Web Visualization of Complex Reality-Based 3D Models with NUBES

B. Jiménez Fernández-Palacios, F. Remondino
3D Optical Metrology (3DOM) unit,
Bruno Kessler Foundation (FBK)
Trento, Italy
(bjfernandez, remondino)@fbk.eu

C. Stefani, J. Lombardo, L. De Luca
CNRS-Map Gamsau
Marseille, France
(chiara.stefani, julie.lombardo, livio.deluca)@map.archi.fr

***Abstract*—This paper discusses the fundamental issues of the real-time web-based visualization of complex reality-based 3D models. As web platform, we use NUBES, an innovative and powerful tool for sharing and analyzing reality-based 3D models online. A new automatic procedure for the setting-up and the uploading of 3D complex scenes into NUBES is presented, including the optimization of the geometric and radiometric information for web visualization.**

***Keywords*—online platform; remote access; real-time interaction; complex 3D models.**

I. INTRODUCTION AND RELATED WORKS

Due to the technological developments of web libraries, API, plug-in applications, game engine, as well as languages and technologies [1], it is nowadays possible to visualize complex 3D models remotely and online. The user sometimes does not need to use or install specific hardware and software during the real-time interaction of a 3D model [2]. In other case, 3D models are linked to databases, for studies, analyses (queries) and divulgation purposes [3]. However, when high definition has to be conserved and visualized, the real-time interaction and online navigation of virtual 3D models can be also problematic, due to their huge size (millions of polygons and large texture files).

Different rendering approaches have been already developed using remote rendering platforms as an image-based LOD method [4], or point-based representation [5] or mesh-based method [6] as well as in a mixed approach of image- and mesh-based rendering [7].

In this paper, after a short overview of the NUBES platform [8], we present (i) an automatic pipeline for converting and uploading complex and hierarchically structured 3D scenes into NUBES and (ii) a facultative (but suggested) methodology to optimize geometry and texture of large 3D models before import them into NUBES.

II. NUBES PLATFORM

NUBES (http://) is a research project [8] which relies on the definition of an information system at architectural scale. It is an online integrated platform for documenting, describing, analyzing and sharing reality-based 3D models of heritage objects (monuments, buildings, objects, etc.). The NUBES platform constitutes today a suitable solution to virtually visit distant places, help education and promotion of historical sites and finally organize, divulgate and share the collected information. However, even if NUBES constitutes a container of different kinds of data for different purposes, the fundamental problems of visualizing online large reality-based 3D models remains open.

III. OPTIMIZATION NEEDS

Photogrammetry and laser scanning produce not only detail and accurate 3D models but also large data. Reality-based 3D models have often millions of polygons and file sizes even larger than 1GB. They are frequently sub-sampled and reduced in geometry and texture, because often they cannot be visualized on a personal computer. Very few solutions permit users to handle and manage on the web 3D models for architectural and archaeological purposes. For these reasons and in order to work without downloading 3D models locally, online resources are being exploited and nowadays it is possible to visualize remotely and online complex 3D models.

The visualization is often constrained by the graphics card performances, which denies a real-time visualization of 3D models, and by the network connection, which leads to long uploading and loading time. The solution is to find a compromise between size, accuracy and visualization of 3D models in order to manage detailed 3D scenes with a fluent real-time interaction.

IV. WEB VISUALIZATION OF COMPLEX 3D SCENES INTO NUBES

In order to show the optimization, import and visualization procedures associated to the NUBES platform, the reality-based 3D model of the Bettini etruscan tomb is presented. The geometric 3D model was produced using Time-Of-Flight (TOF) laser scanner data, with a sub-centimetric resolution. High resolution digital images were also acquired and used as textures, in order to create a photo-realistic 3D model. The 3D model of the Bettini tomb has ca 1.6 million points, 3.2 million polygons and the total file size is **274 MB** (DAE OpenCollada file size is **195 MB** and the compress file (ZIP) size, with high resolution texture, is **79 MB**). Similar o l x d " vy" 3D models, if directly imported in an online visualization platform (like NUBES), would require long loading time and, generally, cause problems to the graphic card of the computer. Moreover, in case of hierarchical scenes composed of multiple 3D entities, an automated procedure is mandatory in order to transfer all the models to the server and allow a web / remote access.

For these reasons, the following sections present an automated pipeline to import 3D models in NUBES (section A) and the pre-optimization of the models geometry (section B) and texture (section C).

A. Automated conversion and import pipeline

The conversion and import pipeline is based on a collection of interconnected processes and steps concerning 3D geometry, textures and hierarchical structures by the integrated use of 3 main languages: PHP, JavaScript and VSL (Virtools Scripting Language). The conversion process is composed by seven different parts (Fig. 1): by means of the main NUBES web interface (step 1), the user uploads firstly a DAE Collada file containing a 3D scene, and secondly a compressed (ZIP) folder containing the adopted textures. Specific options allow defining the way in which textures and materials will be processed (texture aspect ratio, material emissivity, etc.). On the client side, when the upload is finished, a PHP script generates a parameter file (step 2) containing all the chosen parameters and copies it (together with the Collada and the texture ZIP files) into a FTP server (step 3). Once the download on the FTP server is completed (step 4), the PHP script starts a "waiting process". At this moment, on the server side, a VSL script detects the presence of new files and starts the conversion process (step 5). Firstly, the VSL script decomposes the complex scene into a collection of independent 3D files compressed in NMO Virtools format. This is possible by reading the hierarchical structure of the 3D entities described in the XML-based Collada file, and by applying specific geometry and texture compressions. Geometries are simplified by applying homogenous sampling to 3D meshes, while textures are simplified by reducing the size and compressing the images. This step produces three independent versions of the 3D entity (High, Mid, Low). Secondly, the VSL script generates two tables (3D morphological entities and 3D representations) and creates a journal text file. When the conversion process is finished (step 6), the VSL script (server side) sends a message to the PHP script (client side), still waiting for the process completion. If the message is received, the transfer of the obtained 3D file collection into the NUBES database starts (step 7).

The described procedure is mandatory to simplify, during the import phase, reality-based 3D models and allow a faster display and interaction on the web. Fully resoluted 3D geometries and texture could be ingested directly in NUBES using the aforementioned pipeline, but, in order to reduce the loading and display time, still keeping the geometric and visual appearance, a pre-processing optimization step is highly suggested. Some optimization tests have been performed in 3ds Max in order to obtain a reduced 3D model with still reasonable geometric and visual appearance.

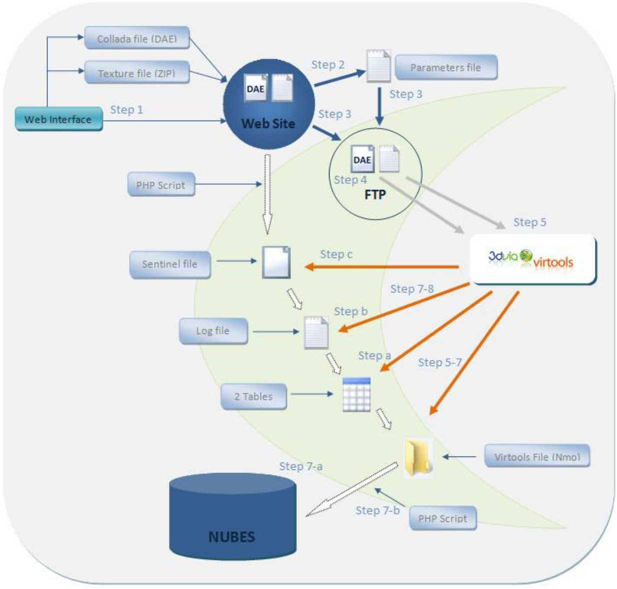


Fig. 1. Automatic conversion pipeline.

B. Pre-optimization of the geometry

Optimized objects can reduce scene memory requirements, reducing object faces. The geometry of the Bettini 3D model has been optimized (using the “P oO t z ” tool 3d M x) in differents levels of resolution (TABLE 1).

BETTINI TOMB TEST								
3D Model (3ds MAX and PS)							NUBES	
Resolution	Geometry optimization	Points	Faces	3D Object size (DAE)	Texture compression PS (JPG)	Texture size (ZIP)	Upload in Nubes (DAE + Texure.ZIP)	Final 3D object size (NMO)
High	100%	1.587.413	3.151.141	195 MB	12 Quality	78,8 MB	273,8 MB	52,5 MB
High / Medium	50%	805.548	1.587.407	118 MB	10 Quality	30,2 MB	148,2 MB	37,7 MB
Medium	25%	414.616	805.543	81 MB	8 Quality	18,8 MB	99,8 MB	17,6 MB
Medium / Low	10%	180.054	336.419	60 MB	6 Quality	14,4 MB	74,4 MB	6,63 MB
Low	5%	101870	180051	53 MB	4 Quality	10,4 MB	63,4 MB	3,58 MB
Extra-Low	1%	39.321	54.953	47 MB	2 Quality	6,26 MB	53,26 MB	1,19 MB

TABLE 1. Relationship between weight and resolution of the Bettini 3D model, from 3ds Max to NUBES.

The two extreme cases are shown in Fig. 2. A high resolution 3D model (100% geometric resolution) and an extra-low resolution 3D model (1% geometric resolution). The high resolution 3D model (Fig. 2.a) still looks good and geometrically well defined because of the high density of the polygons and the high resolution of the images that are used to texture the model.

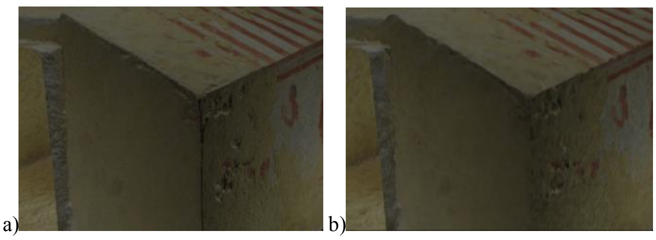
a)  b)

Fig. 2. Bettini tomb: (a) high resolution and (b) extra-low resolution.

The optimization process (in 3ds Max) allows to reduce up to 5% of the geometric resolution, without observing a great damage to the model corners and edges (Fig. 4). However, when the 3D model' geometry is reduced to the limit (Fig. 2.b), it looks sharp and edgier, the polygons are too large and details disappear.

C. Pre-optimization of the visual appearance

Reality-based 3D models are normally textured with high resolution images which also affect negatively the upload procedure and consequently the upload time. Therefore, as shown in TABLE 1, the textures (JPG format) have been also reduced in order to obtain a "lighter" model. Beside this, with the aim of maintaining the quality of the digital 3D representation in NUBES, a normal map approach is also used. Normal maps permit to reproduce high resolution geometry detail when it's mapped onto a low resolution 3D model. Each pixel of the normal map stores a normal, a vector that describes the surface slope of the original high resolution 3D model at that point. 3ds Max lets the user choose which coordinate to use on the normals among 4 different methods: (i) the tangent method which project at a tangent to the target object's surface, (ii) the local XYZ method which projects using the object's local coordinates, (iii) the screen method which projects using screen coordinates and (iv) the world method which projects using world coordinates. When a normal map is applied to a low resolution 3D model, the texture pixels control the direction of each pixel on the low resolution 3D model creating the illusion of having a higher surface detail or a better curvature. However, the silhouette / geometry of the model does not change. In order to obtain the best results, the world method has been used as it seemed the best for objects that do not move or deform. As shown in Fig. 3, both the high resolution and the extra-low resolution of the 3D model, textured using normal maps, have provided practically equal visual results.

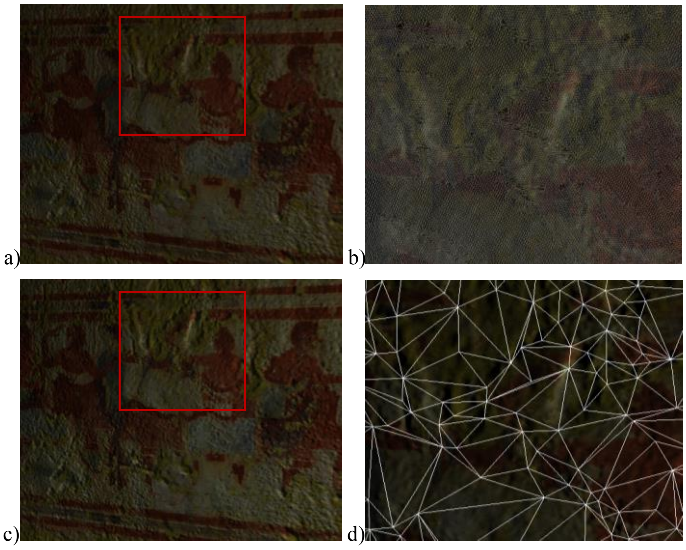


Fig. 3. Bettini tomb: high resolution model (a) and wireframe detail (b); extra-low resolution model textured with normal map: wireframe (c) and wireframe detail (d).

V. RESULTS

The results achieved with the pre-optimization step and the successive compression and import into NUBES are hereafter presented. Three different geometric optimizations have been selected coupled with 12- and 8-quality compression of the texture (TABLE 2). The size of the original high resolution 3D model is **274 MB** while the size of extra-low resolution 3D model is **81 MB** (DAE OpenCollada, textures file (ZIP) and normal map file (ZIP)) i.e. a reduction of **71%** compared to the original model. Therefore, for the purpose of visualization and divulgation, the result is unbeatable. However, for the metric purpose, the silhouette of the model (Fig. 4) does not change and remains an extra-low resolution.

BETTINI TOMB TEST							
From 3D model (3ds MAX and PS)						To NUBES	Reduction
Resolution	Geometry	3D Object (DAE)	Texture PS (JPG)	Texture (ZIP) MB	Normal Map (ZIP) MB	Upload size (DAE + Texure.ZIP + Nomal Map.ZIP)	Percentage %
High	100%	195 MB	12 Quality	78,8 MB	---	273,8 MB	0%
Low	5%	53 MB	8 Quality	18,8 MB	15 MB	86,8 MB	68%
Extra-Low	1%	47 MB	8 Quality	18,8 MB	14,9 MB	80,7 MB	71%

TABLE 2. Optimization (in 3ds for the geometry and Photoshop for the texture) of the Bettini' 3D model textured with normal maps.

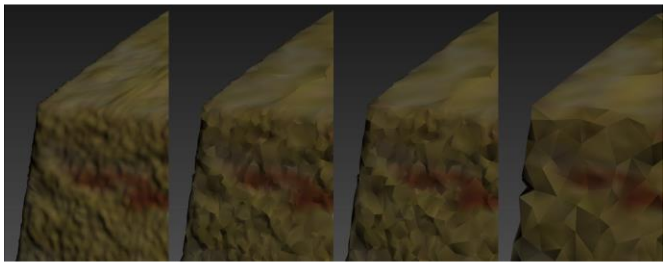


Fig. 4. Silhouette comparison: 100%, 10%, 5% and 1% geometric resolution.

High (100%), low (5%) and extra-low (1%) resolution have been compared (Fig. 5). Yellow color shows high resolution while blue color shows extra-low or low resolution of a detail of the banquet scene in the Bettini tomb. As shown in Fig. 5.a, comparison between high and extra-low resolution, it is possible to observe the lack of geometrical information due to the abundance of large gaps (in blue color).

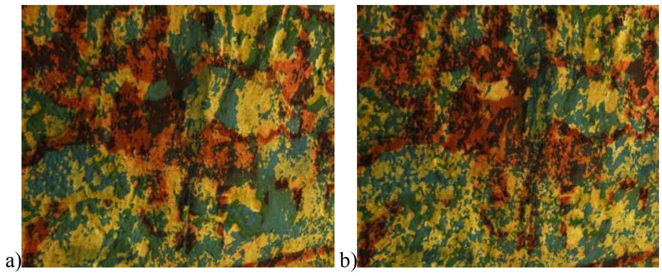


Fig. 5. Comparison between high resolution (yellow color) and extra-low resolution (blue color) (a) and high with low resolution (blue color) (b).

Therefore the low resolution model of the tomb (5% geometric resolution) has been chosen to be textured with the normal map method, in order to find the best compromise (Fig. 5.b). The size of the 3D model is **53 MB**. As shown in Fig.4, the silhouette of the low resolution 3D model (5% geometric resolution) doesn't suffer great change compared to the original, and maintains all the details of the main geometry, while in the case of extra-low resolution (1% geometric resolution), details are lost.

The final size of the produced 3D model, if compared to the extra-low resolution one (1% geometric resolution) is negligible (TABLE 2). The final 3D model size (DAE OpenCollada, textures file (ZIP) and normal maps file (ZIP)) is **87 MB**. This is a **68 %** reduction compared to the original model, only **3%** less than extra-low resolution 3D model. In this way, the best solution/relation between weight, accuracy and quality of complex 3D is guaranteed.

The final 3D model of the Bettini tomb visualized in NUBES is shown in Fig. 6. The optimized 3D model takes less than a minute for loading using a normal wifi connection and then it can be easily moved and manipulated.

a) b) c)

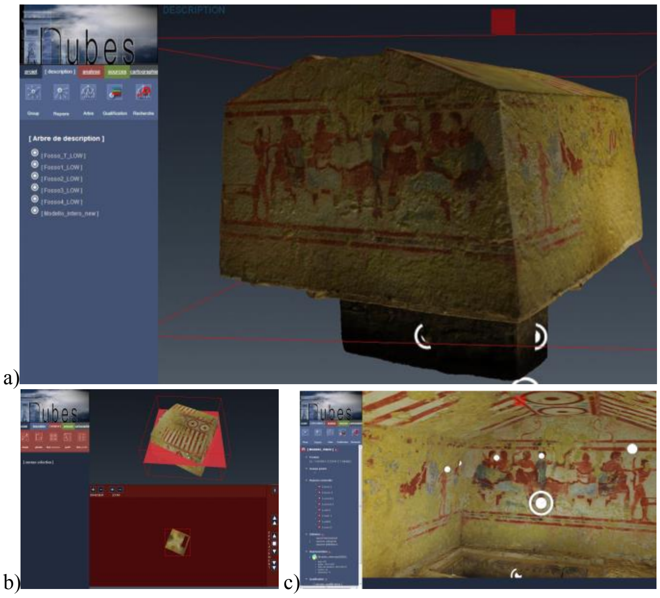


Fig. 6. The final optimized 3D model of the Bettini tomb visualized inside the NUBES platform (a), with the creation of sections (b) and the semantic description using hot-spots (c).

VI. Conclusions

In this paper, some web visualization issues have been investigated proposing the use of the NUBES platform [8] to analyze, visualize and share online complex 3D models.

A new automatic procedure recently implemented to compress, convert and import complex 3D models in NUBES was described. This procedure, although not mandatory, is normally coupled with some pre-optimization steps in order to reduce the geometry and texture of the 3D models. For the texture optimization, beside classical compression, a normal map approach is also applied, which implies a slighter increase of the final 3D model size, but absolutely irrelevant (see TABLE 2).

All in all, the size of the 3D models is drastically reduced compared to the original model still preserving t o j t' appearance, in terms of geometric and radiometric information. The presented methodology is able to reduce a complex 3D model up to 70 % in geometry and 50% in texture and, at the same time, guarantee an optimal solution for visualizing and managing complex 3D models online.However, when it comes to huge datasets (millions of polygons and almost GB of texture files), we cannot establish always a standard pipeline regarding the optimization procedure. Each case is different and the optimization depends on the density of the original mesh, the desired level of detail as well as the object dimensions, shape and complexity.

NUBES is in anycase a powerful platforms for remote access and sharing of large and complex 3D models online, allowing not only simple visualization but also query to external databases or creation of geometric sections (Fig. 6b,c).

Acknowledgment

The work was partly supported by the 3D-ICONS project funded under t E o Co o ' ICT Pol y S o t Programme. The authors would like to thank Sébastien Trousse for its implementation help related to the uploading procedure of complex 3D models in NUBES.

References

[1] A M M d d F R o d o, "A v of reality-based 3D model generation, segmentation and web-based visualization t od ," I t Jo l o H t g t D g t l E , Vol 1(1), pp. 103-124, 2012.

[2] D. Abate, S. Migliori, S., Pierattini, B. Jimenez Fernandez-Palacios, A. Rizzi, and F. Remondi o, "R ot d g d v l z t o o l g t x t d 3D od l ," P o 18t IEEE Intern. Conference on Virtual Systems and MultiMedia, G. Guidi / L. Addison (Eds.), pp. 399-404. Milan, Italy, 2012.

[3] G. Agugiaro, R. Remondino, G. Girardi, J. von Schwerin, H. Richards-Rissetto, and R. De Amicis, "QUERYARCH3D: Querying and visualizing 3D models of a Maya archaeological site in a web- d t ," G o o t FCE CTU Journal, vol. 6, pp. 10-17, Prague, Czech Republic, 2011.

[4] http://graphics.stanford.edu/software/scanview/, Last access date 15/09/2013.

[5] S R k z d M L voy, "St g QS l t: v o t o k d v l z t o o l g d od l ," P o d g of the Interactive 3D Graphics (I3D) Symposium, 2001.

[6] P. Cignoni, F. Gavonelli, E. Gobetti, F. Marton, F. Ponchio, and R S o g o, "Ad t v T t P zzl – efficent out of core o t t o d v l z t o o g g t olygo l od l ," Proc. ACM SIGGRAPH, Vol. 23(3), pp. 796-803, 2004.

[7] Y Ok oto, T O , d K Ik , "I g -based network d g o l g o lo d o t g," I t t o l Journal of Computer Vision, Vol. (94), pp. 12-22, 2011.

[8] L. De Luca, C. Busayarat C. Stefani P. Veron, and M. Flo z o, "A t -based plat-form for the digital analysis o t t t l t g ," Co t & G Vol 35, Issue 2, Elsevier, pp 227-241, 2011.