

Panagiotis Parthenios^a, Anna Karagianni^b, Panagiotis Kotsarinis^c, Katerina Mania^d,

^a School of Architectural Engineering, Technical University of Crete, Chania, Greece parthenios@arch.tuc.gr, ^b School of Architectural Engineering, Technical University of Crete, Chania, Greece, anna.karagianni@gmail.com, ^c School of Electronic and Computer Engineering, Technical University of Crete, Chania, Greece, pkotsarinis@gmail.com, ^d School of Electronic and Computer Engineering, Technical University of Crete, Chania, Greece, amania@isc.tuc.gr,

Conceptualization and abstraction of information as a method for processing and communicating large amounts of data

Abstract

Conceptualization of information resulting in a lower amount of redundant information allows faster interactive visualization and manipulation of a large amount of data, especially in the form of 3D information. While digitizing cultural heritage monuments aiming at visualizing more than one in digital form, abstraction of information becomes the key solution when real-time access of the digitized information is required. Our proposed application aims to present the main archaeological monuments of Crete based on a conceptual 3D model. It allows the user to navigate around the monuments and view them at different levels of spatial and contextual detail. The proposed application allows switching between seven historical periods offering a comparative study of their evolution in time. By providing the minimum needed information at each level of abstraction, e.g. “level of detail”, we can create an online platform, light and easy to use.

The platform comprises the main, central space, where the 3D models are presented and two scrollable sidebars, one horizontal and one vertical. The horizontal one controls time, and allows the user to switch between the seven historical periods and the vertical one controls the level of detail, allowing the user to switch between more or less abstract modes, from viewing a wide area to viewing a specific monument in detail, belonging to that area. At the same time, the user has the ability to navigate in real time in the main space, walk through

around the models using pan, zoom in/out, rotate, etc. Only when the user reaches the fifth Level of Detail, that of a single monument, he/she has access to all the available related information, such as photographs, videos, maps, drawings, text, links, etc. The use of keywords allows a cross-reference function independent of the 3D models.

This paper describes the optimization process of conceptual modeling during the development of the Crete3D interactive platform addressing the inherent challenges of visualizing a large amount of 3D models through the web, interactively, fast and in real-time.

Introduction

The key idea of the Crete3D platform was to provide necessary information of the selected monuments, presented in five levels of detail (Crete, Prefecture, Region, Complex, Monument) and throughout seven different historical periods (Minoan, Classical, Hellenistic, Roman, Byzantine, Venetian, Ottoman, Modern). Crete3D does not intend to promote a strictly architecture-base limited image of Crete's past, but a dynamic understanding of its hybrid cultural identity (Parthenios et al. 2014). Unlike most monument-related visualization methods, the platform is based on the creation of conceptual models instead of providing descriptive details of the monuments. A model is usually constructed in order to discard details from the subject under study and retain only what is essential for a stated purpose (Henderson – Sellers and Gonzalez-Perez, 2010)

Data Handling

The current demo of the application consists of 15 monuments, normally dispersed across the four sub-peripheries of the Prefecture of Crete. The choice to represent the monuments during seven historical periods created the need to cluster the monuments according to their location and historical period and to define the amount of different models that had to be modeled at each Level of Detail. Certain monuments remained untouched throughout two or more subsequent periods of time. In that case, one model corresponds to more than

one historical period. Additionally, there are monuments that share the same Region or Complex and have to be modeled as a group. The final sum of models was 148.

Solid and transparent pins were used to depict the active and inactive historical periods for each monument: a solid pin over a model indicates the monument's transformation, while a transparent pin is used to show that a monument has remained inactive at the specific period of time.

The actual time it took to load the 3D models challenged the overall platform development and design, given that the average capacity of each .skp file prior to optimization reached 15mb. As a result, the code developers defined the maximum file capacity to 100kb, which meant reducing the desired file at a staggering low 0.6% of the original model. In order to achieve this massive percentage of reduction, the optimization was organized in two levels: the synthesis optimization and the geometry optimization.

Synthesis Optimization

Optimizing the synthesis of the 3d models required the identification of the primary and value entities of the monuments. Primary entities are defined as valuable entities, which, when perceived, are understood in the absence of explicit interpretive process, whereas value entities are described as value entities that can only be understood as the outcome of a valuation. (Cesar Gonzalez-Perez et al., 2006). The main scope of the first optimization process was to identify and project the primary entities of the monuments in a compact way.

Each monument, complex or region model consists of two parts: the topography (ground) and the building. The ground was designed using Google Earth terrain and converting the topography into NURBS Surface in Google Sketchup. In most cases, when the monument belongs to an urban area, a map of the area in .png format was superimposed on the topography surface.

Initially, the buildings were constructed using the Google Sketchup software and were based on architectural plans, visual measurements and historical literature. During the synthetic optimization process, morphological details were simplified and higher levels of detail were abrogated in order to

keep the information as diagrammatic as possible without eliminating important historical data of the monument. Decorative elements of the monuments were eliminated in the framework of reducing information not related to the diagrammatic aspect and the historical perception of the monument.

Data Optimization

The second step of the optimization was to remove unnecessary geometry information such as excessive triangulation, duplicate areas and merging coplanar connected surfaces. Initial tests included exporting .skp files into 3ds files and then importing them into Meshlab in order to achieve uniform optimization. This effort was not successful as the software failed to calculate the amount of triangles and reported unexpected failures. Dividing each monument into files that consisted of smaller parts resulting in the same negative result.

Another effort consisted of importing the 3ds files into 3ds Max files. The Pro-Optimize and Optimizer files were tested using the percentage of optimization that better suited each file. After testing several files, we observed that uniform optimization did not generate satisfactory results. When a high percentage of optimization (40-50%) was applied, complex geometry was successfully reduced, while simple geometry was erased. On the other hand, low percentages of optimization (1-20%) reduced by 10% the file capacity.

Geometrical complexity was used as a parameter to create a system that produces optimal results within a certain set of constraints. All monuments, complexes and regions were separated into two distinct files: the ground and the building(s). Then, the buildings that contained curved surfaces were also separated into files containing rectangular and curved parts.

Each sub-file was imported into 3ds Max Design and geometry was reduced using Pro-Optimizer and Optimize modifiers. The Optimize percentages varied according to geometry complexity – simple geometries were reduced by 20%, ground topographies by 30-40%, while complex geometries by 50%. Then the optimized 3ds files were imported in Google Sketchup and exported into .obj files in order to measure the optimization results. Optimized average file capacity was 400kb and corresponded to 2,6% of the initial files, however, the initial optimization did not achieve that requirement. Following to this, two Google Sketchup plug-ins were applied: CleanUp3 and EdgeTools. CleanUp3

has a similar function to the Windows Disk CleanUp: it purges unused items, erases duplicate faces, repairs split edges and merges connected co-planar faces. Initial and optimized files were again compared and the optimized file capacity corresponded to 0.08% of the initial file. In order to achieve maximum optimization, the code was reshaped so that the ground topography model could be loaded just once throughout all periods of time for every monument, complex or region. In that way, the average optimized file reached the 0.6% required percentage.

Common Language

During the early stages of the platform development, it was necessary to adopt a common morphology for visualizing the monuments. A shared texture list was necessary to create a common database of 7 distinct textures (maps) that produced 15 different textures. Texture varied depending on the criteria of creating a uniform visual appearance of the platform while preserving the materiality of each monument sub-parts.

Transcoding and Compatibility Constraints

Before a 3d model can be run through a code, it must be converted from a vector-based format to a computer-readable format. In that sense, 3d models were converted from .skp files to a code compatible format. During the first demo, .skp files were converted to .dae format. During the second and third demo, due to the amount of models, additional formats were examined. Finally, the fact that .obj files were of low capacity drove the code transformation.

Although the choice to export in .obj format drastically decreased the models capacity,

SketchUp will export one face entity as one polygon in an .obj file. This ratio of faces to polygons creates some problems for certain .obj importers (polygons might appear missing or reversed)[..].

The .obj output supports a flat set membership hierarchy meaning that the format identifies which objects belong to any set. This output does

not support a tree hierarchy because it cannot identify if one particular set is actually a component of another set. This limitation is an inherent characteristic of the .obj format¹

Another technical issue of the transcoding process was the incompatibility of the png transparency with the code visualization. The WebGL platform only recognizes png mode 0 or 1, which means that either the image will be visualized as 100% transparent or 0%, despite the intermediary amounts that the modeller has assigned while creating the 3D geometry. As mentioned before, in most topography model cases, a map was superimposed on the NURBS surface to reflect the urban site context of the monument. The map was created using a png format and applied as a Sketchup material on the surface. As a result, the transparency of the png picture could not be visualized, so the modellers had to cut the triangles that corresponded to the transparent surface of the png map using Sketchup tools.

The last issue envisaged during the vector-to-code conversion was the ‘broken texture’ result. Despite the fact that the textures were applied to solids and surfaces of every group of object in the skp file, the visual appearance of some objects was not acceptable, due to the unexpected effect of the geometry after the texture application, especially on textures that had some level of transparency. The surfaces appeared triangulated and some parts of the geometry preserved a higher level of opacity. After examining a few alternatives of texturing aiming to get optimum result, the visual appearance improved by revealing two simple paths of texturing: the texture application to both sides of a given surface and the reduction of transparency to the level that the object would not be read as opaque. Although both techniques ameliorated the visual effect, the problem of transparency triangulation still remained at a minimum level.

Further Optimization Techniques

Optimization is necessary to accomplish online projects that include large geometry data. Further techniques would include the subdivision of each

¹ Obj export known issues and limitations, <http://help.sketchup.com/en/article/114389>

monument into parts that have been transformed over time and parts that remain untouched. This method would enable to load two models in .obj format separately and reload just the changing part at the corresponding historical period of time. Another interesting method that could be examined is the creation of a common texture database at the code level. Until now, the common texture database has served as a tool for visual contingency, whereas it could be loaded as a database just once and save at least 20 Mb of mtl and jpeg files that are currently loaded for every model separately.

Bibliography

Books:

Reas, Casey; McWilliams, Chandler. *Form & Code in Design, Art and Architecture*. New York: Princeton Architectural Press. 2010.

Sakamoto, Tomoko, Ferre, Albert, Kubo, Robert. *From Control and Design*. Barcelona: ACTAR. 2006.

Hasegawa, Toru; Collins, Mark. *Proxy Log*, New York: Lulu Press. 2009.

Reas, Casey; Fry, Ben. *Processing: A Programming Handbook for Visual Designers and Artists*. Cambridge, Massachusetts: The MIT Press. 2007.

Paper in journal:

P.Parthenios; K.Mania; A. Oikonomou; F.Mallouchou; L.Ragia; N.Patsavos; A.Christaki; P.Kotsarinis; M.Dimitriou. . “Using WEBGL to design an interactive 3D Platform for the monuments of Crete”. Proceedings of the 1st CAA GR Conference, Rethymno, 2014.

Gonzalez-Perez, César; Parceró-Oubiña, César. “A conceptual model for cultural definition and motivation”. Proceedings of the 39th Conference in Computer Applications and Quantitative Methods in Archaeology, Beijing, China. 2011.

Website:

<http://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/3DSMax/files/GUID-109C880F-181B-4067-91F0-B4EF79639FB3-htm.html>

<http://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/3DSMax/files/GUID-4D61E710-9D95-4469-B05C-8C8A126D59CC-htm.html>

Figures

	ΜΝΗΜΕΙΟ	νομός	ιστορικές περίοδοι
1	Μινωική Κυδωνία	Χανιά	MIN
3	Αιπέρα	Χανιά	APX, ΡΩΜ, ΕΝΕ, ΟΘΩ, ΝΕΩ
7	Γιαλί τζαμί	Χανιά	ΟΘΩ, ΝΕΩ
8	Σπίτι Βενιζέλου	Χανιά	ΝΕΩ
9	Αγορά	Χανιά	ΝΕΩ
11	Αρκάδι	Ρέθυμνο	BYZ, ΕΝΕ
23	Βίλλα Ετιάς	Λασιθί	ΕΝΕ
2	Πολυρρήνια	Χανιά	APX, ΡΩΜ
4	Ροτόντα	Χανιά	BYZ, ΕΝΕ
5	Μεγας Κουλες	Ρέθυμνο	ΕΝΕ, ΝΕΩ
6	Φιρκάς	Χανιά	ΕΝΕ, ΟΘΩ, ΝΕΩ
10	Φιρκάς	Ρέθυμνο	ΕΝΕ, ΝΕΩ
12	Φορτέτζα	Ρέθυμνο	ΕΝΕ, ΟΘΩ, ΝΕΩ
13	Νερατζέ Τζαμί	Ρέθυμνο	ΟΘΩ
14	Κνωσός	Ηράκλειο	MIN
15	Φαιστός	Ηράκλειο	MIN
16	Γόρτυνα	Ηράκλειο	APX, ΡΩΜ, BYZ
17	Μέγας Κούλες	Ηράκλειο	ΕΝΕ, ΟΘΩ, ΝΕΩ
18	Λότζα	Ηράκλειο	ΕΝΕ
18	Άγιος Τίτος	Ηράκλειο	ΟΘΩ
19	Βικελαία Βιβλιοθήκη	Ηράκλειο	ΝΕΩ
20	Αρχαιολογικό Μουσείο	Ηράκλειο	ΝΕΩ
21	Ζάκρος	Λασιθί	MIN
22	Μονή Τοπλού	Λασιθί	BYZ, ΕΝΕ
24	Οικία Χρονάκη	Ηράκλειο	ΟΘΩ

figure 01. List of models



Figure 02. 'Broken texture' effect

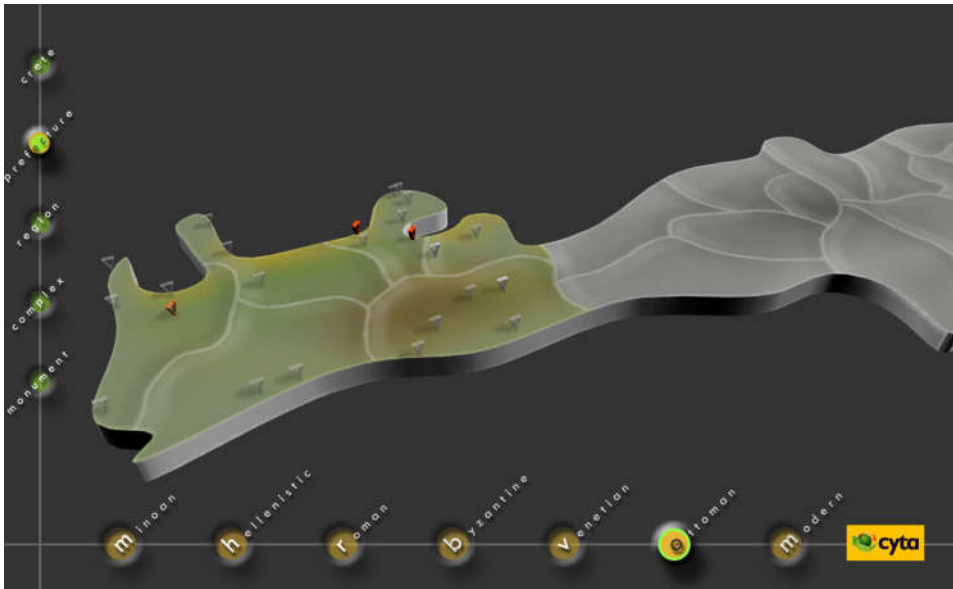


Figure 03. Interface – Prefecture Level



Figure 04. Interface - Region Level