 1

Figure 20: *Pedestrian crowd on a sidewalk.*

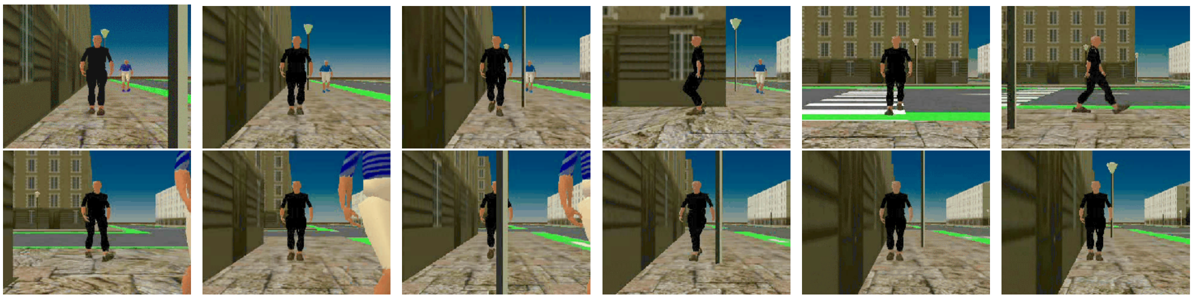


Figure 21: *Path planning and obstacle avoidance.*