Automatic Creation of 3D Virtual Environments Using Artificial Intelligence and Scene Graphs

Juliana Gouveia Denipote\textsuperscript{1,3}, Maria Stela Veludo de Paiva\textsuperscript{1}, Patricia Shirley Herrera Cateriano\textsuperscript{2,3}

\textsuperscript{1}Escola de Engenharia de São Carlos EESC/USP- São Carlos, Brazil
judeini@sel.eesc.usp.br, mstela@sel.eesc.usp.br

\textsuperscript{2}Instituto de Ciências Matemáticas e de Computação ICMC/USP - São Carlos, Brazil
patricia@icmc.usp.br

\textsuperscript{3}Cientistas Associados Ltda. - São Carlos, Brazil
juliana.gouveia@cientistasassociados.com.br

Abstract. The long time spent during the 3D modeling of the environment objects is the main problem of creating virtual environment (VE). Automatic generation tools represent good alternative ways to increase performance when a VE is being built. Furthermore, the representation of the VE needs to be of good quality and fast for an interaction in real time. This work describes an automatic generation tool using scene graphs and neural net to create a VE from a real city photographs.

1. Introduction

The creation of virtual environments (VE) should replicate the most visual aspects as possible from the real objects that are being represented. A VE application, very cited in the nowadays literature, is the representation of real cities [1], [2]. That kind of VE is used in many fields, such as entertainment, tourism, 3D geographic information systems, among others.

Although VE has been used in many areas of applications, its creation is a complex task because of the many objects that can be included in the 3D environment such as houses, squares, buildings, roads. The task complexity is aggravated when large urban areas have to been modeled. To solve that problem, the solution proposed is to model the city objects in an automatic way, using image segmentation methods and a neural net to recognize and classify city elements in urban aerial photographs.

Another important issue is the definition of the criteria used during the navigation in VE. One interesting criterion is the detail level of the objects, in order to maintain as much as possible a realistic environment. For that purpose, this work considered the optimization of the rendering process through the representation of three-dimensional synthetic VE using scene graphs. It should allow a faster and more efficient processing of the VE by the application, maintaining a good quality in the visualization and allowing visualization and interaction in real time.

2. Virtual Environment Creation

It was used a set of photos from an urban area, acquired in an easy and cheap way to guarantee the viability of the application. The photographs were the same used by Hughet [1] and were taken by a common digital camera Kodak DX3700 that has 3 Megapixels precision. However that camera is more accessible, being approximately forty times cheaper than the cameras of medium format usually used in application of this type, as those used in [3]. The photographs were taken in a low altitude, 500 meters from the ground. Figure 1(a) shows one of these photographs. Tests were made with images taken from
Google Earth [4] too, which is free tool for personal use and makes available photograph images taken by satellites and aircrafts in the last three years. Figure 1 (b) shows an example of Google Earth images.

On images of Figure 1 are applied Computational Vision and Artificial Intelligence methods to obtain the expected result of the image processing as showed in Figure 2.

In Figure 2, first, the Image Processing using Computational Vision techniques must be made. The image processing is divided in three activities: 1) pre-processing: the image’s noise is decreased and the quality is enhanced. 2) segmentation: objects of interest as buildings, pavements and green parks are found by segmentation methods. 3) characteristics extraction: some characteristics of the objects found in the segmentation activity are extracted such as color average, perimeter, height, width, pixel quantity, etc. Those characteristics will be the input parameters for the neural net.

The characteristics found in the image processing will be the entrance patterns for the neural net, which has to classify the objects in three different classes: houses, buildings and squares. The neural net
architecture used in this work was Multi Layer Perceptron (MLP). For this work, the used net has the following topology: five entrance characteristics (height, width and pixel color represented by RGB standard); five neurons in the hidden layer due to the amount of patterns characteristics; and three neurons in the output layer, because the patterns were classified in the classes: houses, buildings and squares.

There is a 3D models database to store different models for each of the three classes of objects. As long as each object of the image was properly identified and classified by the neural net, a 3D object is chosen in a random way in the database. For example, if it is identified that the object is a house, one 3D model is chosen among the ten house models. Then, the position of the model to be inserted in the VE is determined by the position of the corresponding object in the image.

3. Virtual Environment Navigation

An interesting and very efficient approach to optimize the process of visualization is the use of scene graphs. The scene graphs store data, creating a tree with the elements that conform the VE. This tree will allow a more accurate processing of the information that should be visualized, discarding those that are not in the user’s visual area. The leaves of the tree store the geometry of the scene and the nodes of higher levels store complementary information. Two techniques were used in the scene graphs in order to increment the render process speed: Impostor and LOD (Level Of Detail).

Impostor works in scene graphs as a particular type of node that intends to substitute part of the scene’s geometry by images of the same geometry computed by the system. The substitution is determined by a distance parameter that is predetermined in the application (a threshold distance). The use of the Impostor technique represents a lower cost of processing, as it is not necessary to process over and again the complete geometry of the original scene.

The LOD technique is implemented as another type of node, used as well by the Impostor technique. LOD allows the substitution of objects in the scene. In this technique are used two different models of the same object: one more detailed which have a high number of polygons and other simpler model, represented by a low number of polygons. The replacement of one by another depends on the threshold distance between the object and the viewpoint.

4. Tests and Results

It was observed that the network reached an excellent classification with an estimative error of 9% for houses class, 12.28% for buildings class and 0.29% for square class. A momentum term was used to accelerate the neural net training processes. Analyzing the results, it is possible to observe that the samples quality influence the pattern classification. The patterns normalization is a way to enhance the samples, and it was confirmed that normalizing the dates, the accuracy of classification reached the rate of 100% of the tested samples, while without the normalization, the rate was 80%.

The system implemented used the Open Scene Graph (OSG) library [5] to create and manipulate the VE. The use of LOD and Impostor techniques increase the performance in more than 50% of the visualization’s time. The experiments with an environment composed by over one million polygons reached a visualization time, before the optimization, about 32 and 42 fps (frame per second) for a dual Xeon machine with 1GB of RAM. This time was improved after the use of LOD and Impostor and it achieved a frame rate around 65 to 72 fps. Some scenes of this VE can be seen in the Figures 4 and 5.

5. Conclusion

The MLP network is a good solution for the problem of the automatic VE creation with low error rates and an acceptable time for the network training processing after the momentum inclusion. For a best
performance of the MLP training, the image processing must reach a satisfactory accuracy during the segmentation stage in order to result on high quality samples for the network.

The use of scene graphs with different techniques of computer graphics allows to create and visualize complex models, with a high number of polygons, in a much reduced time. The use of libraries based on scene graphs makes easier the development of graphic applications and allows the improvement of the performance in terms of the application’s time. The applications that use this kind of interface allow the user to better understand the data to be studied and obtaining more complex information, because it is possible to offer a great number of information in an organized way.

![Figure 4: A house block inserted in the VE. (a) Close view of the building (b) Far view of the same building as sprite generated by the system with Impostor (in detail)](image)

![Figure 5: Navigation from the VE. (a) A 3D model inserted in the VE represented by the LOD model of full detail (b) the same building in a low level of detail.](image)

6. References


