

DISTRIBUTED VIRTUAL ENVIRONMENTS FOR VISUALIZATION AND VISUAL DATA MINING

E. Paquet^a and H. L. Viktor^b

^a Visual Information Technology, National Research Council (NRC), M-50 Montreal Road, Ottawa (Ontario) K1A 0R6
Canada, eric.paquet@nrc-cnrc.gc.ca

^b School of Information Technology and Engineering (SITE), University of Ottawa, 800 King Edward St., Ottawa
(Ontario) K1N 6N5 Canada, hlviktor@site.uottawa.ca

^aCommission VI, WG V/2

KEY WORDS: Data Mining, Distributed Systems, Virtual Environments, Virtualisation, Visualization

ABSTRACT:

This paper presents a mobile virtual environment designed for the visualization of high-resolution virtualised scenes and for visual data mining. The architecture and the implementation of the mobile virtual environment are described. This environment is characterized by its distributed asynchronous architecture that optimises the rendering performances by distributing the graphical load over various computers. A client-server architecture insures the robustness of the connections and the scalability of the system. The various components of the virtual environment can be configured with an XML file. In stereo mode, two computers render the scene. The passive stereo projection system is composed of two circularly polarized DLP video projectors and the scene is rendered with two laptops (up to 400.000 polygons in real-time in this specific case). Both rear and front projection are supported. In tiled wall display mode, the number of computers is equal to the number of flat screens required for the visual display. Various applications of this virtual environment to visual data mining are considered. Finally, the system is utilized to visualize the crypt of Santa Cristina, a Byzantine crypt of the medieval period near Lecce (Italy) that has been scanned at high resolution with a MENS range scanner and texture mapped with photogrammetric techniques.

1. INTRODUCTION

Many applications spanning numerous fields of knowledge require visualisation [1]. Two of these are considered in this paper: mobile high-resolution scene visualisation and visual data mining. Most of the solutions that are currently proposed are characterized by their complexity, cost and size. This is due to the fact that low-end visualisation systems usually present poor performances that are incompatible with most applications.

This paper proposed a mobile, high-performance, cost effective virtual environment for visualisation and visual data mining. A conceptual and technical description is provided. The design is considered from various aspects: photorealism, the scene's size scalability, the display's size scalability, resolution and refresh rate; the user's visual comfort and intellectual property protection. The implementation of the mobile virtual environment is addressed. Two types cases are considered: a mobile stereo system and an ultra high resolution tiled wall display. The protection of intellectual property is of the utmost importance. In order to enforce such a protection, the content is encrypted in order to limit the access to its appearance and not to the data per se.

This paper is organized as follows. In section 2, we discuss the current Java-based implementation of our system. This is followed, in section 3 to 5, with descriptions of the architecture of the tiled wall display and the stereo system. In section 6, various applications of our system to visual data mining are discussed. Finally, in section 7, an example of visualisation is given with the crypt of Santa Cristina, near Lecce, Italy that has

been virtualised with a laser scanner and photogrammetric techniques.

2. ON THE IMPLEMENTATION OF THE MOBILE VIRTUAL ENVIRONMENT

As far as implementation is concerned, the choice of the API is of the utmost importance. Our system must fulfil many requirements. It must be portable on various platforms, have high graphical performances and it should not be bounded to any particular graphical API.

The implementation of our systems is based on Java: the Java 2 Platform, Java 3D and Java Cryptography Extension or JCE [2]. These API are currently maintained by Sun Microsystems™. Java 3D supports many operating systems including Microsoft Windows™, Linux, HP Unix™ and of course Sun Solaris™. The same bytecode can run on all platforms because of the Java compiler that compiles in real-time the code into the Java Virtual Machine or JVM. Such platform independence helps to level previous and future investments. Java 3D is based on either OpenGL or DirectX. The OpenGL API is supported by most platforms while the DirectX version is specific to Microsoft Windows™ XP/2000/NT.

Java offers various networking capabilities. In the current implementation, networking is based on TCP/IP. We also describe a system based on UDP that is currently in the process of being implemented

Content protection is of primary concern. The content should be accessible only to those who own it or who have obtained the right to use it. Content can be protected with encryption. Encryption is performed with the Java Cryptography Extension and relies on standard commercial encryption algorithms. The content is decrypted in real-time when loaded in the Java Virtual Machine. This insure that only the visual appearance of the content is accessible and not the data per se.

3. ON THE ARCHITECTURE OF THE TILED WALL DISPLAY

Tiled wall displays [3] are composed of flat LCD panels that are aligned on a wall. A large area is usually covered so that many persons can work simultaneously on the same data set at very high resolution. In order to implement such a visual display, we describe a system based on client-server architecture in which each client is responsible for performing the rendering for a determined LCD panel. The synchronization of the various displays is obtained by continuously sending requests from the clients to the server, in order to update their information about the input devices for which the server is responsible [4]. The clients update the scene information according to their relative position in the tiled wall display. Consequently, the graphical load can be distributed over various computers. The architecture is discussed in detail in the subsequent paragraphs.

SERVER COMPUTER 0	CLIENT COMPUTER K SCREEN K
Input events from: mouse, trackers, etc	
Transformation available to clients for queries	
	Send a request to server in order to obtain the transformation
Receive request from client	
Send transformation to client	
	Receive transformation

Table 1: One events loop in a tiled wall display. The server is in the left column, the client is in the right column and the time direction is from top to bottom. The size of the cells does not correspond to the effective duration of the events.

Let us consider the system from a server perspective. The synchronization is performed within the so-called event loop. The event loop is the loop in which the input events are collected from various input devices. Among them are the mouse, trackers, haptic sensors and so on. The events from the input device are collected at constant rate and converted into virtual environments coordinates. Let us refer to them the transformation. Those coordinates or transformation correspond to the virtual coordinates utilised in the virtual scene and depends on various calibration parameters, including the

dimensions of the visual display and the size of the scene. The transformation are then compressed and put in a data structure. Upon completion of the previous task, the structure is sent to the server, which makes it available to the clients. Each time there is a request from one of the clients, the server sends the structure to the corresponding client. The server has a multithread architecture: a thread is associated to each client. It has to be noticed that the events loop is totally independent from the rendering loop. The events loop affects the rate at which the navigation and manipulation information can be updated, but not the rate at which the graphical calculations are effectively performed.

Now let us consider the system from the client perspective. At the commencement of each events loop, the client sends a request to the server in order to obtain the transformation. After the request has been sent, the events loop is paused. One should remember that the server manages the input events. Upon reception of the data structure from the server, the structure is uncompressed, the client events' loop is resumed and the rendering is performed with the updated information. Since each client displays a different part of the virtual world, a transformation has to be applied on the coordinates in order to take into account the relative position and orientation of the client's screen compared to the server's screen. A schematic representation of the system is provided in Table 1. It should be noticed that there is a computer associated with each client and server.

Let us take a closer look at synchronization. In order to have acceptable visual performances, the refresh rate must be over a certain rate. This rate is function of the retinal persistence and has a value of around 30 frames/sec. Higher rates improve the visual quality and the overall usability of the virtual environment. Below the minimum rate, the display might not be comfortable and cause the well-known VR-sickness, which is similar from many aspects to travelling sickness.

Sockets' performances determine the time involves in telecommunication. Since the client-server architecture is based on the TCP/IP protocol, the time involved in a request is about 5 msec. All clients are synchronized relatively to the server, which means that the relative delay between them is around 10 msec. That means that it is possible to render around 100 frames/sec within the proposed virtual environment. In practice, the graphic load and the performances of the graphic card limit the frame rate. The graphic load is tributary of the number of polygons composing the scenes, the number of groups of polygons and the number and dimensions of the texture maps. But because of its distributed architecture, our system spread the graphical load between the computers. Each computer renders a subset of the overall scene, which means that better performances can be achieved both from the resolution and frame rate point of view.

This architecture is robust due to the client-server architecture. The performances of the system are not affected if one of the clients connects or disconnects. The disconnected client can resume connection at any time since the coordinates are sent in a non-incremental manner. The virtual environment is designed to resume automatically connection if one of the connections is lost. The virtual environment is also highly scalable and adaptive in the sense that it supports any number of LCD display in any configuration. This characteristic comes from the distributed client-server multithread architecture.

4. ON THE ARCHITECTURE OF THE PASSIVE STEREO SYSTEM

Stereo vision is inherent to our nature. It originates from the fact that our brain can reconstruct a three-dimensional representation by analysing the information coming from our visual system.

Active stereo virtual environments are based on the principle of retinal persistence: the left and the right views are rendered alternatively and sequentially on the display. In order to achieve depth perception, the user must wear shutter glasses that are synchronized with the display. When the left view is rendered, the left shutter is opened while the right shutter is closed and vice versa.

Active stereo presents a serious drawback: the bandwidth limitation. One depth frame consists of two sequential video frames: one for the left eye view and another for the right eye view. That means that the stereo rate is only half of the refresh rate. In practice, due to limitations on the graphic adapters, a high frame rate is obtained via low resolution or expensive systems.

SERVER RIGHT EYE VIEW COMPUTER 0	CLIENT LEFT EYE VIEW COMPUTER 1
Input events from: Mouse, trackers, etc	
Transformation available to client for queries	
	Send request to server in order to obtain transformation
Receive request from client	
Send transformation to client	
	Receive transformation

Table 2: One events loop of our passive stereo system. The server is in the left column, the client is in the right column and the time direction is from top to bottom. The size of the cells does not correspond to the effective duration of the events.

Our stereo virtual environment is based on passive polarization. Two circularly polarized video projectors simultaneously project the left and right eye view. Table 2 illustrates this. The left eye view is encoded with a circular left polarization while the right eye view is encoded with a circular right polarization. In order to experience stereo vision, users must wear polarized glasses. Circularly polarized filters performances do not degrade with head lateral inclination, which is why it has been selected.

Our projectors are based on Texas Instrument's™ Digital Micromirror Device or DMD [5]. The DMD is a semiconductor light switch made of thousands of microscopic

mirrors. DMD projectors preserve polarization and, for this reason, have been selected for our system.

In order to obtain depth perception, the output of the two projectors must be projected on a surface that preserves the polarization state. For front-projection, a standard metallic screen is utilised. For back-projection, a specialised screen is required but is commercially available. Front-projection is a cheaper option but back-projection offers more possibilities both from the visualisation and interaction point of view.

A transformation is applied to the client VE coordinates in order to emulate the disparity between the left and the right eye view. Because our virtual environment distributes the graphical load between two computers, high refresh rate and resolution can be achieved. Instead of rendering alternatively the right and the left eye view as in an active stereo system, our virtual environment renders them simultaneously on two separate computers. In addition, both views are projected simultaneously on the visual, which means that the bandwidth is almost quadruple. It is important to notice that stereo systems must be calibrated properly in order to take into account the dimensions of the display, the eye separation, the field of view and the viewing distance from the display.

A schematic representation of the system is provided in Table 2. The similarity between Table 1 and 2 should be noticed. The same virtual environment is used for the tiled wall display and the stereo system. A custom-made builder configures the virtual environment via an XML file. This file contains all the information for the configuration of the system, both from a system and scene point of view. This information includes the parameters for the virtual environment (screen size, disparity, etc), the scene (lighting, manipulations, objects, rendering, etc) and for telecommunications between the various components (server or client, IP address, port, etc). This implies that our system provides a general framework for virtual environments.

5. ON THE ARCHITECTURE OF THE HIGH REFRESH RATE, LOW LATENCY PASSIVE STEREO SYSTEM

The architecture proposed so far is characterized by its scalability, adaptability and robustness. Sockets connections are based on the TCP/IP protocol. This protocol is robust but relatively slow. For applications in which refresh rate and not robustness is the main concern, faster protocols must be considered. We propose a new architecture based on the UDP protocol in which the synchronization is performed within the rendering loop, as opposed to the previous architecture in which the synchronization is performed within the events loop. This system is currently under development and is described in the following paragraphs. It should allow better refresh rate and very low latency between the two computers.

The UDP protocol is a very low-level networking interface. It is not stream-based and does not establish any connection between the sender and the receiver. Also, there are no attempts to insure that the packets have arrived or not. Because of the low overhead, the transmission is very fast.

In the proposed architecture, a computer called the sender collects the input events. This computer also renders the right eye view. A second computer, called the receiver, receives the transformation from the sender and renders the left eye view.

SENDER RIGHT EYE VIEW COMPUTER 0	RECEIVER LEFT EYE VIEW COMPUTER 1
Input events from: mouse, trackers, etc	
Send transformation to receiver	
	Receive and uncompress transformation
Render scene	Render scene

Table 3: One rendering loop of the UDP-based passive stereo virtual environment. The sender is in the left column, the receiver is in the right column and the time direction is from top to bottom. The size of the cells does not correspond to the effective duration of the events.

In order to obtain low latency, the synchronization is performed within the rendering loop. The rendering loop of each computer is accessed by extending the Canvas3D class of the Java 3D API and by overriding the preRender() method. On each call of the preRender() method, i.e. before each rendering, the sender collects the events from the input devices, convert them into a transformation and send this transformation to the receiver.

At each call of its preRender() method, the receiver receives the transformation and applies it to the scene in addition to the disparity transformation. Rendering is then performed on both computers. The scene is not transformed if there are no new inputs from the input devices in order to minimize the amount of calculations. A schematic view of the virtual environment is provided in Table 3.

6. THE MOBILE VIRTUAL ENVIRONMENT AND VISUAL DATA MINING

This section explores the use of the mobile virtual environment for data mining.

Data mining concerns the discovery of novel, previously unknown and meaningful knowledge from data. Data mining and its related technologies are especially useful in complex domains, which contain numerous free parameters. Data mining techniques such as classification techniques, association rule mining and clustering techniques can be fruitfully used to provide us with a better understanding of our mobile virtual environment [6].

Recall from Section 3 that, in the current client-server implementation, data regarding the graphic loads of the clients and servers are maintained. Through data mining, typical patterns with respect to the load can be detected and predicted. From this information, the distribution of the rendering load over multiple machines can be optimised. For a particular scene, this data may be collected in real-time, through the repeated rendering thereof. Data mining can subsequently be used to predict which subset of the overall scene should be distributed to which client. This prediction thus includes information such

as ideal subset size and the optimal number of clients. In this way, the best performances can be achieved both from the resolution and frame rate point of view.

Recall that our system inputs data from multiple sources, including the mouse and trackers. In order to better understand how users experience and interact with our system, the events as obtained from the input sensors may be analysed. For example, experience shows that different users are more prone to VR-sickness. Through adapting the rendering loop and other parameters, a better understanding of individuals' personal tolerances should be obtained. Also, the impact of other factors on our users' experience, including parameters for the screen, virtual environment and telecommunications, may be analysed through data mining.

The mobile virtual environment can be used for clickstream analysis, which creates profiles of users' typical interaction with the environment [7]. For example, consider a virtual environment, which visualizes an art gallery. The method in which a user "walks through" the virtual environment; the duration he/she remains in a particular virtual room, views a certain item; and the number of times he/she returns to view an item, all indicates the "points of interest" in our virtual environment. The results of the analysis of such clickstream may subsequently be used to re-order the rooms of our virtual environment and its contents. The applications of this knowledge to e-Business, in order to streamline on-line purchasing and marketing, amongst other, are numerous.

In particular, the extension of mobile virtual environment for visual data mining is proposed. Visual data mining, which is defined as the discovery of knowledge through data and/or knowledge visualization, consists of the integration of two disciplines, namely data visualization and data mining [6]. Current data visualization systems typically view data through a single graphics computer screen in visual formats such as boxplots, distributes charts, 3-D cubes and link graphs. This representation, though useful, has many limitations, mainly due to humans' inability to handle scale and the complexity of typical data repositories [8].

Consider the use of the mobile virtual environment for visualizing the results of data mining in an informative way. A major data mining task is cluster analysis, which involves finding similar groups of data. Here, the original data set is segmented into clusters of data with similar tendencies, based on some metric such as the Euclidean, Manhattan or Minkowski distances [9]. For example, a clustering algorithm may segment the users of an on-line art gallery into having similar purchasing patterns, gender and age. The walk-through behaviour of the members of each of these clusters can subsequently be visualized in the mobile virtual environment, together with the information regarding their common profiles.

In addition, the use of the mobile virtual environment to interactivity visualize the data sets to be used for data mining is proposed. In this proposed system, the user is provided with an immersive, virtualised representation of the data set itself. This proposed system is interactive, where the user can interact with the system in order to explore subsets of the data in the virtual environment. This display aims to give users a clear impression and overview of the data characteristics to be mined. In this way, a view of the data, which reduces complexity, while capturing important information, is created. For example, the data sets of the clusters of clients in our online art gallery may

be visualized within our immerse environment, thus taking advance of the “information richness” of the stereo rendering system.

The following observation is noteworthy. Typically, a data set to be used for data mining consists of hundreds or even thousands of dimensional attributes. Usually, only a few of these attributes are relevant to the data mining problem. One or more data reduction techniques, such as sensitivity analysis, should be applied to the data prior to visualization. Alternatively, a data mining technique, such as classification, may be used to reduce the complexity without loss of information. The data mining algorithm therefore acts as a pre-processor, in order to focus the user’s attention on the attribute that are relevant to the problem being mined [10]. The use of data compression and numerosity reduction techniques may also be considered; whilst care should be taken that this does not lead to loss of important information [6].

7. EXPERIMENTAL RESULTS: THE CRYPT OF SANTA CRISTINA

In order to illustrate the performances of the socket-based stereo system, a high-resolution model of a 9th century Byzantine crypt is rendered. The Santa Cristina crypt was virtualised by the SIBA at the University of Lecce and by the Visual Information Technology Group at the National Research Council [11]. Theophylact painted the oldest frescoes in 959 A.D. The crypt was scanned with a MENSIS SOISIC™ laser scanner and the texture maps were acquired from a set of photographs. Those photographs were mapped on the models with photogrammetric techniques and the model was created with the help of the Polyworks™ software.



Figure 1: Set-up for our mobile virtual environment. From top to bottom: the two video projectors with their polarized filters and the two laptops. The computers are interconnected with a twisted wire.

Our mobile stereo virtual environment is used to visualise the crypt. The system is formed of two Dell Latitude C810 laptops with 512 MB of RAM, a mobile Pentium III™ processor and a GeForce2 Go™ graphic card with 32 MB of texture memory. A twisted pair wire connects the two network cards. A DLP-based video projector Toshiba™ TDP-T1 is assigned to each

computer. A right circular polarizer is put in front of the server’s projector while a left circular polarizer is put in front of the client’s projector. The two projectors are aligned. The procedure takes a few seconds and is based on the projection and alignment of two targets. The video signal is display on a metallic screen in order to ensure polarization preservation. The set-up is illustrated in Figure 1.



Figure 2: Details of a medieval crypt rendered with our virtual environment.

Figure 2 shows an overview of the crypt rendered with our VE. The crypt has 400.000 polygons and 24 texture maps of 512x1024 pixels each. Navigation through the crypt is fluid and can be performed in real time. With more powerful graphic cards like the GeForce4™ or Wildcat 4™ (AGP 8x, 384 MB of memory), it is possible to render the model at a resolution of 1.000.000 polygons. The high refresh rate allows comfortable viewing at high resolution without visual tiredness. The system is entirely mobile and can be deploy in no time in any suitable location. The costs involved are much lower than traditional VR systems. Figure 3 shows a detail view of the Christ Pentecrator surrounded by the Theotocos on the left and an angel on the right. For both Figures 2 and 3, the quality of the rendering should be noticed.



Figure 3: Details of a medieval crypt rendered with our virtual environment. The Christ Pentecrator is shown with Mary his mother on the left side.

8. CONCLUSIONS

The architecture of a new mobile and distributed virtual environment for visualisation and data mining has been described. This environment is configurable via a XML file and

support high-resolution and refresh rate passive stereo and tiled wall display. Further work will involve the application of this virtual environment to data mining and for graphical optimisation.

Data mining will be performed in the framework of the CAESAR™ database. The CAESAR™ database consists of thousands of three-dimensional scans of human bodies in addition to the corresponding anthropometric and statistical data. The objective of the project is to cluster the population in order to generate archetypes that shall be used by the transportation and fashion industry in order to design their products in a more anthropometric and ergonomic way.

Dynamical graphical load balancing will allow exceeding current limitation of graphical processing units. This may be achieved by distribution the graphical load and updating the communication scheme among the units in a way that would optimise the rendering.

We would like to thank Sabry El Hakim and Angelo Beraldin, from the National Research Council, Canada for their precious collaboration. We would like also to thanks the SIBA, at the University of Lecce, Italy for their inestimable collaboration with the crypt and the equipment.

9. REFERENCES

1. E. F. Churchill et al. (2001). Collaborative Virtual Environments, Berlin: Springer Verlag
2. Java 2 / Java 3D / JCE: <http://www.javasoft.com>
3. Multiple authors (2000). Special Issue on Large Wall Displays, IEEE Computer Graphics and Applications, 20 (4).
4. S. Singhal et al. (1999). Networked Virtual Environments: Design and Implementation, Addison-Wesley Pub Co.
5. Digital Light Processing <http://www.dlp.com>
6. J. Han and M. Kamber, 2001. Data Mining: Concepts and Techniques, Morgan Kaufman, California: USA.
7. R. Kimball et al. (1998). The data warehouse toolkit: expert methods for designing, developing & deploying data warehouse, John Wiley & Son, USA.
8. U. Fayyad, G. G. Grinstein and A Wierse, 2002. Information Visualization in Data mining and Knowledge Discovery, Morgan Kaufman, California: USA.
9. A. K. Jain, M. N. Murty and P. J. Flynn, 1999. Data clustering: A survey. ACM computer Surveys, 21, pp.264-323.
10. H. L. Viktor and N. du Plooy, 2002. Assessing and Improving the Quality of Knowledge Discovery Data, Data Warehousing and Web Engineering, Shirley Becker (editor), IRM Press, Hershey, PA: USA, pp.198-205.
11. J. - A. Beraldin, M. Picard, S.F. El-Hakim, G. Godin, V. Valzano, A. Bandiera, and C. Latouche, "Virtualizing a Byzantine Crypt by Combining High-resolution Textures with Laser Scanner 3D Data," VSMM 2002, 25-27 Sept. 2002, Korea.