Crystal Ball, Virtual Gallery Creation System for Immersive Environments

Rui Nuno de Almeida
MEI-M ISCTE-IUL - Lisboa
rnaao@iscte.pt

Miguel Sales Dias
ISCTE-IUL – Lisboa
MLDC, Microsoft Language Development Center – Porto Salvo
miguel.dias@microsoft.com

Abstract

This paper presents a concept and a tool suite for authoring, configuring, organizing and displaying 3D virtual galleries, depicting multimedia content, like art, picture, movie or any other kind of media, which are appropriate for large scale immersive virtual environment such as the CAVE™. *A comparison was established by the authors with standard PC environments, while experimenting virtual environments produced by Crystal Ball, showing the superior experience obtained by an installation of the type of* CAVE™ *that ins operation at Lousal, south of Portugal.*

Keywords

3D, *Virtual Reality, Immersive Virtual Environments, On-line Virtual Worlds, Virtual Galleries*

1. INTRODUCTION

Over the last years, 3D virtual environments and in particular, immersive 3d virtual environments, have turned into a popular tool specially for critical industries like architecture, oil exploration, aeronautics, flight simulation, defence training and medicine, amongst many others. These environments and new 3D user interfaces will likely provide the user with a different and more complete experience than the usual desktop PC with a keyboard, a mouse or a joystick. Content creation and application interfaces for such environments still have a long way to run, but it is now possible for people with limited knowledge of programming and/or scripting languages, to author, configure and use 3D and multimedia content for immersive environments.

To address the need of creating multimedia rich content to be experienced in large scale virtual environments, we propose a new concept and suite of tools, able to generate 3D and multimedia content for these, the Crystal Ball. The rich content produced by such tool suite, was tested both in desktop-based PC systems and in the CaveH, an immersive virtual reality environment that exists in Lousal, in south of Portugal. The system was developed in C++ and was based on the Openscenegraph [openscenegraph.org] and includes a series of tools available in the community. With Crystal Ball we can configure and customize galleries in a text file, which after being imported into the CaveH middleware will build a ready-to-use scene in the 3D virtual environment.

Crystal Ball galleries are made of several nodes. Each of these nodes may have a variable number of images, or other "display elements". Nodes can connect to other nodes by "link elements", thus generating a structured network. All those nodes are defined in configuration text files, with the location and other parameters. Images and other media are also defined in text files that the system imports turning the settings into elements in the scene graph. These files can be constructed manually or by computer applications. Nodes can be independent pieces of geometry made by any 3D modeller or artist, in any software tool that produces the common 3D formats, such as 3DS or OBJ. The nodes of the gallery will have to comply with certain pre-determined size, so that the configuration of "displays" will stay in line with the respective node.

In this paper, we will start by providing in section 2, an insight in related work regarding virtual galleries authoring systems and tools. In section 3 we present the overall concept of how Crystal Ball defines virtual galleries. Section 4, presents the Crystal Ball scene description in more detail. In section 5, we present the basic technical approaches for the Crystal Ball content development and describe how we have performed PC and immersive virtual reality experiments, highlighting some of the findings. Finally, in section 6, we extract some conclusions and point some line of further research.

2. RELATED WORK

Three dimensional, spatially-oriented, exploratory and multiplayer 3D virtual environments are not a new thing, and have been used worldwide, particularly in education **[**Schendel08**]**.

2.1 Virtual Worlds

In Virtual Worlds like Second Life® [SL09], Virtual Galleries, can be seen all over the Grid. “*In 2007 there were more than 150 museums in Second Life, such as the International Spaceflight Museum (Spaceport Alpha 48, 78, 24), Second Louvre Museum (Tompson 153, 97, 100) , Sci-Fi Museum (Indigo 75, 213, 22), Bayside Beach Galleria Museum of Contemporary Art (Flyingroc Chung 109, 118, 22), Paris 1900 (Paris 1900 9, 174, 16)*” [Urban07]

Figure 1– Outside and inside of Second Louvre Museum (Pictures: Bettina Tizzy).

Recently the makers of the 'Second Louvre Museum’ (Fig. 1), announced that they intend to close its creation and have put on sale this replica of the real Paris museum. The Second Life world creation “bubble boom”, occurred specially during the year of 2007. At that time, thousands of companies created virtual environments to put their presence in this new platform for marketing and communications. The following year many companies have found that virtual worlds users (also known as “residents”), occupied themselves with the affairs of a social network, and rarely stuck to business premises and services. As a result, many of these companies and organizations have abandoned Second Life. Apparently, residents only occasionally looked for a virtual art gallery experience even when it is a spectacular creation such as the replica of the Louvre. The residents were in fact not interested and the number of visitors was very low. They looked mainly for other virtual spaces for social engagement. We can conclude that many of the residents want to create things for themselves and only occasionally visit big museums and artwork galleries.

Despite the proven disinterest of residents, organizations like ‘TheArchNetwork’ keep developing art projects like the recent 'Frank Lloyd Wright Virtual Museum' (Fig. 2), which opened during 2009 in Second Life ®.

Figure 2 - Frank Lloyd Wright Virtual Is-land/Museum (Pictures: Palup Ling).

The site has now been officially recognized by the Frank Lloyd Wright Foundation, the Foundation created in 1940 by Frank Lloyd Wright and one of the greatest symbols of Modern Architecture.

On an island exclusively dedicated to Wright, visitors can depict many of its most emblematic works (mainly residential), as well as permanent and touring exhibits. The island / Museum have exhibitions like the one on 'Usonian Houses' (the 'style' of American homes identified by Wright) [Frias09].

The most common exhibitions in Virtual Worlds are promoted and developed by the artists themselves (see Fig.3), to exploit their work and get close to the public. They usually use socially prestigious or crowded virtual spaces (hub Islands) to build the galleries.

Figure 3 - Exhibition of Vasco Lago Pinto (Jazz62 Masala in SL™). Black and White pictures of the artist exposed on “LX Gallery”, in Portucalis Island, in Second Life®. Galeria LX,on Portucalis the island, in Sec-ond Life ®, part of 'Art Galleries in Second Life' network. A retrospective exhibition of all artists which exhibited their works over the year 2009.

2.2 CANVAS [Kaczmarski05]

A team at University of Illinois in USA, developed CANVAS (Collaborative Advanced Navigation Virtual Art Studio, Fig. 4). It is a room-sized immersive 3D environment with origins in CAVE™ technology from the Electronic Visualization Laboratory at the University of Illinois, Chicago. It is a scalable, reconfigurable display technology for modern art museums and is not intended as a work itself but as an environment to facilitate the creation and display of immersive art works. CANVAS is collaborative because it can be connected to an array of geographically dispersed immersive virtual spaces, has advanced navigation to allow viewers in different locations to interact with virtual art and allows for the creation and presentation of virtual art that exists not in two or three-dimensional space like a painting or sculpture, but in the multi-dimensional world of virtual images. Viewers see the CANVAS in 3D stereoscopy, by wearing passive stereo glasses.

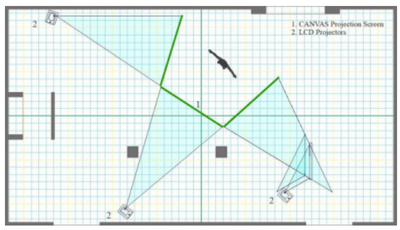


Figure 4 - The layout of the CANVAS at the Krannert Art Museum showing (1) the three-walled projection screen and (2) the set of six LCD projectors. Illustration by Camille Goudeseune.

KAMScript, [Marshack07] developed by the ISL and the Krannert Art Museum at the University of Illinois at Urbana-Champaign, is a system that allows non-programmer artists to create immersive 3D environments that can be experienced in the CANVAS.

2.3 VIRTUAL ART GALLERY TOOL [Semiao08]

Pedro Semião and Beatriz Carmo from the Science Department of Lisbon University, developed a Virtual Art Gallery tool set (Fig. 5). The purpose of this tool entirely based in free software is to allow users to interactively create a virtual exhibition of artworks in a pre-built virtual model. It uses X3D for the models and Java with Xj3D for the display and handling. The works of art are handled through a MySQL database. The toolset comprises the Virtual Art Gallery tool, the Space Picker application and the Virtual Exhibition Builder application (Fig. 6).

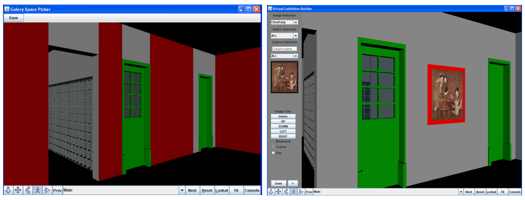


Figure 5 - The Space Picker application and Virtual Exhibition Builder application.

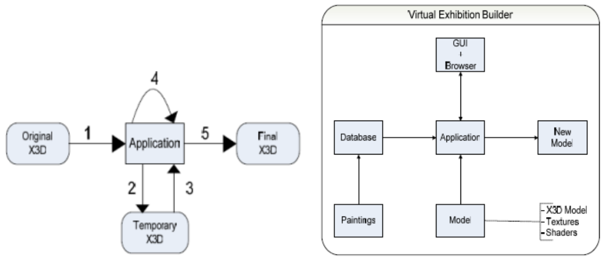


Figure 6 - Space Picker Data Flow and Virtual Exhibition Builder Application Layout.

2.4 XVM scene description [Lu08]

XVM [LU 2008], is a XML based scene description language for 3D virtual museum (Fig. 7). XVM describes several classes of nodes like dummy nodes, built-in shape nodes, model appearance nodes, line wall nodes.

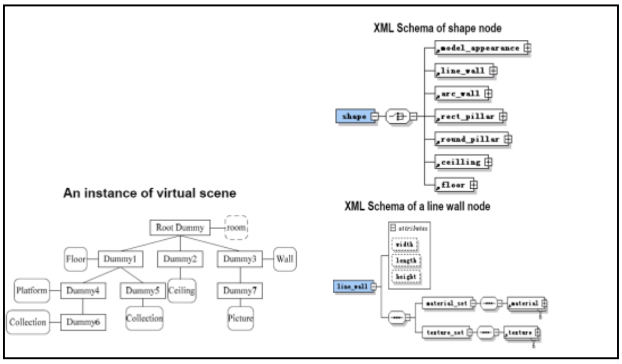


Figure 7 – Virtual scene and Node representations in XVM.

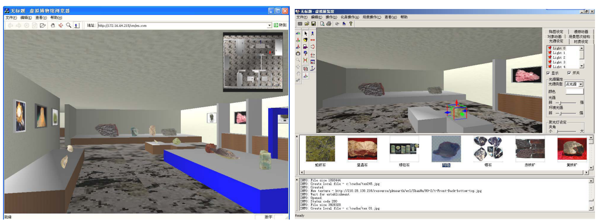


Figure 8 - GUI of virtual navigation and editing scene.

2.5 KAMScript describing [Marshack07]

Another way of describing a virtual gallery scene, is to use KAMScript (Fig. 9). This is a tool that allows people with no programming experience to create three-dimensional virtual environments that can be displayed on the University of Illinois CANVAS system. Users simply create a text file that describes the location, orientation and size of 2D images (in JPEG file format) and 3D models (in WaveFront *.OBJ format). KAMScript reads this text file, along with the supporting data files and renders the scene. With KAMScript, sounds (in the form of MP3 files) can be inserted into the virtual world at defined locations. Primitive volumes - such as boxes, ellipsoids, tori, cones and teapots – can be created in any RGB color. KAMScript essentially provides access to various openGL and Syzygy functions (the graphics and programming libraries used to create applications for the CANVAS cluster-based CAVE™), but does not require users to have full knowledge of what these functions are or how they are implemented.

```
// This is a comment.

IMAGE(helloworld.jpg,0,3,-2)

IMAGE(helloworld.jpg,[2,3,2],[0,45,0],[2,2])

IMAGE(helloworld.jpg,[-1,5,2],[45,0,0],[2,2])

SOUND(music.mp3,0,5,0)
```

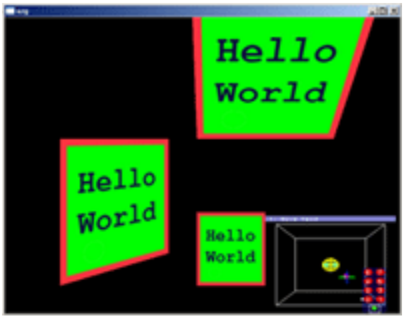


Figure 9 - An example of a KAMScript text file and resulting world.The first line places the image named “helloworld.jpg” at x=0, y=3, z=-2. The second line adds rotation and scaling as the second and third sets of parameters. The resulting image is shown using the standalone player.

3. CRYSTAL BALL (CBG)

We propose Crystal Ball as a concept and a series of tools to facilitate the authoring, configuring, organizing and displaying virtual galleries in 3D immersive virtual environments, which was much influenced by the CANVAS system. The Crystal Ball concept is based upon three entities – “nodes”, “links” and “displays”. To better understand what nodes and links are, we present a practical example of its parameters. We will start by assuming we want to create a room (node) for 6 planes "displays" with pictures or paintings of an artist. It is like a room in which the avatar will travel when visiting the Virtual Gallery. The designer and the gallery planner decide the basic dimensions for the nodes as well as the shape of the node, diameters and other measures as in the example of Fig 10.

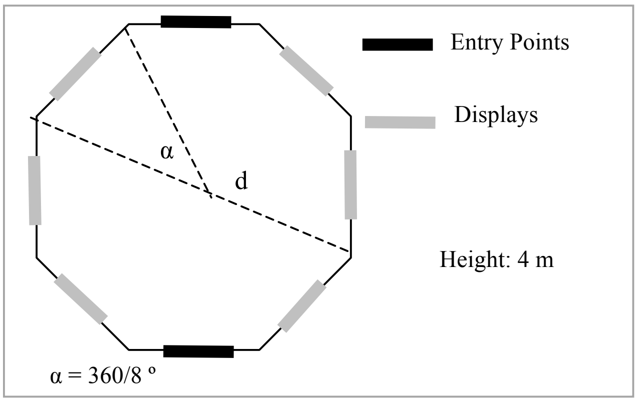


Figure 10 – Example of a node and his dimensions.

The modeller may create a *node* in accordance with the specifications and it will be added to the geometry of the scene. Each node can be reused as often as necessary in many other rooms or galleries. *Nodes* can be *empty* or filled (Fig. 11). A *filled node* is a node with a defined set of displays.

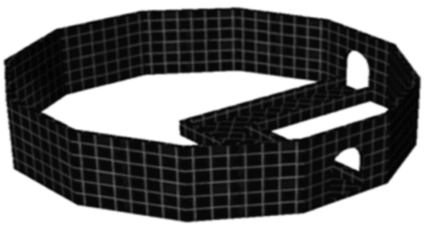


Figure 11 – Ready-made 3D empty node in Crystal Ball.

A *filled node* will have the displays placed on the defined geometric position. Nodes can have connections between them. Those connections link the *entry point* of one node to an *entry point* of another node. For this work we assume a maximum number of 4 links for each node. Seen from top view, the gallery will look like an organized grid structure that reminds a Printed Circuit Board (Figs. 12, 13).

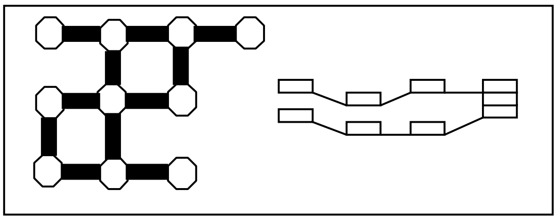


Figure 12 – Example of top view and side view of a CBG.

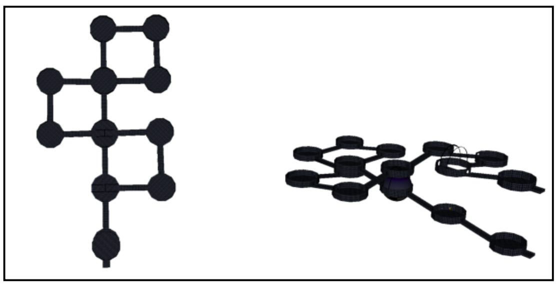
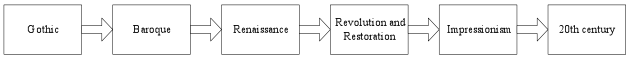


Figure 13 – Down view and perspective view of a CBG.

Each gallery can have an unlimited number of connected nodes. The only limit is machine storage and memory capacity. Based on a user-defined "flow file" and nodes topology, the system creates a corridor made of pre-defined planar or non-planar links.

Nodes can be connected according to a logic flow, in which the avatar will travel following a prescribed order that makes sense for that gallery in particular. The complexity of the structure depends on the context and multimedia contents of the gallery. The aim of the CBG system, is not the creation of art galleries simulating the real world ones, but rather create a logical navigation for the avatar in a tutorial manner. Suppose we want to create a gallery to make a virtual travelling system for the all contents of the Paris Louvre Museum. For sure we can imagine how big and complex that structure would be. We may want the avatar to flow inside the gallery with certain progression logic, like the following:



We must then build a linked structure between the nodes always following that order. This way we can define a workflow to orient and simplify the visiting or learning experience.

4. SCENE DESCRIPTION IN CRYSTAL BALL

Our scene description based in text files, was the way to reach our goal of building the scene with the simplest structure, but is obviously extensible to something more structured, like the XML-based XVM scene description language for 3D virtual museums.

4.1 Display Structure

In the Crystal Ball approach, we will simply call nodes to our display structures. Each node has a set of properties, and the gallery is the set of all nodes, links and displays.

4.1.1 Plane Displays

Plane displays can be representations of pictures, paintings, movies or any other piece of artwork that can be represented in a planar surface. There are 12 properties defined and stored in the "display set" text files:

ID	Key identifier
Node_ID	Node Key identifier
Width	Represents the physical width of image in pixels.
Height	Represents the physical height of image in pixels.
Real Width	Represents the real world with of the image or painting.
Real Height	Represents the real world height of the image or painting.
Image path	System path where the image is stored
Display description	Description to be used in the scene graph next to the display. For example, "Las Meninas" or "The Family of Philip IV".
Relative X rotation	X-rotation to apply around the x axis in the middle point relative to the image, in the scene graph.
Relative Y rotation	Y-rotation to apply around the x axis in the middle point relative to the image, in the scene graph.
Relative Z rotation	Z-rotation to apply around the x axis in the middle point relative to the image, in the scene graph.
Relative X	Translation of the display over the x-axis.

position	
Relative Y position	Translation of the display over the y-axis.
Relative Z position	Translation of the display over the z-axis

4.1.2 Panoramic Displays

Panoramas can be represented by a 360ºx180º spherical picture panorama or a cylindrical or any other non-planar display. For this work we used the POV-Ray[POV] tool and we made scripts to produce images of spherical panoramas. We then mapped those images in 3D spherical models specially modelled for this kind of nodes. We also used spherical panorama images available under free public licenses, or under author consent.

4.2 Node Structure

A node is a piece of 3D Geometry in which the displays will be placed. It can be a 3D object modelled in Blender [BL] or any other 3D application, that will host the displays in a higher level node in the scene graph, for example a room space. Nodes are defined by 10 properties for each node inside the text file:

ID	Key identifier
Name	Name of the node. For example, "Velazquez 1620 - 1630". Can be used as text in the scene graph.
Description	Description of the node. Can be used as text in the scene graph.
Category	Category of the node. Used to classify different classes of nodes.
3D Geometry file name	Path + file name of physical disk location of the model file. For example "images\room4.3ds".
X rotation	X-rotation to apply around the x axis in the central point relative to the object in the scene graph.
Y rotation	Y-rotation to apply around the x axis in the central point relative to the object in the scene graph.
Z rotation	Z-rotation to apply around the x axis in the central point relative to the object in the scene graph.
X position	Translation of the object over the x-axis.
Y position	Translation of the object over the y-axis.
Z position	Translation of the object over the z-axis.

5. CONTENT DEVELOPMENT AND TESTING

5.1.1 Content Development

A number of tools were used for developing content in the Crystal Ball framework:

Microsoft® Visual Studio .Net 2003™ with C++ was used to program the base scene graph. The programming was done in Visual C++ with Openscenegraph libraries, generating a gallery structure and an OSG file.

Autodesk® 3ds Max 9™ [3DS] was used to author the 3D models of the nodes and links (see Figure 11 and Figure 17).This tool was also used to perform some 3D modelling conversions.

The Blender 3D modelling tool was used to give more detail to the 3D models of nodes and links.

OpenSceneGraph 2.0 [OSG], was the selected object-oriented framework on top of OpenGL, that manages the scene graph of the system. It includes an API to create and transform geometry elements, animations, visual effects and supports a set of 3D functions and plug-ins.

GtkRadiant 1.5.0 [GTK] (Fig. 14), is the adopted tool to create or edit ready-to use maps for the CaveH environment. This is a well known tool of Quake III game community and was adapted for the purposes of the CaveH platform. It supports reading of 3D file formats produced in several modelling applications. GtkRadiant also supports the definition of triggered actions.

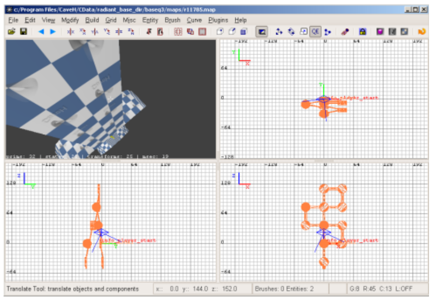


Figure 14 – GTK Radiant screenshot

OSG objects to be placed in the 3D scene can be generated in two different ways:

- Through C + + programming, using classes available in the OSG libraries, exporting the scene graph to a .osg file, containing geometry, lights, smoke or other visual effects.
- By modelling in Autodesk® 3ds Max 9™ or Blender and exporting the result model to the OSG format.

CaveHSpawner (Fig. 15) is an interface for a series of tools to create, edit and run ready-to use maps in the CAVEHollowspace. CaveHSpawner will compile the map produced by GTK Radiant into an octree structure in OSG format, covering all referenced objects and properties and producing as the output, a game environment like Quake 3. This environment will be distributed to the slave nodes of the multi-projection system of the CAVEHollowspace, in the topology described in the preceding paragraphs, using the CaveH middleware.

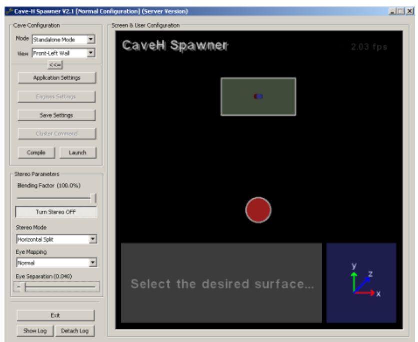


Figure 15 – CaveHSpawner screenshot

POVRay[POV], is an open source ray tracing tool. It has a powerful scripting language that uses complex math functions to create complete ray traced 3D scenes. It was used to generate 3D panoramas used on spherical gallery panoramas. A 360ºx180º panoramic virtual camera was used to make an image mapped on the spherical display object (Fig. 16).

Gimp[Gimp] is an Open Source tool used for picture/photo effects and adjustments. We have used this tool to improve textures and image colours.

The MeshLab[Mesh] tool was used to make 3D model corrections, adjustments and format conversions.

Last but not least, the Second Life® platform with LSL (Linden™ Scripting Language), was used to make a preliminary study and an essay on the plan of operation of virtual galleries. It helped to compose the storyboard of the Crystal Ball immersive experience.

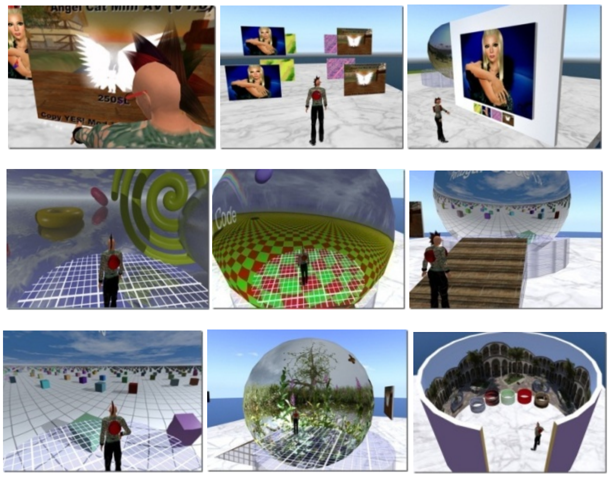


Figure 16 – Storyboard. [SL09] Second Life® World Pictures – Rui de Almeida.

Second Life® World Modelling and Scripting – Rui de Almeida.

Image panoramas(screens 4,5,6) "PT-Code1" and "PT-Code2" and POV Ray Programming – Rui de Almeida.

Image panorama(screen 9) "Patio" in POV Ray by Jaime Vives Piqueres.

Image panorama(screen 8) "The Dark Side Of Trees" in POV Ray by Gilles Tran.

5.2 Preliminary Results

To assess the applicability of CBG, we have performed two experiments: a desktop PC experiment and an immersive Virtual Reality System (CaveH [Dias 07]) one.

5.2.1 Desktop PC Experiment

The first navigation experience was made by nine unpaid male subjects, selected for the undergraduate course of ISCTE-IUL, using Microsoft Windows XP® on a desktop PC, equipped with a 22-inch monitor and a 2 button + wheel mouse (Fig. 17). To navigate through the scene the subjects have used the CaveH PC viewer and also the Openscenegraph viewer. The Second Life® viewer was used for some earlier experiments. There was nothing new in what concerns to sense of immersion, as experienced by the subjects. It was a common 3D video game like experience. The mouse and keyboard navigation paradigm always left us with a sensation of being out of the scene, and being only an outside observer.

Figure 17 – CBG Desktop PC screenshots.

We have not detected any performance, colour or image processing problem. In general, image quality was perceived as good, but not wide enough for a Virtual Reality experience. It was a poor interaction experience when compared with live painting observation in a real exhibition.

5.2.2 Immersive Virtual Reality Experiment

The CAVE™ [Cruz-Neira93] is a room-sized, high-resolution 3D video and audio environment invented in 1991 at Electronic Visualization Laboratory of University of Illinois. It was created by Carolina Cruz-Neira, Dan Sandin and Tom DeFanti, along with other students and staff. Graphics are projected in stereo onto several planes, such as walls and the floor, and viewed with stereo glasses that can be equipped with a tracking sensor. As the user moves within the display boundaries, the correct perspective is displayed in real-time to achieve a fully immersive experience.

5.2.2.1 CAVE Hollowspace at Lousal [Dias07]

The CAVEHollowspace system, that follows the basic architecture of the CAVE™, is located at Lousal Mine in the Alentejo Region in Portugal that was operated between 1900 and 1988. The physical configuration of the CAVEHollowspace of Lousal, includes six projection planes that make a U-shaped projection surface (Fig. 18): there are two retro-projected planes in the front (5.6x2.7 m), two front-projected planes in the pavement (5.6x2.7 m) and a retro-projected plane on each side (3.4 m 2.7 m).

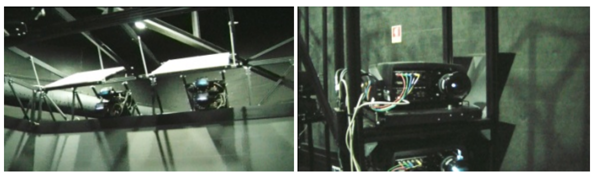


Figure 18 – CaveH projectors.

For stereo viewing, the Infitec passive system [INFITEC] is used (Fig. 19). There are seven speakers that distribute 3D spatial sound over a 7.1 topology, including a 800 watt subwoofer. The system is controlled by a network cluster of 8 computers running Microsoft operating systems and connected by a Gigabit Ethernet.

Figure 19 - CaveH Stereo projection and Infitec passive glasses and CaveH stereo projection.

5.2.2.2 CAVE Hollowspace Experience

The same nine subjects, who had previously experienced the system as a desktop PC, went to Lousal to experience the immersive system. Although this was a new experience for them, unlike the use of desktop PC, all were unanimous in referring an emerged and more involved feeling of immersion in the art gallery visiting experience, even to the point of feeling nausea when navigation was subject to rapid changes in direction and speed.

6. CONCLUSIONS

The ultimate immersive 3D Virtual Reality experience would be in an environment where we could walk, run, look around and feel the surrounding sound and other sensorial input, as we do in real life environments. This is definitely not possible in front of a desktop PC screen. For sure this is not yet fully possible in CAVE™ type of environments, but in this case, in addition to the psychological engagement, there is also an increased sensorial and a physical engagement.

In this paper we have presented Crystal Ball, as a concept and a tool suite for authoring, configuring, organizing and displaying 3D virtual galleries, depicting multimedia content, like art, picture, movie or any other kind of media. Our preliminary ad-hoc studies carried with nine students of ISCTE-IUL, gave us some hint on the superiority of large scale immersive virtual environments, such as the CAVEHollowspace (CaveH) of Lousal, in experimenting a visit to a virtual gallery produced by Crystall Ball, if we compare it to a standard PC virtual navigation experience. In fact, in CaveH, we can look and walk around, having the full perception of space and dimensions in 3D. At the same time we can feel the surrounding sound. All of this gives a different sensorial stimulation and a completely different 3D Virtual Reality experience. More complete usability evaluation experiments, involving a panel of test subjects and analysis of completion of task assignments via objective metrics is needed, to further extract conclusions about the benefits brought by immersive environments to the "virtual gallery visit" metaphor. In 3D Virtual Worlds, the user represented by an avatar, although having an immense amount of content, social networking engagement and a myriad of tools and possibilities for communication marketing, business, and entertainment, still is kept as an outsider observer over the world. Future human-computer interaction and visualization paradigms will bring the possibility of making available online virtual environments, more close to CAVE™ type of immersive systems, and the authors believe that Crystal Ball is a contribution in that direction.

7. REFERENCES

[BL] Blender Foundation *http://www.blender.org*

[Cruz-Neira93] Cruz-Neira, C., Sandin, D., DeFanti, T. "Virtual Reality: The Design and Implementation of the CAVE®" *Proceedings of SIGGRAPH 93 Computer Graphics Conference*, 08/01/1993 - 08/01/1993

[Dias07] Dias, M. S. et al., "CaveHollowspace do Lousal – Príncipios Teóricos e Desenvolvimento", *Curso Curto, 15 Encontro Português de Computação Gráfica, Microsoft, Porto Salvo*, Out 2007.

[Frias09] Frias P., "Palup Ling in SL", *Communication Sciences Department - Oporto University* http://discursosdooutromundo.blogspot.com/

[Gimp] GIMP - The GNU Image Manipulation Program *http://www.gimp.org*

[GTK] GtkRadiant, Maio 2008. *http://www.qeradiant.com*

[INFITEC] Infiscape, "INFITEC Stereo Display Technology", 2009 *http://www.infitec.net/infi_e.html*

[Kaczmarski05] Kaczmarski, H. ,Goudesuene, C., Schaeffer, B., Chong, L., Marshack, R., Hendriskson, L., Crowell, J. "CANVAS: A Virtual Reality Environment for Museums" , *EVA Firenze 2005*, "Application Framework for CANVAS, The Virtual Reality Environment for Museums", *EVA London 2005*.

[Lu08] LU, W., Zeng, D., Pan, J., "An XML-Based Scene Description Language for 3D Virtual Museum", *Department of Computer Science and Technology, Nanjing University, Nanjing*

[Marshack07] Marshack, R., Duchnowski, N., Watt, A., Jackson, J., Kaczmarski, **H.,** "Creating Immersive Art Without a Programmer: The first year for CANVAS, A Virtual Reality Environment for Museums", *EVA Firenze 2007*

[Mesh] Meshlab, *http://meshlab.sourceforge.net/*

[OSG] OpenSceneGraph, Jan 2009. *http://www.openscenegraph.org/*

[POV] POV-Ray The Persistence of Vision Raytracer *http://www.povray.org/*

[Schendel08] Schendel, J., Liu, C., Chelberg, D., Franklin, T., "Virtual Gallery Walk, an Innovative Outlet for Sharing Student Research Work in K-12Classrooms" , *Ohio University*

[Semiao08] Semião, P. M., Carmo, M. B., "Virtual Art Gallery Tool" , GRAPP 2008 - International Conference on Computer Graphics Theory and Applications, Faculdade de Ciências da Universidade de Lisboa,

[SL09] Virtual world Second Life *http://www.secondlife.com*

[Urban07] Urban, R. et al., "A Second Life for Your Museum: 3D Multi-User Virtual Environments and Museums", *in J. Trant and D. Bearman (eds.). Museums and the Web 2007: Proceedings, Toronto: Archives & Museum Informatics, published March 1, 2007 Consulted January 26, 2010. .http://www.archimuse.com/mw2007/papers/urban/urban.html.*

[3DS] Autodesk - Autodesk 3ds Max, 2009. *http://usa.autodesk.com/*