



ARQUEOLÓGICA 2.0

CONGRESO INTERNACIONAL DE ARQUEOLOGÍA E INFORMÁTICA GRÁFICA, PATRIMONIO CULTURAL E INNOVACION
INTERNATIONAL CONGRESS ON ARCHAEOLOGY, COMPUTER GRAPHICS, CULTURAL HERITAGE AND INNOVATION

Proceedings of the 8th
International Congress
on Archaeology,
Computer Graphics,
Cultural Heritage and
Innovation
'ARQUEOLÓGICA 2.0'
in Valencia (Spain),
Sept. 5 – 7, 2016

VIRTUAL CITIES INHABITED BY AUTONOMOUS CHARACTERS: A PIPELINE FOR THEIR PRODUCTION

Ana Paula Cláudio^{a,*}, Maria Beatriz Carmo^a, Alexandre Antonio de Carvalho^a, Willian Xavier^a, Rui Filipe Antunes^{a,b}

^a BioISI - Biosystems & Integrative Sciences Institute, Faculdade de Ciências, Universidade de Lisboa, Campo Grande, 1749-016 Lisboa, Portugal. apclaudio@fc.ul.pt; mbcarmo@fc.ul.pt; fc47014@alunos.fc.ul.pt; fc34918@alunos.fc.ul.pt; rfantunes@fc.ul.pt

^b MIRALab, University of Geneva, Switzerland.

Abstract:

Creating 3D reconstruction of the past is a challenging task requiring a vast set of technical skills, with teams usually including historians, 3D artists and IT technicians. In this paper, we describe our combined efforts to work under a low-budget pipeline to make a 3D simulation of the medieval village of Mértola in the south of Portugal. The work we describe in this paper is part of a larger project called "BIHC - Bio- Inspired Human Crowds" whose partial objectives are the development of a set of tools for the animation of 3D reconstructions of cultural heritage sites. We have focused on developing a lightweight model of the village that would allow the incorporation of virtual autonomous inhabitants in a simulation running in real-time and visitable by online audiences in both i) bird's-eye and ii) first-person points of view. The contribution of this paper is the identification of a set of tools and stages that can be used to obtain the final interactive environment recreating the inhabited ancient village.

Key words: virtual archaeology, digital archaeology, cultural heritage, 3D reconstruction, 3D data interoperability

1. Introduction

Amongst the daunting difficulties of creating 3D simulation of ancient sites are the costs associated with these types of simulations. One of our goals was to address this issue by building upon a pipeline of tools and methods available for free in an educational setting or with campus licenses, and basing it upon volunteer work. This work resulted from a multidisciplinary team of volunteers, including archaeologists from the Campo Arqueológico de Mértola, and students and teachers from Faculdade de Ciências da Universidade de Lisboa working as part of their curricular activities.

The former group of participants have provided useful insights and historically relevant guidelines about the architecture and culture of the ancient site. This input included information about the civil housing and military structures in the urban landscape, such as the defensive wall and the watchtower. Based on this information, students' activities were then organized by teachers in such a way as to include tasks that would allow the modeling of these features of the urban scene.

As a result, initially, a sample of a few houses was built using Autocad (www.autodesk.com) and later textured using 3ds Max (www.autodesk.com) and Photoshop (creative.adobe.com). Students have made three versions of each house. These variations allow us to

integrate different levels of detail: i) a lightweight version with a geometry of a reduced number of polygons, to be visible at a distance; ii) a second version, where houses only have the exterior walls but have a higher definition with detailed roofs and doors, to be seen at close range; iii) another version, also in high definition, with the interior walls and props, allowing a visit using a first-person point-of-view. This restricted set of houses provided us with templates to use with CityEngine (www.esri.com) to create general areas of the village where we did not have much available information. A second group of students used Autocad to model the iconic watchtower, again as part of their curricular program. Finally, a third group is currently modeling the agglomerate of fifteen houses of the Alcáçova (citadel) in 3ds Max. The model of the defensive wall is being improved based on a more accurate description.

The second step of this reconstruction departed from these architectural structures and entailed using CityEngine to generate the combined landscape. This part of the simulation comprised of the architectures described as well as the topographical features and vegetation of the site. Finally, this 3D model was exported to a format compatible with Unity3D (unity3d.com), a software tool to develop videogames. Then, animated virtual characters were incorporated and some simple behaviors were implemented.

* Corresponding Author: Ana Paula Cláudio, apclaudio@fc.ul.pt

The models of the characters and their basic animations were obtained from specialized software companies like Daz 3D (www.daz3d.com/) and Mixamo Fuse (www.mixamo.com/fuse). Their movements were accomplished utilizing the physical engine and algorithms that are incorporated in Unity3D; some scripts were used to tune movements and behaviors (e.g. character following another character; donkey following a man).

Virtual models that reproduce ancient civilizations anchored in credible historical information are being used widely. The virtual recreation of historical places that no longer exist or that are considerably damaged is usually based on 3D modeling tools or procedural modeling (Muller *et al.* 2006; Dylla *et al.* 2010; Saldaña and Johanson 2013). Some authors include virtual humans in the environment; see Anderson *et al.* (2010) for a review about serious games for cultural heritage.

Our approach explores the interoperability of 3D software tools to combine the models of the scenario with animated virtual characters to obtain an interactive application. For the time being, the virtual characters in the scenario have simple intelligent behaviors that will be improved by algorithms inspired in Artificial Life.

2. Creating the Virtual Scenario

The creation of the virtual scenario comprised of terrain generation, modeling the buildings (houses, the watchtower and the defensive wall) and their integration into the same environment. Figure 1 highlights the main components used in the pipeline. We applied the geographic processing capabilities of the ArcGIS platform (desktop.arcgis.com) to import the model of the terrain and 3D modeling tools to draw some of the buildings. These models were then exported to CityEngine, where more houses were added by procedural modeling. Afterwards, the complete virtual environment was exported in the format .fbx, to Unity, where the virtual characters were integrated.

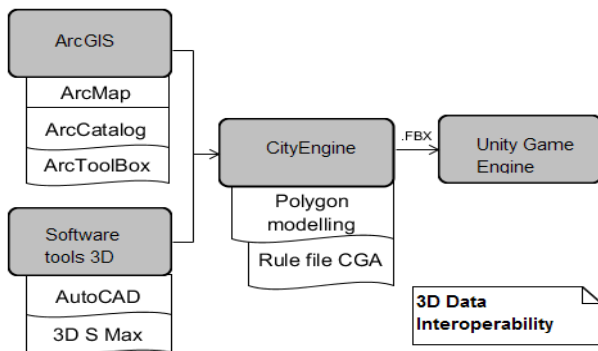


Figure 1: Pipeline overview

2.1. Terrain Generation

To obtain the model of the terrain, several steps were carried out. OpenTopography (opentopo.sdsc.edu/datasets) offers raster datasets for digital terrain models (DTM). The data on the area of Mértola was selected and captured as an image in GeoTIFF format. This file was imported to the ArcGIS platform and processed with ArcMap, ArcCatalog and ArcToolBox tools.

With ArcCatalog, we accessed the WebFeatureService (mapas.dgterritorio.pt) to get spatial data. The shapefile

used in this step was also used to export the coordinates of the region of interest in KML format in order to capture an image on Google Earth to apply as a texture to the terrain. With ArcMap, this image was georeferenced and combined with the above GeoTIFF image. Finally, the ArcToolbox generated the digital elevation model of the terrain (height map). The height map and the image with the texture were imported into CityEngine. An additional step was required to import the terrain model into Unity, as explained later.

2.2. Modeling the Buildings

To model the buildings, we used two approaches: 3D modeling tools to draw buildings for which there was more information available and procedural modeling to create similar models for general areas of the village.

AutoCAD was used to build models with different levels of detail, which were later exported to 3ds Max to be textured. These models were imported into CityEngine, in .obj format with the texture in an .mtl file. Actually, CityEngine imports data in several formats to provide interoperability with tools that work with 3D models.

Models can be imported into CityEngine as static models or as shapes. Static models can also be converted into shapes (Fig. 2), however they cannot be modified, neither their individual vertices nor their textures. They can only be scaled and positioned by translations or rotations, while shapes can be modified by grammar operations described in CGA files.

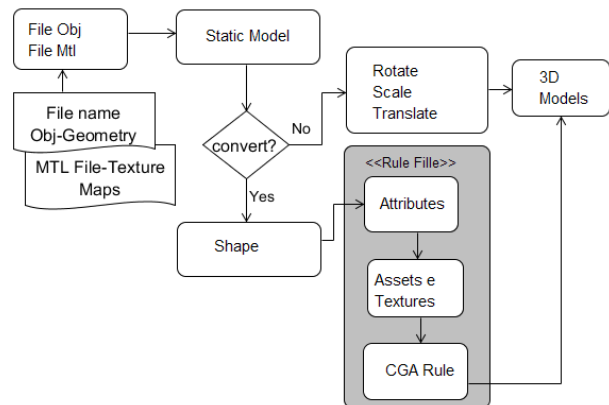


Figure 2: The pipeline inside CityEngine.

2.3. Combining the 3D elements

The virtual urban environment integrates several layers: for the terrain, the static models (comprising objects imported from 3D modeling tools) and the shapes. The shape layers include the models generated by applying grammar rules. These include buildings and streets generation, as well as landscape creation with tree models from a vegetation library (www.arcgis.com/apps/CEWebViewer/viewer.html?3dWebScene=6015f7d48dff4b3084de76bcf22c5bca). We used the layer alignment feature to combine all the layers: “align graphic to terrain”, “align static model” and “align terrain to shape”.

The complete virtual 3D CityEngine model (Fig. 3) was exported to Unity using the .fbx format. To ensure navigation in real-time, it is necessary to have a lightweight model. Considering this, some elements, like

the houses, were modeled with different levels of detail, and set in distinct layers before being exported to the .fbx file. This way, the Unity game engine can reproduce models that are away from the camera with less polygons while the ones that are closer are displayed with more polygons to attain a more realistic effect.



Figure 3: Virtual urban environment in CityEngine.

3. Inserting the Animated Characters

For the time being, three models were included in the Unity scenario: a man, a child and a donkey. The first one, together with a skeleton and a set of basic animations, was obtained from Daz 3D Studio that supports several export formats compatible with Unity. The animations were obtained from Mixamo Fuse (e.g., walk, run, sit on a bench, climb stairs). The model of the donkey, including a skeleton and a set of animations, was obtained from Arteria3D (arteria3d.myshopify.com).

To ensure that the movements of the characters are credible, two factors are important: the quality of the animations and the reproduction of the laws of physics, a responsibility of the physics engine included in Unity. The physical engine uses colliders to deal with physical collisions. So, to guarantee that characters move on the ground and do not exhibit unnatural movements, like trespassing walls, it is mandatory to define the colliders for every mesh corresponding to elements in the scenario that may be in the way of a moving character, such as other characters, houses, walls or trees.

Also the terrain needs a collider to ensure the virtual characters will move on top of it, preventing the gravity force (implemented by the physics engine) from making them fall vertically and disappear from the scene.

To export the terrain from CityEngine to Unity, a process analogous to the one described in Saldaña and Johanson (2013) was adopted. First, the terrain in CityEngine was saved to a .png file which was converted to the raw format using Photoshop; then the Unity option

“Terrain” was used to import this file, and so a “Terrain Collider” was associated with the ground.

The basic animations of the characters were combined with some scripts included in Unity that implement simple artificial intelligence algorithms, such as the A* algorithm for choosing the shortest path between two points and a script called “follow target” used, for instance, to make the donkey follow the man (Fig. 4).



Figure 4: Virtual environment with virtual characters in Unity.

4. Conclusions and future work

Currently the virtual model of the medieval village of Mértola is being improved whilst taking into consideration that a balance between realism and number of polygons needs to be attained, because it is necessary to obtain a lightweight model that supports interaction in real-time.

Concerning the virtual inhabitants, algorithms inspired in Artificial Life are being developed to obtain a generative model for heterogeneity and spontaneity in behavioral animation of small, medium and large gatherings of virtual characters.

Acknowledgements

This project was funded by the EU Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 655226 (fellowship to Dr. Rui Filipe Antunes). We acknowledge the support of BiolSI R&D unit (UID/MULTI/04046/2013), Portugal, the information provided by the team of Campo Arqueológico de Mértola and the students that have produced 3D models. Proofreading by Piki Productions.

References

- ANDERSON, E.F., McLOUGHLIN, L., LIAROKAPIS, F., PETERS, C., PETRIDIS, P. and FREITAS, S., 2010, Developing serious games for cultural heritage: a state-of-the-art review. *Virtual Reality*, 14.4, pp: 255-275.
- DYLLA, K., FRISCHER, B., MULLER, P., ULMER, A. and HAEGLER, S., 2010. Rome Reborn 2.0: A Case Study of Virtual City Reconstruction Using Procedural Modeling Techniques. *CAA 2009, Making History Interactive. 37th Proceedings of the CAA Conference*, March 22-26, 2009, Williamsburg, Virginia (Archaeopress: Oxford, 2010), pp. 62-66.

- MULLER, P., WONKA, P., HAEGLER, S., ULMER, A. and GOOL, L., 2006. Procedural Modeling of Buildings. *ACM Transactions On Graphics*, Jul, Vol. 25, pp. 614 – 623.
- SALDAÑA, M. and JOHANSON, C., 2013. Procedural modeling for rapid-prototyping of multiple building phases. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Feb., Vol. XL-5/W1.