Real-time Visualization of the Forum of Pompei

Alice Pignatel,[1] Fausto Brevi,[1] and Sebastiano Ercoli[1]

[1] INDACO, Politecnico di Milano. Italy.

Abstract

This paper describes the process to create a single realistic 3D digital model of a wide archeological site, to be used for virtual reality application. The incoming data are texturized models coming from an accurate 3D survey of the entire area: the ground, buildings and finds were acquired with active and passive sensors and converted into polygonal models. Afterwards, the models were assembled in a single virtual scene that reproduces the Forum as it would actually appear to a tourist. The scene can be visualized by means of stereoscopic devices to increase the feeling of immersion.
During the data conversion process we had to face three main kind of problems: the database optimization, the correction of digital objects position into the scene and the realism of the displayed model. The real time rendered model of the Forum was displayed on a large screen with stereoscopic technology, as a support for archeological studies.

Key words: *Virtual reality, 3D visualization, real time rendering*

1 Introduction

This work is part of a major project of multi-resolution survey of the archeological site of Pompei[1].
This part of the process started from the polygonal models obtained during the 3D survey and used them to build a digital scene to be displayed with real-time rendering techniques. The interactive realistic visualization of Forum area was used for evaluation purposes.
For the visualization step, we used a commercial real-time rendering software mostly employed during design review in the industrial field. This choice is due to its capability of managing large data set describing complex and very detailed geometries.

2 Input data

The input data used in this project step are texturized polygonal 3D models. All the models come from an accurate survey of the entire Forum area and they were previously treated to obtain coherent and error-free polygonal meshes, with the right image textures already applied to the surfaces.
All the geometric models of the Forum components are VRML files, containing objects with variable dimensions and complexity, from few centimeters up to hundreds of meters. The size of the VRML files varies from 3KB to about 14MB, according to the geometrical complexity and the number of polygons describing the object.
The images used as textures are square files in JPEG format. The dimension of these images varies from 512x512 pixels up to 4096x4096 pixels. Every VRML file refers to one or more JPEG image, according to the object complexity, with a maximum of five associated textures.
The 3D models of the Forum components can be divided in three different typologies, according to their dimension and geometrical complexity.

1 Guidi, G., Remondino, F., Russo, M., Voltolini, F., Rizzi, A., Menna, F., Masci, M.E. and Benedetti, B. (2008), "A multi-resolution methodology for archeological survey: the Pompeii forum", *Conference in Virtual Systems and Multimedia Dedicated to Digital Heritage (VSMM).*

The biggest model is the ground model. It covers the whole Forum area, and it measures about 150 x 80 meters. It contains only the floor objects, such as the remnants of roads and the grass now covering the ruins, and it was used as a reference for the correct placing of all the other objects on the scene. It is described by a single polygon mesh, with a large number of triangles and a middle level of geometric complexity. It has a single large image file applied to the polygonal surface.

A second typology of model is represented by the buildings facing the Forum area. This type of models consists of eight structures distributed all along the Forum perimeter. Their size varies from about 5 up to 20 meters and their geometric complexity is very high. In fact, the average polygon number of every file is about 13,000 triangles, even if the level of detail varies a lot from a building to another one. The smallest buildings need only one image texture, while some larger and more complex structure requires up to five image files. Also the image dimension is very different from one building to another: in fact, JPEG files range from 512x512 pixel up to 4096x4096 pixel.

Finally, the third data typology is represented by the findings. These are 377 small objects spreading all over the Forum area, which dimension ranges from about 0.5 centimeters up to 5 meters. The overall complexity of these objects is very small and the number of polygons of each file ranges from 39 up to 1,854. Moreover, every VRML model refers only to a single image file which size ranges from 512x512 pixel up to 2048x2048 pixel.

The following table summarizes the typologies of incoming data and their main characteristics.

Model Type	Ground	Building	Finding
Geometry			
File size	14 MB	128 KB ÷ 8.3 MB	3 KB ÷ 1.7 MB
Object dimension	150 x 80 m	5÷20 m	0.5÷5 m
Polygon number	216,175	143÷116,557 (av 12,856)	39÷1,854 (av 514)
Level of detail	middle	high	low
Texture			
File size	2.5 MB	128 KB ÷ 3.2 MB	43 KB ÷ 1.7 MB
Image dimension	4096x4096 px	512x512 ÷ 4096x4096 px	512x512 ÷ 2048x2048 px
Texture per model	1	1÷5	1

Figure 1. Typologies of input data.

3 Process overview

The first step of the process is to import geometry into the real-time rendering software. Some of the 3D models were imported with manual procedures, while others were imported automatically by the software. After every geometry import step, the incoming geometries were accurately checked in order to detect and fix any possible problem that would affect the 3D visualization.

It is very important to control the visual quality of digital objects on a virtual scene, especially when dealing with polygonal models coming from 3D surveys. Usually, a real-time rendering software is designed to import NURBS models from CAD or CAID systems. These files contain the mathematical description of digital objects, so that they can be displayed at different level of detail: the real-time rendering software can obtain, from the same geometric description, an infinite number of polygonal model, used to render the scene, approximating the original geometry with different accuracy level. Thus, the visual quality can be easily tuned to get the better compromise between high visual quality and system performances. Moreover, errors on the geometry of NURBS models can be easily fixed, recalculating the object topology contained in the mathematic description. On the contrary, polygonal models contain only a list of points describing the objects surface and cannot be fixed nor recalculated at different level of detail without using an external editing software. Anyway, polygonal model file formats are supported by the real-time rendering software, and they have the advantage of skipping the initial step of polygon calculation that is necessary when dealing with NURBS models.

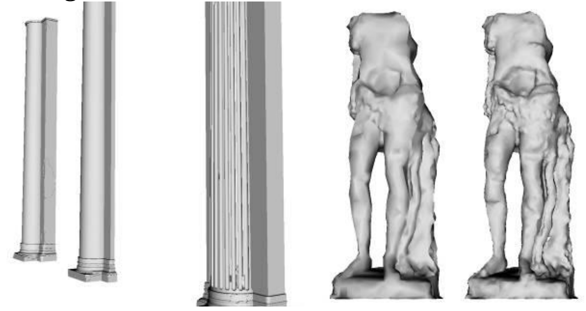

Figure 2. Different level of detail (LOD) on NURBS surfaces (left) and on polygonal surfaces (right).

In addition, polygonal models are able to reproduce exactly the shape of existing objects, especially in the archeological field, while the construction of NURBS models of the whole Forum area will be more time expensive and less accurate.

The large database dimension was a critical point to check. Thus, after importing all the items on the virtual scene, we checked the system performances, to understand if the database could be managed by the workstation or not. Memory usage for both CPU and GPU was tested, in order to understand the critical point and to optimize the database consequently.

The next step after the database optimization, is the scene lighting. To improve the realism of virtual environments, it is necessary to add lights and shadows that would prevent to obtain a "flat" effect for the surfaces on the scene. For digital models reproducing real places it is necessary to study the real light conditions, in order to obtain a consistent reproduction of the existing site.

Afterwards, the visualization system can be prepared in order to obtain the best immersive experience, setting up the correct parameters for stereoscopic visualization and choosing the appropriate navigation paradigm to move the digital camera all across the virtual scene.

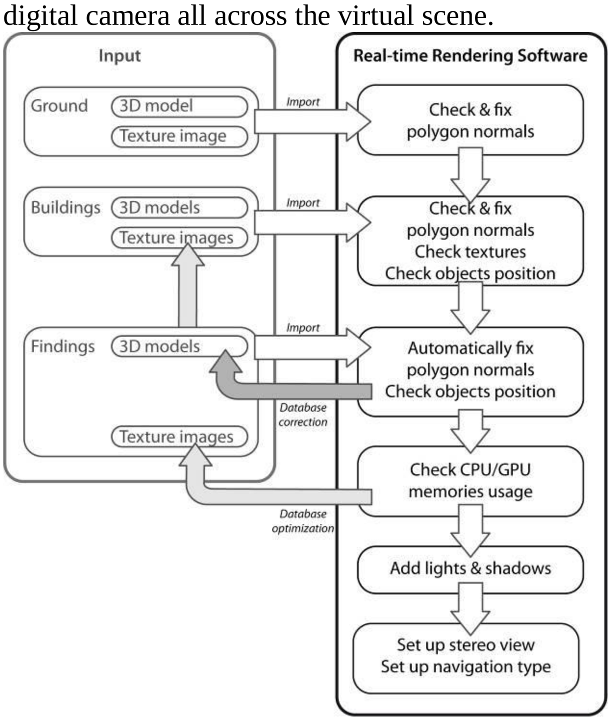

Figure 3. Process layout.

4 Placing the objects into the scene

To place the objects into the virtual scene we imported the geometries according to their hierarchical classification. Thus, to begin the process, we manually imported the ground model into the real-time rendering software. In fact, this model was always used as a reference for all the other elements of the scene, so it was necessary to check this part first. After the model was imported into the scene we checked that the polygonal mesh was correctly displayed and that no triangles were missing. We also checked that the image texture associated with the 3D model was correctly displayed and applied onto the surfaces.

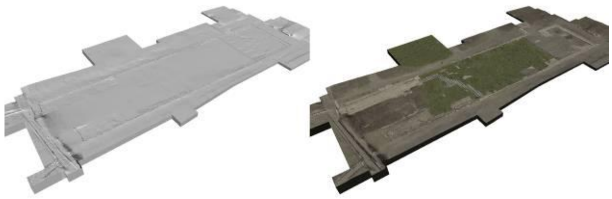

Figure 4. The ground model.

Afterwards, we imported the buildings into the scene and we decided to import and check manually every single file. This choice is mainly due to the big impact that these models have on the whole scene. In fact, the large structures surrounding the Forum area can be seen even from a far point of view, so it is important that everything looks right. Moreover, this type of model is highly detailed and it is necessary that every part on the building surface is correctly displayed when focusing on the single element. Finally, it was possible to manually manage this part of the process, thanks to the small amount of separate files to process.

During the building import process we had to face two different problems. A first problem occurred only for two models containing parts of the remnant walls on the north-west side of the Forum area. The polygonal mesh showed some missing triangle. This problem is due to differences in the VRML decoding procedure between the real-time rendering software and the software used to prepare the geometries and to apply the image textures. This kind of problem is very usual when dealing with exchange file format and usually it can be fixed modifying the software settings while saving the file. In fact, in that case, we saved new VRML file of the damaged geometries and we substituted the objects with missing elements.

Another problem raised during the import step was that most of the 3D models showed wrong normals on the polygonal surface. This problem is connected to the fact that the real-time rendering software we used is optimized for the NURBS surface import and not for the polygonal meshes import, even if they are supported. To solve this problem we had to recalculate the polygon normal distribution for each 3D mesh. During this operation a special parameter must be fixed: the "crease angle". This parameter represents the maximum angle the polygons can have in relation to one another, to retain shared vertex normals and thus also soft shadowing on the shared edges. To have the best shadowing of the polygon edges the crease angle must be set to the right value according to the geometric characteristics of every model.

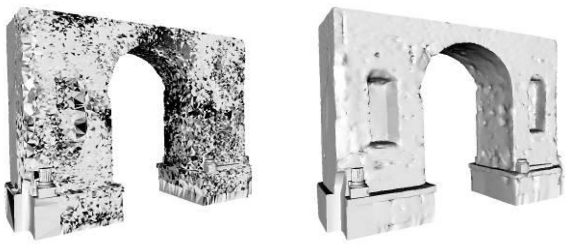

Figure 5. A model of a building showing wrong normals orientation.

Finally, once all the main structures in the Forum were correctly imported and checked, we imported the findings. As previously mentioned, these elements are smaller than the other objects in the scene and they are characterized by a low level of geometry details. Nevertheless, they are very relevant for the overall appearance of the Forum area, especially because of their large number. For this reason, it was impossible to manage a manual import and to check all the elements, so we decided to proceed with an automized procedure. We imported all the geometries in one single step, performing all the fixing procedures, even if they were not necessary on some file. To fix the normal distribution we fixed the crease angle on 70°. We choose a high value because most of the finds have almost cubic shapes, with almost perpendicular surfaces. This value allows also to have a soft shadowing of all "rounded" objects, such as columns, without checking in detail the finding geometry.

Figure 6. A model of a find showing wrong normals orientation.

Before starting with the automatic import of the findings, we tested the procedure on a sample find on the center of the Forum area. Thus, it was possible to check the right placing of the objects before starting the whole process. During this preliminary test we noticed an inconsistency between the right position of the find on the Forum area and the current placing of the finding model compared to the ground mesh. The source of the problem is the previous step of polygonal mesh and image texture set up. In fact, the 3D models of the findings were treated as separate items, every one with its own reference system coming from the instrument used for the 3D survey. To solve the problem, we used the reference system obtained from the ground model as a common reference system to place every object in the right place: the oriented cloud of points generated with the 3D scanner was used to define the transformation to be applied to each model. For each model the 4x4 matrix, in homogeneous coordinates, with the roto-translation values was used to obtain the six orientation parameters generally used by modeling software.

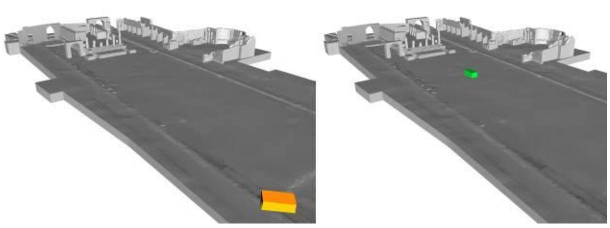

Figure 7. Wrong placing of objects into the scene: the finding was originally positioned with wrong orientation (left) and then corrected to its right position.

The text database containing all the 3D survey values was used to start a batch procedure to correct the VRML script, in order to modify the placing of the objects, avoiding any interference with the already performed texture placing and mesh editing steps.
After the VRML files containing the finding models were corrected, we started the automatic import procedure and we obtained a single file containing the whole Forum area, with all buildings and objects that were there at the 3D survey moment.

5 Database optimization

The whole database is composed by 209 VRML files and 213 JPEG files. This means that the real-time rendering software must manage and display 610,128 polygons and 117 MB of image textures. This large database is very difficult to manage, even if this kind of software is designed to control very complex and detailed 3D models. In fact, during the import process we noticed a remarkable reduction of workstation performance. The workstation we used on the project is a Windows based system, with Dual core processor 2.40GHz, 4 GB RAM and Nvidia FX5500 graphic board.
We performed different tests to check the memory usage for both CPU and GPU, in order to understand if the main problem was the geometric or the image database.

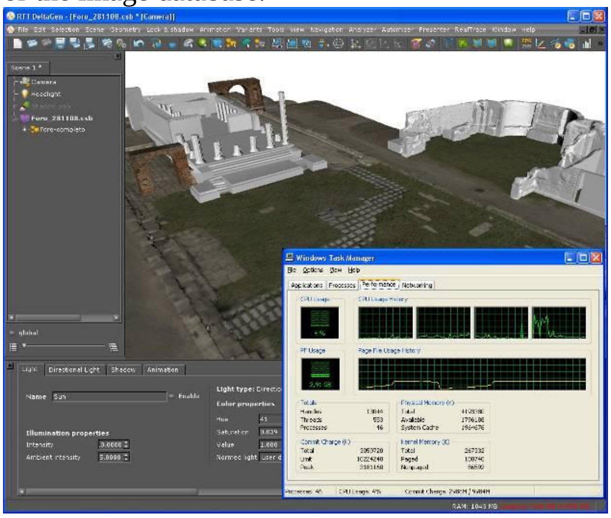

Figure 8. CPU/GPU usage test: textures ON.

First we displayed both images and geometry database and we checked the memory usage.
Afterwards, we discarded all the image data, displaying only the geometry of the Forum elements. We checked again the memory usage and compared the new values with the previous results.

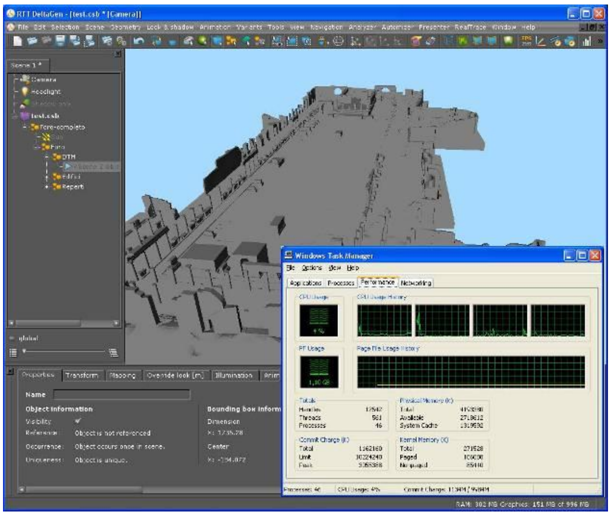

Figure 9. CPU/GPU usage test: textures OFF.

The test shows that the CPU usage is almost the same in both situation and it is not a critical issue. On the opposite, the main problem to solve is the GPU memory usage: the geometry database uses a small amount of memory and it can be easily managed by the workstation; on the contrary, the image database require a large amount of GPU memory, according to the images size.
From this test, we concluded that the actual bottleneck is represented by image handling. Thus, we had to optimize the information database by reducing the image size.

The texture reduction followed different strategies according to the 3D model typology.
First of all, we considered the necessity of displaying high quality texture on every model type; second, we checked the image resolution comparing the image size to the geometry average size; finally, we considered the number of image files used by every type of models.
The ground model is very large, and it is always visible on the background, even when focusing on a specific Forum item. Thus, it is necessary to have a high image quality on the texture applied to this element. Moreover, it has a single image applied to the whole surface, with a final low resolution of

about 30 pixel/m. For this reason, we decided to maintain the ground texture to its original size.
The building textures had original resolution almost hundred times higher than the ground model, so that a reduction would be necessary in order to obtain a similar image quality when displaying both models at the same time. Moreover, this models have many textures to manage, so we decided to reduce the images. Nevertheless, the high detail of these objects and their relevance for the overall display result, imposed a small reduction intervention. We decided to halve every image side, reducing the texture area to a quarter of the original image size.
For the findings, the original resolution was even higher than the building texture resolution, thus, it was possible to perform a larger database reduction. Moreover, these objects are smaller than the buildings and they are less important for the overall perception of the scene, even if their number is very high. Finally, the very large number of images to manage suggested a large reduction of the image size. We admitted a maximum texture size of 1024x1024 pixel, reducing every image side of a quarter.
After the database reduction we tested again the workstation performances, and we found that the GPU memory usage was now acceptable, and the scene can be easily managed. Thus, we proceeded with the next step of the process.

6 Making the Forum more realistic

To add a realistic look to the whole scene it was necessary to add lights and shadows. In fact, the lack of realism of the virtual environments is often due to the inadequate quality level of the scene lighting. In the real-time rendered scenes two lighting strategies can be used: most of the real-time rendering software can manage both pre-calculated or real-time shadows.
Pre-calculated shadows are applied to the objects surfaces on the scene as additional textures using the transparency channel. Usually, they look smooth and highly realistic and they cause just a little reduction of the software performances. Nevertheless, they must be calculated in advance and the processing time can be very long, according to the scene complexity. Moreover, the results cannot be evaluated in advance, so, to correct a light that has been incorrectly placed, it is necessary to re-do the whole calculation after correcting the light position. Finally, the shadows are “fixed” to the surfaces and no changes can be made without replacing the whole shadow textures set.
In contrast, real-time calculated shadows are immediately displayed and the light position can be interactively modified. Moreover, it is possible to create light animations producing the realistic motion of the shadows onto the surfaces on the scene. Usually, the real-time shadows quality is lower than the pre-calculated shadows quality and they look less realistic. To improve the shadows quality the software needs to increase the memory usage, decreasing the system performances.
For this project we decided to use real-time shadows, mainly for two reasons. First of all, the whole process required high flexibility and it was not possible to spend a long time for the shadows calculation. Moreover, the main visualization problem was in the texture database dimension so that we had to reduce it considerably. Thus, it was not possible to add other image textures to an already over-loaded database.
To choose and position the light, we made some preliminary remarks on the place characteristics and on the image textures quality. The general guideline we chose to follow is to keep an effect that could be consistent with the original light condition that could be observed in the real place. This solution was chosen also to minimize some imperfection that could be observed on the image textures. In fact, it was impossible to clean the whole large image database: some texture presented residual shadows projected by the surrounding structures.

Figure 10. Real shadows on the image texture.

In this condition, a light reproducing exactly the original situation would produce the same shadows that could be observed onto some surface on the digital model: thus, the error can be easily minimized without a difficult and long intervention on the original database.

Figure 11. Set up of the directional light: choosing the position and the color temperature.

We chose a light source that could easily reproduce the sunlight, especially because we knew the geographical references and the right orientation of the archaeological site. We chose a directional light oriented like the sun at the Pompei latitude during the survey period. Most of the photographs were taken at about midday during the month of june; thus, the directional light is coming from south, with an almost vertical inclination.
Moreover, the color of the light can be exactly established: it has a color temperature of about 5,800 K.

7 Set up of the real-time visualization

The final step of the process was to set up the digital camera and to prepare the file for the real-time interaction with the digital scene.

First of all, we set up the correct eye separation to have the better stereoscopic perception of the virtual environment.

The stereoscopic visualization reproduces the 3D perception of real things, given by the distance between the observer's eyes. Two digital cameras placed at a fixed distance and looking at the same point are used in virtual environments to artificially obtain a realistic result. When objects are displayed at the real scale, the distance between the two cameras must be set to the real eye separation value. Of course, this value is not constant and varies from a person to another; nevertheless, the average value of about 7 centimeters is usually accepted. Obviously, in virtual environments different eye separation values can be used, for example when the model is not displayed on its real scale or when special 3D effects are required.
For this project, we had to consider two main issues to set the better eye separation value. First, the whole model covers a very large area, so it necessary to pass to different display scales without losing the immersion feeling given by the stereoscopic visualization.

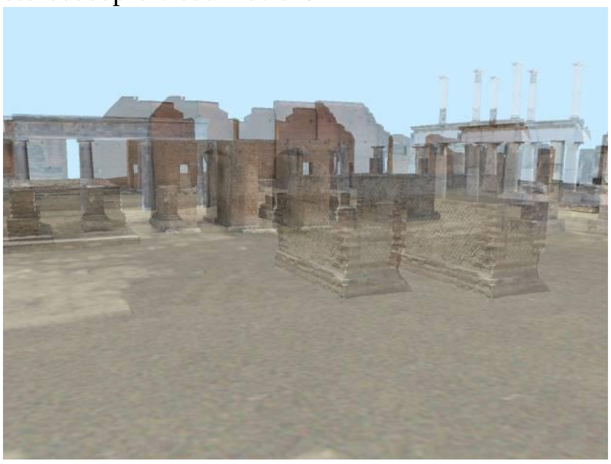

Figure 12. Setting up the right eye separation for the visualization on a large screen.

Moreover, when observing the scene, several details at different distance from the observer's eyes are displayed at the same time. Thus, it is useful to improve the stereoscopic effect increasing the eye separation value, in order to keep a higher 3D effect, even for the objects on the background. For this reason, we set a very high eye separation value, almost ten times bigger than usual. That way the stereoscopic effect for nearest objects is very strong and impressive; at the same time also distant objects keep a good tridimensional effect, helping the observer to perceive the scene depth.

Figure 13. The real-time visualization of the Forum.

Finally, we set up the navigation mode to explore the digital Forum area. Two navigation paradigms can be chosen: the “examine” mode and the “walk” mode. The first navigation type gives complete freedom in moving the virtual camera. Thus, the user can explore every object on the scene from every point of view; he can perform also non-realistic flights above the scene, gaining the opportunity to check every detail on the scene. This navigation mode is very useful during technical reviews, when the focus is on exploring and checking items on the scene: in fact, it is very simple to change the point of view and to skip from a model part to another. The walk mode, on the contrary, is very useful for less technical application when it is important to increase the immersion feeling on the virtual scene. The camera motion is constrained to a specific height and can only be shifted on the horizontal plane or rotated to change the focus point. Thus, the user can explore the Forum area as he was in the real place, as he was really walking through the ruins. At the moment, we performed only technical reviews, to check the overall quality of the 3D models and of their digital representation.

8 Conclusion

This paper described the process to set up a real-time rendered realistic scene of the Forum of Pompei, starting from texturized polygonal models. We followed a step-by-step process started with the models import and check, to fix possible geometrical problems and placing issues.
Afterwards, the CPU and GPU memory usage was tested, in order to check the workstation performances and to optimize the database reduction strategy according to the main visualization problem.
Finally, to add realism to the scene, light and real-time shadows were added. In fact, the lack of lighting effects in the real time virtual environments decreases the realistic effect of the overall scene. To have a lighting condition that was consistent with the applied textures, we build a light set based on geographical information. We used a single directional light that could reproduce the sun position at the moment of the survey, according with Pompei latitude and the survey time.
The real time rendered model of the Forum was then displayed on a large screen with stereoscopic technology, as a support for archeological studies.

Bibliography

[1] Bryson, S., *Virtual reality in scientific visualization, Communications of the ACM,* 39, Association for Computing Machinery, New York, 1996.

[2] Burdea, G.C., Coiffet, P., *Virtual reality technology*, Wiley-IEEE Press, Hoboken, 2003.

[3] Celentano, A., Pittarello, F., *Observing and adapting user behaviour in navigational 3D interfaces*, AVI'04: Proceedings of the working conference on Advanced visual interfaces, ACM Press, New York, 2004.

[4] Guidi, G., Remondino, F., Russo, M., Voltolini, F., Rizzi, A., Menna, F., Masci, M.E. and Benedetti, B. (2008), "A multi-resolution methodology for archeological survey: the Pompeii forum", *Conference in Virtual Systems and Multimedia Dedicated to Digital Heritage (VSMM)*.

[5] Guidi, G., Remondino, F., Russo, M., Menna, F. and Rizzi, A. (2008), "3D modeling of large and complex site using multi-sensor integration and multi-resolution data", *9th Int. Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)*: 85-92.

[6] Heidrich, W. and Seidel, H.P., "Realistic, Hardware-accelated Shading and Lighting" in *Alyn Rockwood, Ed. Computer Graphics Proceedings*, Annual Conference Series, 1999, pp. 171- 178.

[7] Möller, T., Haines, E., *Real-time rendering,* A K Peters, 1999.

[8] Rushmeier, H., Rogowitz, B.E., Piatko, C., "Perceptual Issues in Substituting Texture for Geometry", *Proceedings of SPIE--the international society for optical engineering*, vol. 3959, 2000, pp 372-383.

[9] Stuart, R., *The design of virtual environments,* McGraw Hill, Milano, 1996.