Visual Comput (2007) 23: 409–418
DOI 10.1007/s00371-007-0099-6

ORIGINAL ARTICLE

Cledja Rolim
Alejandro C. Frery
Eliana Almeida
Evandro Costa
Luiz Marcos Gonçalves

Enhancing the experience of 3D virtual worlds with a cartographic generalization approach

Published online: 27 March 2007

C. Rolim (✉)
Centro Federal de Educação Tecnológica de Alagoas
Av. Barão de Atalaia S/N, Maceió, AL Brazil
cledja@yahoo.com.br

A.C. Frery[1] · E. Almeida[1] · E. Costa[2]
[1] CPMAT/IC, [2] IC
Universidade Federal de Alagoas
BR 104 Norte km 97
57072-970 Maceió, AL, Brazil
{acfrery, eliana.almeida}@pesquisador.cnpq.br, ebc@fapeal.br

L.M. Gonçalves
Universidade Federal do Rio Grande do Norte
DCA-CT-UFRN, Campus Universitario Lagoa Nova
59072-970 Natal, RN, Brazil
lmarcos@dca.ufrn.br

Abstract In this work we propose a new approach for fast visualization and exploration of virtual worlds based on the use of cartographic concepts and techniques. Versions of cartographic maps with different levels of details can be created by using a set of operations named *cartographic generalization.* Cartographic generalization employs twelve operators and domain-specific knowledge, being the contribution of this work their transposition to 3D virtual worlds. The architecture of a system for 3D generalization is proposed and the system is implemented. Differently from traditional cartographic processes, we use artificial intelligence for both selecting the key objects and applying the operators. As a case study, we present the simplification of the historical quarter of Recife (Brazil).

Keywords Cartographic generalization · Virtual worlds · Simplification · Artificial intelligence · Image processing

1 Introduction

Virtual worlds, virtual reality environments, or 3D worlds can be seen as a computational metaphor of worlds where people and objects can interact. They are used in entertainment, medicine, psychology (treatment of phobias), arts, and robot control, among other applications.

Virtual worlds are usually composed of many objects with varying degrees of complexity. The simplest possible object is a geometric shape, while complex objects can be formed by organizing simple ones and adding texture. The complexity of a world can be measured with, for instance, the number of polygons, colors and textures it comprises. Navigation and exploration of worlds with complex objects can be hampered since scene rendering can be slow, causing latency problems in the immersion experience. Some of the main problems concerned with navigation in urban virtual worlds are presented in the work of Bourdakis [2], while some solutions to those problems are discussed also by Bourdakis [1, 3] and Frery et al. [8]. Among the techniques that can be applied to solve navigation problems are the algorithms based on culling [4].

Generalization is an information abstraction process, very well defined for 2D objects in cartography. In a virtual world, this process may be responsible for simplifying and/or for removing objects considering, for example, the user position. Virtual reality generalizations could be obtained with, for instance, *levels of detail* – LODs, with versions of objects in progressive levels of complexity. If the user is far from the object (close to, respectively), the simple (detailed, resp.) version is displayed [15, 21]. In this paper we propose the transposition of cartography generalization techniques to the realm of virtual reality.

The current work deals with generalization for virtual worlds built in VRML [20]: a popular technology for the development of virtual worlds because it is free software, these worlds can be seen in the Internet, and the implementation process is fast and easy. Our proposal, though, can be used in any other virtual reality infrastructure.

Applications using detailed VRML worlds are generally slow because programmers frequently disregard the use of LODs. The process of generating LODs in VRML, however, can be complex, because each LOD has to be, in general, manually made and associated to a distance. In our proposal, the process of building and assembling LODs is made in an assisted manner using artificial intelligence and cartographic generalization tools and concepts.

LODs are usually obtained by polygon simplification [16]. Also, there are many techniques for selecting each version or LOD [5]; the one considered here is based on the object-observer distance. Figure 1 (from [6]) is an example of versions of an object with polygonal simplifications.

In digital cartography, the construction of generalizations is a process used to produce versions of cartographic maps with cartographic generalization. These versions are built considering the desired scale and theme, using specific cartographer knowledge, in order to limit the amount of detail to be shown in the map. That is, based on human perception capabilities, a given scale and the user purposes, maps are made both clean and with just enough detail for a nice visual feedback and an accurate analysis.

In this work, we show how cartography concepts and tools can be used in virtual reality to produce versions of worlds in agreement with the user's objectives. We use an expert system for selecting objects, and applying the operators and algorithms that we developed in order to get generalized versions of virtual worlds.

This paper, which extends the preliminary results presented in [9], is organized as follows. Section 2 presents the cartographic generalization and its operators. Section 3 presents the transposition of concepts and operators employed in cartographic generalization to virtual reality. Section 4 presents the 3D generalization system, and the validation of our proposal with a case study. Finally, Sect. 5 presents the conclusions and future works.

2 Cartographic generalization

Cartographic generalization can be defined as a set of procedures that are applied for the construction and visualization of models (typically secondary), aiming to improve the interpretation of the information. Cartographic generalization is employed when new maps in new scales are needed. It is concerned with the ways the information is shown, emphasizing, distributing, and deleting features as necessary. This process relies on the cartographer's knowledge about the requirements and the desired scale.

Figure 2 presents the main high-level generalizations that occur in cartography.

The *Primary Model* or *Generalization Object* [13] is the information about real world that experts in topography and photogrammetry, for example, judge important for constructing cartographic products, e.g., relevant geographic and man-made objects, without a prior knowledge about the future use of this information. A possible application is the production of touristic maps, where the relevant information consists of landscapes, historical sites and resources; see Bologna's city customizable map at http://sit.comune.bologna.it/tourism/ricerca.htm. Each product that can be extracted for a specific domain is called a *Secondary Model* and the process of producing it is the *Secondary Model Generation*. Here, the models cannot be completely finished for visualization and interaction with the final user.

Fig. 1. Versions of a rabbit by polygonal simplification [6]

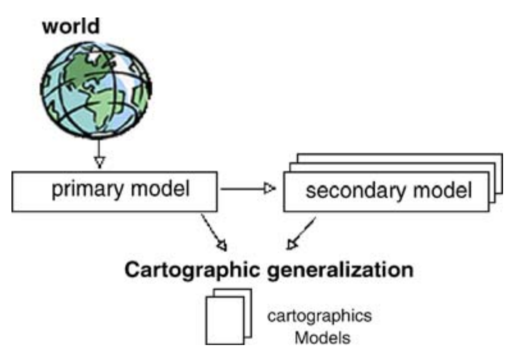


Fig. 2. Cartographic generalization

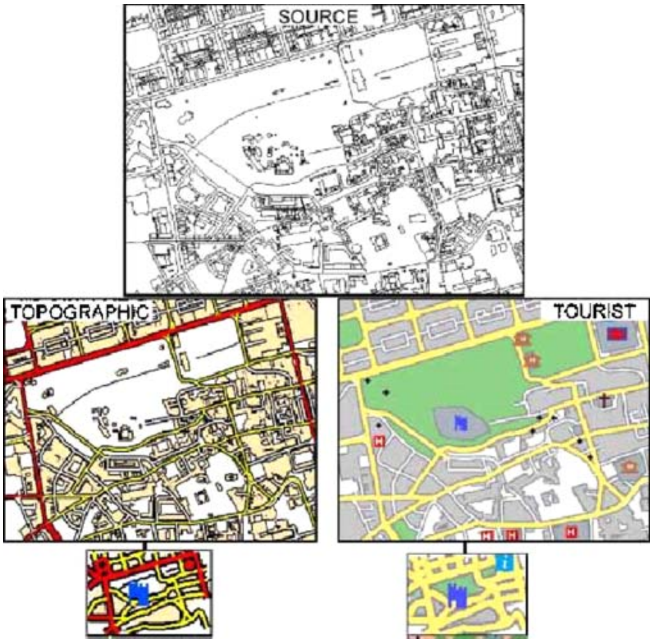


Fig. 3. Generalizations [10]

Figure 3 (top) presents a map generalized in two ways. The first (left) considers topographic features with emphasis on the distances and number of objects. The second (right) considers the touristic features, enhancing important objects in the area. Note that the first is a topographic map and the second a touristic map.

McMaster and Shea [18] present twelve operators that can be applied in map generalization; each operator is responsible for changing the way information is presented, and they are applied by the cartographer using domain-specific knowledge. As can be seen in the work of Glover and Mackaness [10], the generalization is influenced by map theme, the objects and their context. These operators are:

Simplification (OP1): reduces the number of vertices employed to represent the element, preserving the original appearance (see Fig. 4).
Smoothing (OP2): small scale information is reduced, in order to eliminate disturbances and to capture the overall shape (Fig. 5).
Aggregation (OP3): joins nearby elements (Fig. 6).
Amalgamation (OP4): joins nearly contiguous and similar areas, by eliminating borders between them (Fig. 6).
Merging (OP5): joins two or more close parallel lines into a single line.
Collapse (OP6): reduces the dimension of the representation of an object (Fig. 7).
Refinement (OP7): discards unimportant elements which are close to important ones (Fig. 8).
Exaggeration (OP8): increases the dimensions of elements that are considered important for the map (Fig. 9).

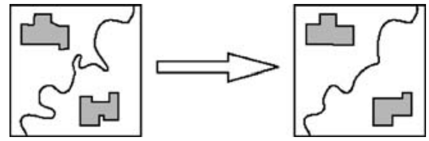


Fig. 4. Simplification operator [7]

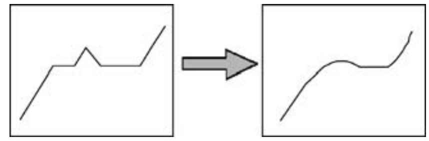


Fig. 5. Smoothing operator

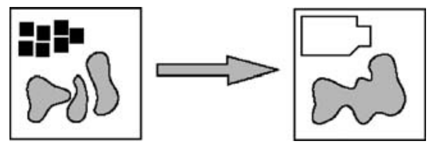


Fig. 6. Aggregation and Amalgamation operators [7]

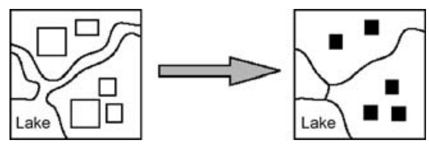


Fig. 7. Collapse operator [7]

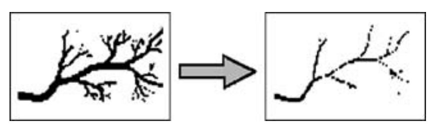


Fig. 8. Refinement operator (adapted from [18])

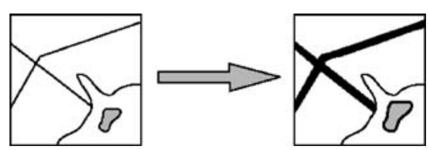


Fig. 9. Exaggeration operator [7]

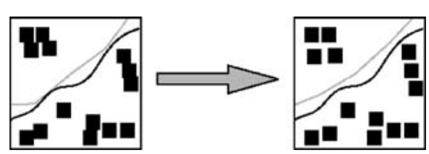


Fig. 10. Displacement operator [7]

Enhancement (OP9): increases the dimensions of symbols.
Displacement (OP10): shifts the position of a feature in order to make it distinct from others (Fig. 10).
Classification (OP11): groups objects with identical or similar characteristics into categories (Fig. 11).
Symbolization (OP12): changes objects (or categories) for symbols (Fig. 12).

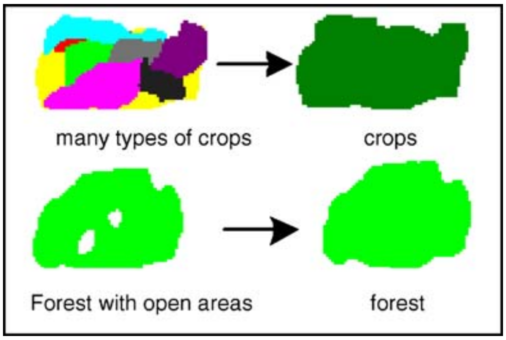


Fig. 11. Classification operator [19]

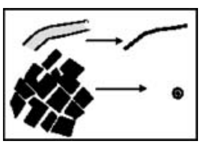


Fig. 12. Symbolization operator [19]

3 Applying cartographic generalization to virtual reality

This section shows how the concepts used in cartographic generalization can be applied to construct 3D virtual worlds. The previous cartographic generalization process, namely the *Generalization Object*, is the cartography counterpart of the development (construction and implementation) of the virtual world in virtual reality. In this step, the programmer is responsible for the creation of a virtual world, including definition or declaration of the objects to be implemented. This step is considered as finished, done in a previous work [23].

We next present the transposition of the cartographic generalization operators and then how domain-specific knowledge was modeled as rules of a rule-based expert system.

3.1 Cartographic generalization operators in virtual reality

In this section, we present how cartographic generalization operators can be applied in virtual reality, adapting 2D ideas and techniques to the 3D realm.

OP1: This operator consists of techniques for polygon simplification; related works in virtual reality can be found in the literature [11, 26]. Our proposal is to apply the simplification of line maps algorithms to the simplification of VRML objects built with the node `IndexedFaceSet`. These algorithms can be adapted to manipulate vectors and points in $\mathbb{R}^3$.
OP2: Coarse resolution imagery conveys enough information in the virtual world, along the idea of capturing the overall shape while eliminating small details. Such imagery is obtained using image filtering algorithms (typically low-pass) and subsampling [12]. When the user moves away from texture-enhanced objects, textures can be switched to ones with coarser resolution.
OP3, OP4, OP5: These operators can be built as a single operator in virtual reality, or as the initial stage of operators OP11 and OP12.
OP6: The result of applying OP1 iteratively.
OP7: Small objects (or objects defined by the user) may not be shown in the LOD.
OP8: The scale transformation in 3D objects.
OP9: 2D objects or textures already provided are subjected to the scale transformation.
OP10: The translation of 3D objects.
OP11: This is the most difficult operator to implement because it is necessary to find a way to group objects into categories according to their main features. If these instances are near, they can be grouped into a specific category. The classification will depend on the user viewpoint, so objects near to the user may not be categorized.
OP12: Objects are replaced by symbols or icons defined by the user, in agreement with predefined distances.

3.2 Expert systems

Expert systems aim to simulate the behavior of a human expert in activities related to problem solving and decision making, in specific domains [17, 22]. The architecture of a typical expert system can be seen in Fig. 13. It consists of three essential components: the knowledge base, the inference engine, and the explanation module. In addition, many systems also include a knowledge base editor and a user interface.

The knowledge base stores information about the subject domain. This knowledge is necessary for problem solving in a particular domain, and is formed by two parts: the general knowledge base and the problem-specific data. The general knowledge base contains the permanent information while the other, also known as "work memory," stores the temporary knowledge about a specific session of the system, as facts, partial conclusions, and other relevant information of the instance under consideration. The usual

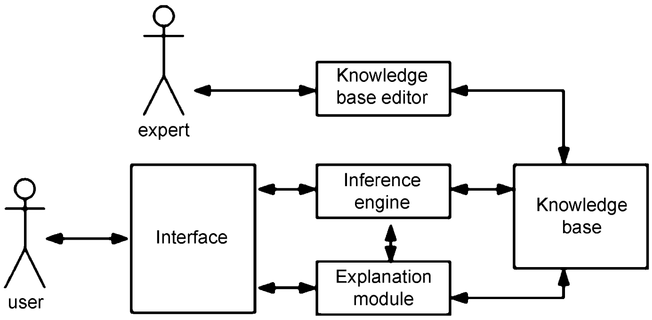


Fig. 13. Architecture of a typical expert system

form to represent the information in the knowledge base is by means of *if-then* rules. This is the method used in this work.

The inference engine is responsible for contrasting the general knowledge base and the information about the problem in order to draw conclusions about specific situations presented to the system. Typically, it follows one of two reasoning strategies: forward or backward chaining. Forward chaining is data-driven and, therefore, involves working from evidence to conclusions. Backward chaining is goal-driven, working from hypothesis to evidence; the engine selects a hypothesis and looks for data to support or refute it. While running, the expert system can also ask the user questions about the session.

The explanation module is responsible for explaining the expert system reasoning to the user. These explanations include justifications for the system's conclusions.

The user interacts with the system through an interface in at least one of the following styles: natural language, a graphic interface, menus and question-and-answer. The choice depends on the requirement of the system and the user necessities.

The human expert is responsible for constructing and maintaining the knowledge base by inserting, updating and deleting information through the use of the editor.

The expert system used here is based on this architecture and was developed in the Java Language [25]; its inference engine uses backward chaining [17, 24].

A cartographer builds map generalizations using his/her knowledge and experience about the selection of objects and the application of operators. We modeled this knowledge using *if-then* rules, and these rules are applied to VRML virtual worlds for building 3D generalizations in the form of LODs. Also, objects that are unimportant for the theme must be discarded. So, two knowledge bases were modeled: one for the selection of important objects and the other for the application of operators. These rules are described in the next section that also explains these knowledge bases and how they are applied.

4 3D generalization system

Figure 14 shows the architecture of our generalization 3D system (`SisGen3D`). The snapshot represents the real world that feeds a SIG (Geographic Information System); VRML files are generated from this input for a specific area regardless of the application. The generalization 3D system is not concerned with these steps.

The *Representation Model* reads and identifies the objects in the virtual world. VRML files, already stored in

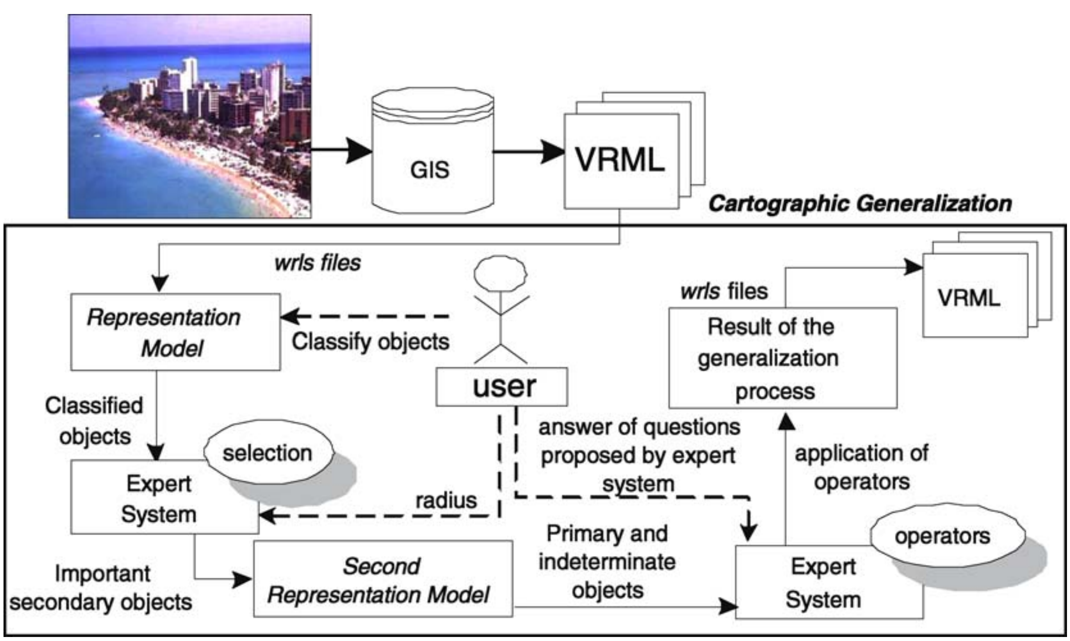


Fig. 14. Generalization 3D system architecture

a MySQL database, are the input of this step. VRML objects defined using the DEF node are labeled as "complex," otherwise they are considered "simple" by the system. This labeling is needed in order to decide which simplification strategy will be adopted.

The user classifies complex objects in one of three categories: *primary*, *secondary* and *indeterminate*. In the first are large objects, such as rivers and mountains. In the second are the medium-sized objects, as buildings and houses. The remaining objects are classified in the third category. The first category deals with geographic modeling, while the other two with urban modeling. This classification is important because it is taken into account by the operators.

All geographic objects are used in the final world. The expert system, with a knowledge base about object selection, works in the secondary category selecting the objects in agreement with the theme, e.g., tourism in our case study. Nothing is done on the third category.

A rule for selection of secondary objects is shown in Table 1: the system looks for keywords in the object name, as "museum" and "restaurant," and checks if there are other close objects (the user provides the radius). If there is not a keyword in the name, but there is only the object in the area of interest (in agreement with the radius), this means that the object has importance for the region, e.g., castles and churches can be far from urban areas and have to be included even if they are not labeled.

The *Second Representation Model* is similar to the first, but with less objects (unimportant objects were eliminated in the first representation model). The expert system, using the knowledge base regarding the application of operators, applies the operators to the virtual world. In our implementation, the system builds objects defining three levels of distance, namely `LOD1`, `LOD2` and `LOD3`, that will be employed at distances defined by the user. The user interacts in this step, defining which operators will be applied and the objects category (primary, secondary and indeterminate). Table 2 presents the LODs and their relation with the generalization operators.

As we can see in Table 2, `LOD1` refers to the simplification operator, `LOD2` refers to the smoothing operator and `LOD3` to the symbolization operator. The user can choose the simplification algorithm to be used.

The result of the generalization process is stored into VRML files for user visualization. If the result is not good enough (visually), the process can be done again. The system is developed in Java Language [25].

4.1 Implementation of operators

This section presents details of the implementation of generalization operators in the context of virtual reality. Our main objective is system validation, not the implementation of all the operators. Each operator can be implemented using those algorithms that better suit the user's needs and prior experience.

Table 1. Rule for secondary object selection

Number	Rules
1	*If* there is keyword in object name *or* there are no others secondary objects nearby *then* Select object

Table 2. Main rules for operators application

Number	Rules
1	*If* `LOD1` <> 0 *then* apply the simplification operator
2	*If* apply the simplification operator *and* simplify primitives *then* select the object category
3	*If* apply the simplification operator *and* simplify `IndexedFaceSet` *then* select the object category
4	*If* simplify `IndexedFaceSet` *then* select the `IndexedFaceSet` algorithm simplification
5	*If* `LOD2` <> 0 *then* apply the smoothing operator
6	*If* apply the smoothing operator *then* select the object category
7	*If* `LOD3` <> 0 *then* apply the symbolization operator
8	*If* apply the symbolization operator *then* select the object category

The operators OP8, OP9 and OP10 are provided in VRML through the `Transform` node, since it provides scale and translation operations. The implemented operators were simplification, smoothing and symbolization:

Simplification: It is composed of two algorithms, namely primitive simplification and `IndexedFaceSet` simplification. The first is responsible for simplifying VRML primitives: `box`, `sphere`, `cone` and `cylinder`. A VRML primitive can be built with many faces. For instance, a sphere can be rendered with sixty faces requiring computational resources. This algorithm produces a simplified version of each primitive by projecting it onto a convenient plane (see Fig. 15). The new flat object, built as an `IndexedFaceSet`, inherits some of the properties of the original primitive, e.g., color, texture and size. Spheres become circles, cones become triangles and boxes become rectangles. Figure 16a presents an object built with VRML primitives, and Fig. 16b presents the result of the simplification primitive algorithm; they look alike from a certain distance and certain viewpoints. Objects built with `IndexedFaceSet` have, in most cases, many, even millions of faces. Many of

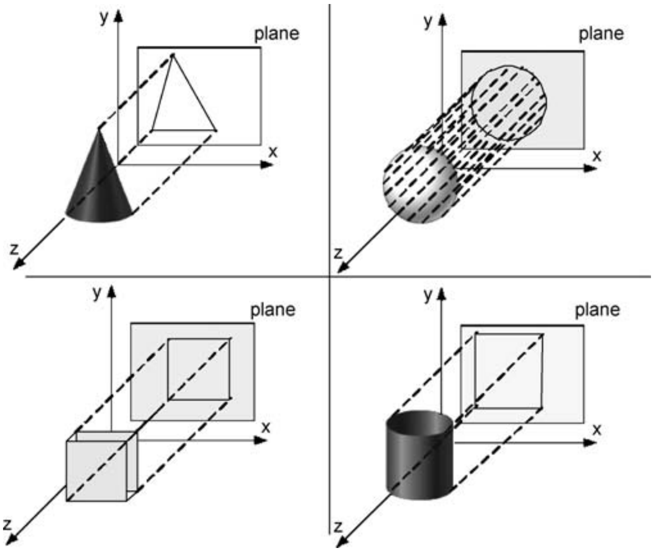


Fig. 15. VRML primitives and projections

these objects are the result of exporting from 3D CAD platforms, and they are comprised of triangles. The literature on mesh triangle simplification is vast, but the works by Guéziek et al. [11] and Vieira et al. [26] provide particularly useful insights and techniques. The `IndexedFaceSet` algorithm simplification reduces the number of faces of the original object. Figures 17a,b from [11], show a horse described by an `IndexedFaceSet` object (4350 triangles) and its simplification (247 triangles), respectively. For our system, we adapted the line simplification Lang Algorithm [18].

Smoothing: This operator consists of image processing techniques, and is applied to textures to produce new, similar, however smaller textures in two steps: applying a low-pass filter [14] to blur the image and then sampling it. Figure 18 presents an example of this operator. Left image is the original one, top right image is the blurred one and bottom right image is the subsampled one. Their sizes are, respectively, 118 kB, 52 kB and 12 kB. The subsampling rate is $1 \div 3$.

Symbolization: This operator changes the objects for symbols which, in turn, are textures over single-faced `IndexedFaceSets`. Each texture is related to a keyword, and the system looks for keywords in the object name. If a texture with the same keyword as the object is found, a symbol is created. Figure 19 (left) presents an object called “statue” built with nine box primitives, two spheres and three `IndexedFaceSets` (each one with millions of points); it has 191 kB and was inserted in the system with the keyword statue. The corresponding symbol is shown in the right top, and right bottom is its visualization from some distance. The symbol requires only 3 kB.

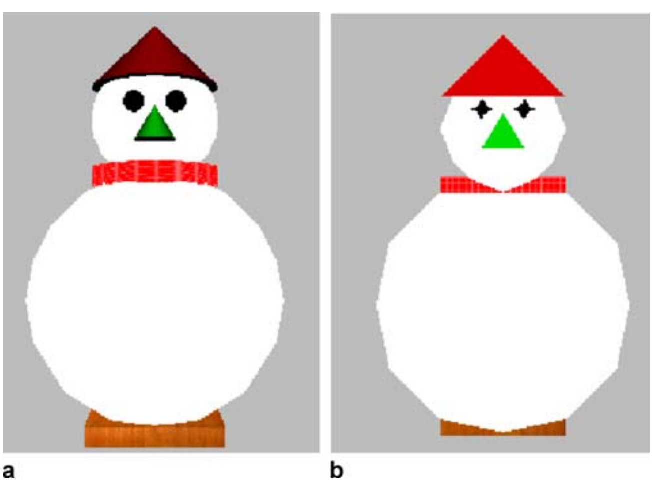


Fig. 16a,b. Original and simplified object by projection. **a** Original VRML object. **b** Simplified object

Regarding the influence of the viewpoint on the quality of simplified objects, this can be alleviated by choosing a set of projection planes: one for each sensible viewpoint. Our implementation uses a single plane, namely the one that produces the best result for a walking tour.

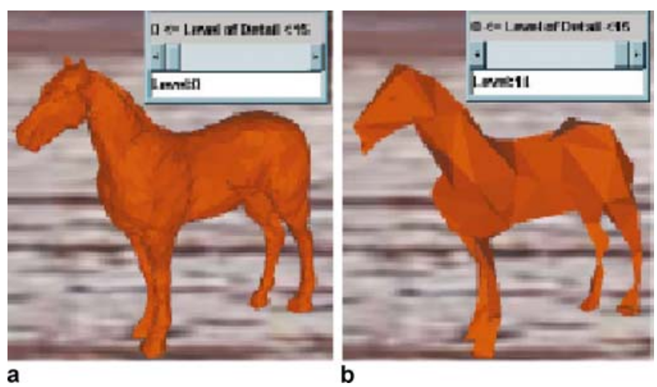


Fig. 17a,b. Simplification of an `IndexedFaceSet` object [11]. **a** Original object. **b** Simplified object

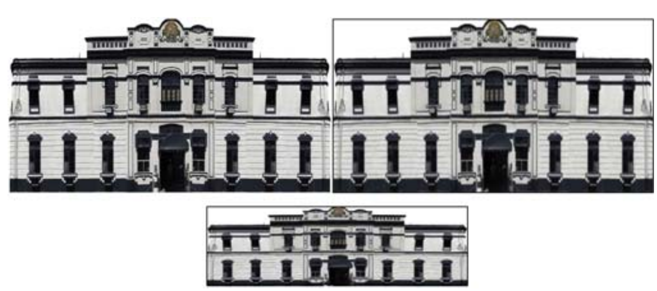


Fig. 18. Original, blurred and sampled images

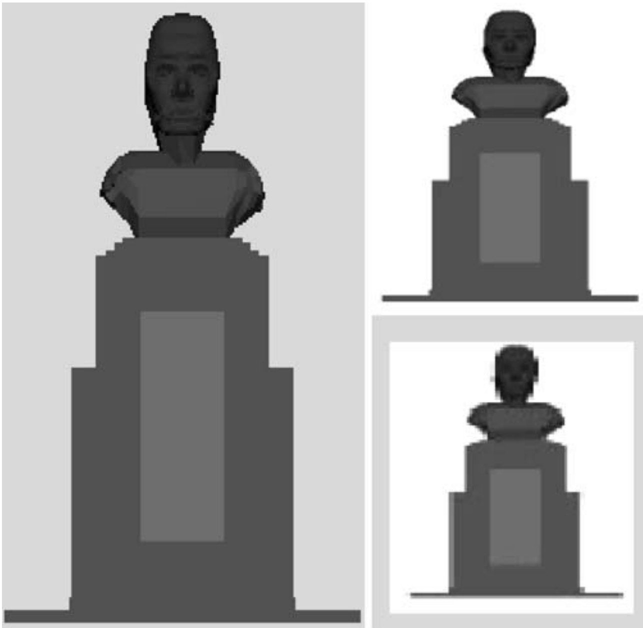


Fig. 19. Example of symbolization operator

4.2 Case study: historical quarter of Recife, Brazil

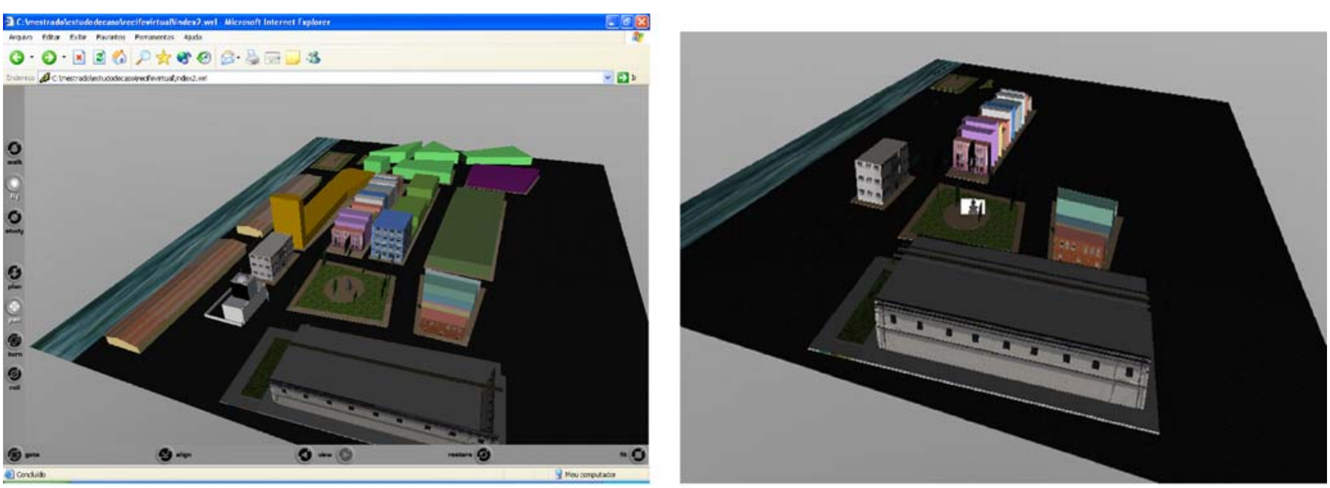
The developed system was tested on a large virtual world depicting the historical quarter of the city of Recife, Brazil, available at http://geocities.yahoo.com.br/recifeantigovirtual/index.wrl. This world comprises more than a dozen small graduate and undergraduate projects, each aiming at developing a specific use of VRML and associate technologies (sound and image editing, field data acquisition, etc.). This wide range of interests produced a heavy to load world, with all kinds of information spread in different scales and supports. As a practical result of using the proposed methodology, navigation was enhanced for an application aiming at tourism, reducing by a factor of 4 the required time to explore it.

Figure 20a exhibits this historical quarter. In agreement with Sect. 4, the `SisGen3D` reads and identifies the objects in the virtual world. Some of them are: *ground*, *sea*, *trees* and *buildings*.

After the initial classification, the `SisGen3D` applies the expert system with the knowledge base about object selection in the secondary objects, with tourism as the theme. About ten percent of the objects were selected for this application, the rest were considered irrelevant for the next stages and final visualization.

An important monument was associated to an icon by using the symbolization operator (the process shown in Fig. 19), and the expert system was applied using three operators: simplification, smoothing and symbolization. Primary objects were submitted to the smoothing operator, and indeterminate objects were subjected to `IndexedFaceSet` simplification. One of the primary objects that was subjected to simplification was the texture that represents the sea (the blue strip to the left); after applying OP2, the new version is more than ten times smaller than the original one, with no visual loss of information. Secondary objects were treated by symbolization and primitive simplification. Figure 20b shows the result of the generalization process in this virtual world.

This result, available at http://geocities.yahoo.com.br/recifeantigovirtual/index5.wrl, is in agreement with expectations about a touristic application, since relevant information was preserved and enhanced, while elements with no touristic interest were deleted.



a b

Fig. 20a,b. Two versions of Recife's historical quarter. **a** Original world. **b** Generalized world

This case study was done with an AMD AthlonTM (XP) 1.57 GHz processor with 256 MB of RAM computer running Windows XP. For illustration, the statue was originally defined with nine `Boxes`, two `Spheres` and three `IndexedFaceSets`: with 4500, 285 and 48 faces, respectively. This statue was transformed into an icon (symbol) of 3 kB.

The time of visualization of the original virtual world was approximately five seconds, while less than two seconds were required for the generalized world. Depending on the objects complexity, the time of the generalization process can be large. In particular, the simplification of `IndexedFaceSet` objects is computationally expensive, but the results are worthwhile.

5 Conclusion and future works

In this paper, the simplification of complex scenes and objects is managed by means of artificial intelligent tools that coordinate the use of cartographic generalization ideas transposed from their 2D realm to 3D ambients. This proposal aims at simplifying complex 3D worlds in order to enhance the user experience through the navigation and information retrieval in environments whose content is focused on his/her desires and intentions.

Specific domain simplified worlds can be generated with this technique. A system for implementing this transposition was developed, and it was successfully used in reducing the time spent in the exploration of a virtual world, after it was simplified for touristic applications.

Several areas, not only virtual reality, frequently use simplification ideas in order to allow fast visualization of data. As we have shown, different techniques, as image processing, symbolization and simplification, can be combined in order to obtain the performance required in visualization or, more generally, exploration tasks. So, with this work, we have accomplished the initial purpose of showing that cartographic generalization is a suitable approach for visualization and exploration improvement in virtual reality and other application areas, such as image synthesis in computer graphics.

For future works we will develop other operators for extending the functionality of the system. We also intend to extend this proposal to other domains as, for instance, image processing and computer graphics; in order to do so we will augment the power of the knowledge base. Also, we will perform more extensive tests, using other themes besides “tourism” in other worlds.

Acknowledgement The authors are grateful to the referees for their comments that improved this paper.

References

1. Bourdakis, V.: CAAD Towards New Design Conventions, chap. Virtual Reality: A Communication Tool for Urban Planning, pp. 45–59. Technical University of Bialystok (1997)
2. Bourdakis, V.: Navigation in large VR urban models. In: Virtual Worlds, pp. 345–356. Springer, Berlin Heidelberg New York (1998)
3. Bourdakis, V.: On developing standards for the creation of VR city models. In: H. Penttila (ed.) Architectural Information Management: eCAADe2001 Proceedings, pp. 404–409 (2001)
4. Cohen-Or, D., Chrysanthou, Y., Silva, C., Durand, F.: A survey of visibility for walkthrough applications. Proceedings of EUROGRAPHICS’00, Course Notes (2000). URL http://citeseer.nj.nec.com/cohen-or00survey.html
5. Constantinescu, Z.: Levels of detail: An overview. Nonlinear Analy. Model. Control **5**, 39–52 (2000)
6. Erikson, C.M.: Hierarchical levels of detail to accelerate the rendering of large static and dynamic polygonal environments. Dissertation, Department of Computer Science, University of North Carolina, Chapel Hill (2000)
7. ESRI: Automation of Map Generalization – The Cutting-Edge Technology. White Paper: Environmental Systems Research Institute, USA (1996). URL http://www.geo.ulg.ac.be/internet/Fichiers/Interet/General/Cartographie/AutoMapGenESRI.pdf
8. Frery, A.C., Kelner, J., Moreira, J., Teichrieb, V.: Satisfaction through empathy and orientation in three-dimensional worlds. CyberPsychol. Behavior **5**(5), 451–459 (2002)
9. Frery, A.C., Rolim, C., Costa, E., Almeida, E.: Cartographic generalization in virtual reality. In: XX International Society for Photogrammetry and Remote Sensing Congress, pp. 200–205. ISPRS, Creaturk Interactive, Istanbul (2004)
10. Glover, E., Mackaness, W.A.: Dynamic generalisation from single detailed database to support web based interaction. In: ICA/ACI 1999 Proceedings, Actes, Ottawa, Canada, pp. 1175–1183 (1999)
11. Guéziec, A., Taubin, G., Horn, B., Lazarus, F.: A framework for streaming geometry in VRML. IEEE Comput. Graph. Appl. **19**(2), 68–77 (1999)
12. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice-Hall International Editions, Englewood Cliffs, NJ (1989)
13. João, E.M.: Causes and Consequences of Map Generalisation. Taylor & Francis, London (1998)
14. Lim, J.S.: Two-Dimensional Signal and Image Processing. Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs (1989)
15. Luebke, D., Reddy, M., Cohen, J.D., Varshney, A., Watson, B., Huebner, R.: Level of Detail for 3D Graphics. The Morgan Kaufmann Series in Computer Graphics. Kaufmann, San Francisco (2003)
16. Luebke, D.P.: A developer’s survey of polygonal simplification algorithms. IEEE Comput. Graph. Appl. **21**(3), 24–35 (2001)
17. Luger, G.F.: Artificial Intelligence: Structures and Strategies for Complex Problem Solving, 4th edn. Addison-Wesley, Boston (2001)
18. McMaster, R.B., Shea, K.S.: Generalization in Digital Cartography. Associaton of American Geographies, Washington (1992)
19. Meng, L.: Automatic Generalization of Geographic Data. Svenska Försvarsmakten (Swedish Armed Forces), Stockholm (1997)
20. Ramos, A.L., Nadeau, D.R., Moreland, J.L.: VRML 2.0 SourceBook, 2nd edn. Wiley, New York (1997)
21. Reddy, M.: A survey of level of detail support in current virtual reality solutions. Virt. Real. Res. Develop. Appl. **1**(2), 85–88 (1995)
22. Rodriguez-Bachiller, A.: Expert Systems and Geographic Information Systems for

Impact Assessment. CRC, Boca Raton, FL (2003)
23. Rolim, C., Frery, A.C., Costa, E., Almeida, E.: Aplicação da generalização cartográfica em realidade virtual. In: Proceedings of SVR – VI Symposium on Virtual Reality, pp. 369–371. Ribeirão Preto, SP, Brazil (2003)
24. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice-Hall, New York (2003)
25. Sun Microsystems, I.: The source for Java (TM) technology (2006). URL http://java.sun.com
26. Vieira, W., Lopes, H., Tavares, G., Lewiner, T., Velho, L.: Fast stellar mesh simplification. In: Proceedings of SIBGRAPI 2003 – XVI Brazilian Symposium on Computer Graphics and Image Processing, pp. 27–34. IEEE Press, Washington, DC (2003)





CLEDJA ROLIM received the Computer Science degree from the Universidade Federal de Alagoas (Brazil) in 2001 and the MSc from the Universidade Federal de Pernambuco (Brazil) in 2004. She is currently associate professor at the Centro Federal de Educação Tecnológica (Alagoas, Brazil). Her research interests are virtual reality and mobile applications.

ALEJANDRO C. FRERY received the Electronics Engineer degree from the Universidad de Mendoza (Argentina) in 1985, the MSc in Applied Mathematics (Statistics) from the Instituto de Matemática Pura e Aplicada (Brazil) in 1990 and the PhD degree in Applied Computing from the Instituto Nacional de Pesquisas Espaciais (Brazil) in 1993. He has been full professor at the Universidade Federal de Alagoas, Brazil, since 2003. His main research interests are complex models and statistical computing.

ELIANA S. ALMEIDA has a major in Civil Engineering from the Universidade Federal de Alagoas. She obtained the PhD in Informatics at the Pontifícia Universidade Católica do Rio de Janeiro in 1998. Her research interests include formal methods, logic as a specification language, and formal aspects in scientific computing. Currently, she is associate professor at the CPMAT (Centro de Pesquisa em Matemática Computacional).

EVANDRO COSTA is associate professor at the Universidade Federal de Alagoas. His research interests include agent systems, e-learning and e-commerce systems, intelligent tutoring systems, groupware, computer supported collaborative learning and software composition. He is a member of the Brazilian Computer Society and also Society of Artificial Intelligence in Education. He obtained his DSc in Electrical Engineering from the Universidade Federal da Paraíba, Brazil, in 1997.

LUIZ MARCOS G. GONÇALVES holds a PhD in Systems and Computing Engineering from the Universidade Federal do Rio de Janeiro. He is associate professor at the Universidade Federal do Rio Grande do Norte (Brazil). He has received prizes such as the South America ACM Programming Contest Gold Medal, and he has served on several committees (Robocup Latin America Council, IEEE Latin America Robotics Sub-Committee). His research interests include several aspects of graphics processing.