I^2-NEXT: Digital Heritage Expo*

Dongpyo Hong and Woontack Woo
GIST U-VR Lab, Gwangju 500-712, S. Korea
{dhong, wwoo}@gist.ac.kr

Abstract

In this paper, we propose the enhanced I-NEXT, called I^2-NEXT, which enables users to interact with virtual objects by tangible objects in immersive networked virtual environment. The proposed system consists of user interaction, immersive virtual environment, and networking interface. Its primary enhancement over the previous work is refinements of the framework for immersive and interactive VR applications by adopting various design patterns. One of the enhancements is the device module which supports a natural user interaction in a virtual environment. For example, the proposed system provides users with tangible objects interaction so that users are able to manipulate virtual objects by touching real objects. The proposed system also supports large scale stereoscopic display through clustering technique. As to show the effectiveness of the proposed system, we have been developing an application for the reconstruction of cultural heritage. Having been through development of the system, we believe that virtual reality technology is one of the promising technologies which enable users to experience realities in digital space. Detailed explanation of each component and system architecture is presented.

Key words: Interaction, Immersion, Virtual Reality, Networked Virtual Environment, Digital Heritage

1. Introduction

With advancement of information technologies (IT), the reconstruction of cultural heritage has become popular over the past few years. Among the IT, Virtual Reality (VR) becomes a promising technology for preserving the cultural heritage sites in digital domain. For example, VR technology enables the participants to experience a full immersion through large scale stereoscopic display and various interactive user interfaces, such as gestures, tangible objects and so on. There have been many research oriented virtual heritage reconstruction systems in order to present new experiences to the public [1-6].

The development and implementation of VR components are coherent to the need for highly trained and specialized engineers in the field of real-time 3D graphics programming, virtual reality and system integration knowledge. Due to these inherences, it is apparent that artists and non-technical users have been kept away from constructing VR applications. Moreover, the amount of time and effort for developers to implement would be considerable. However, such systems are dependent on applications due to the lack of framework [4]. In fact, many VR related toolkits have been reported but they are too general to be utilized in the reconstruction of virtual heritage applications. Meanwhile, one of the advantages in VR applications is that users can interact with abundant 3D contents and the virtual environment. In order to support interactions, VR technology exploits many interaction devices such as wand, tracker, camera, and so on. Nevertheless, most of interactions in the previous systems were focused on navigations [5]. In aspects of visual immersion, a large size stereoscopic display is indispensable in VR applications. For this purpose, multiple pipes supported and high computational powered workstations are used in the large size stereoscopic display such as SGI Onyx systems [1-3]. For all that, one of the recent trends in VR for the large scale stereoscopic display is the clustered personal computers with a low cost instead of high computational powered workstations.

In this paper, we propose an immersive and interactive VR application framework based on I-NEXT [6]. The main enhancement over the previous work is the refinements of framework in which we apply various design patterns. The proposed framework enables not only VR developers but also non-technical users to construct and experiment VR applications. Moreover, we concentrate on a mechanism for rapid VR application developments by simply combining of the VR components. In the proposed framework, we also refine device module for natural user interactions, virtual environment management for visual immersion and simple VR contents, and networking interface for collaborations. Through the proposed framework, developers are able to implement VR applications by only providing VR contents. To support natural user interactions in the proposed framework, we develop the device module which can handle arbitrary imaging devices as well as haptic devices. With the device module, we can separate the development process of user interface devices from the framework. Thus, the

* *This work is supported by GIST and DML*

user interface developers are able to concentrate on their work without considering of complex VR application structure. In addition, VR application developers are able to utilize not only conventional VR interaction devices, but also natural user interfaces such as camera, haptics and so on. The proposed device module enables users to interact with virtual objects through haptic device and tangible objects in immersive networked virtual environment. The other enhancement over the previous framework is visual immersion. For the sake of simplicity, we currently use CAVE library which easily display a stereoscopic virtual environment on a large screen using clustered personal computers. However, we have been implementing this function through VR Juggler which we used the previous version [7] as well as NAVER [8] which supports PC-cluster based VR system.

This paper is organized as follows. In section 2, we present the architecture of the proposed system and its components as well. The experimental setup and results, and discussion will be presented in Section 3 and 4, respectively.

2. System Overview

The proposed I^2-NEXT is composed of user interface, virtual environment management (VEManager), and networking interface components. The main enhancement of I^2-NEXT over the previous version is to provide immersion to users through haptic and tangible objects user interfaces as well as through a large scale stereoscopic display. Figure 1 illustrates the conceptual system architecture of I^2-NEXT.

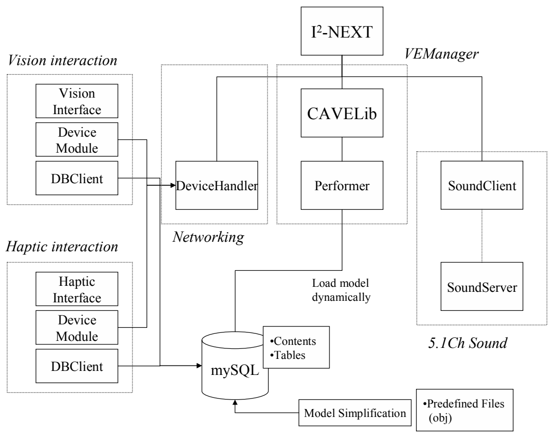

Fig 1. Conceptual System Architecture

2.1 Software Architecture

In the development of VR applications, the amount of time and effort for developers to implement VR applications would be considerable. Therefore, the framework for the rapid VR application development is required. In the proposed system, our main consideration is to make artists and non-technical users construct VR applications if they have VR contents. Thus, we exploit object oriented programming (OOP) mechanism for those who are not familiar with programming because OOP mechanism enables them to implement their own applications at first hand. In order to support OOP mechanism in the proposed framework, we implement all components as classes which are fundamental units in the proposed system. Figure 2 shows the relationships among components of I^2-NEXT.

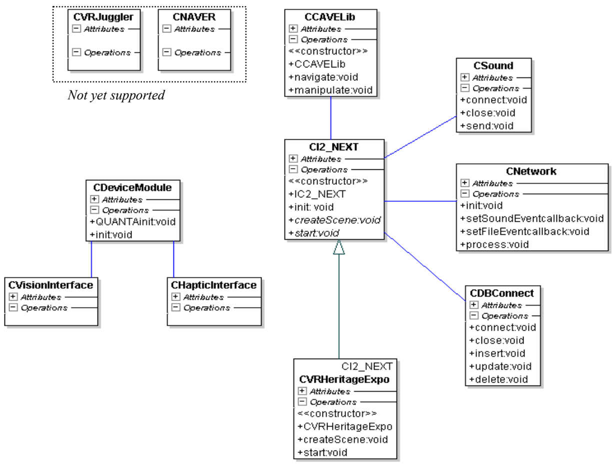

Fig 2. Class Diagram of I^2-NEXT

As shown in Figure 2, developers simply inherit their applications from "*CI2_NEXT*" class and fill out necessary functions without considering the complex system components for the immersive interactive VR applications. For example, if developers want to construct a VR application, then they simply fill out functions like *createScene()* and *start()* in *CI2_NEXT* class. They do not have to struggle with the specific details in the implementation of networking, sound generation mechanism, database connection and stereoscopic display. In this case, however, they have to inherit their VR application class from *CI2_NEXT* class. Meanwhile, artists and non-technical users are able to experiment their VR contents by providing the file name of VR contents in "*CVRHeritageExpo*" class which is subclass of *CI2_NEXT*. I^2-NEXT supplies vision-based and haptic user interface as well as conventional VR user interfaces such as wand and joystick.

2.2 User Interfaces

The user interfaces in I^2-NEXT supports conventional VR interaction interface like wand as well as haptic and vision based interaction interface through device client component. In haptic user interaction, a user is able to experience the generated force from the device when he or she interacts with virtual objects in a virtual environment. Thus, the haptic device affords the user to feel the existence of objects in the virtual environment.

As the vision-based user interface in I^2-NEXT, we adopt vision-based augmented reality (AR) technique for user interaction method rather than gesture-based user interaction in the previous work. We have been developing various methods to provide natural interactions with users such as vision techniques [9]. Through the experiences of vision-based user interface, we believe that AR technique is relatively easy to implement tangible user interface (TUI) [10] because of ARToolkit [11]. Furthermore, AR becomes feasible in real-time interactions. In the proposed vision-based user interface, we combine AR technique and TUI to provide natural immersion to users when they manipulate virtual objects. The difference from the previous version in vision-based user interface is to exploit infrared (IR) camera instead of the 3D camera. It is because lightless environment gives more immersive visual effects to the participants. Thus, we replace the image acquisition device and use retro-reflective materials in I^2-NEXT. Figure 3 shows the proposed vision-based user interface.

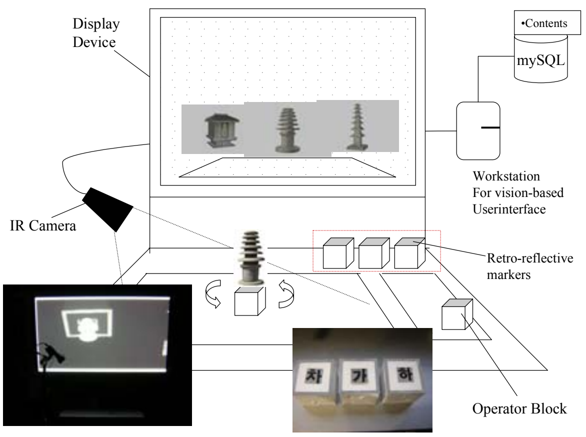

Fig 3. The proposed vision-based user interface

As shown in Figure 3, the proposed vision-based user interface enables users to use various tangible objects wrapped by retro-reflective markers in order to manipulate (rotate, scale, select, etc. shown in Table 1) various Buddha towers. For the proposed vision-based user interface, we implement that various application programming interfaces (APIs) for manipulating ARToolkit in addition to the basic camera manipulation. The overall procedure of the vision-based interactions is similar to the procedure of ARToolkit. However, we provide a set of operations to manipulate virtual objects which are loaded from the shared database in a VR application. For instance, we can load virtual objects from the database using a loading operator block, and then select a virtual object after viewing various virtual objects by rotating blocks. When we select a virtual object, we can locate the selected virtual object into VR environment by map operator block and sending operator block. In this process, the selected virtual object is sent to the shared database through the device client module. In fact, we send the information (model number, location, etc.) of the selected virtual object instead of the actual model data itself. After locating the selected virtual object, we can check the selected virtual object while we are navigating in the virtual environment. Figure 4 shows the interactions process using ARToolkit in the proposed vision-based user interface.

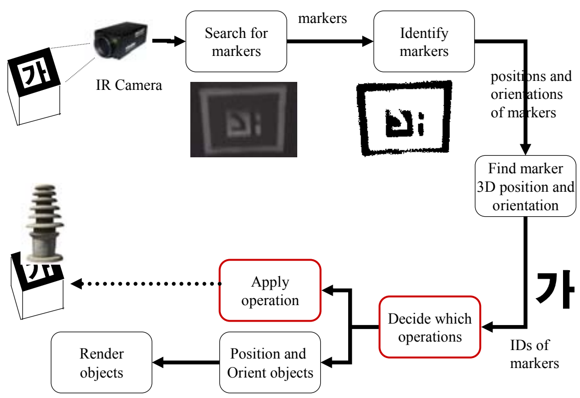

Fig 4. The diagram of interaction processing in the proposed vision-based user interface

Table 1. Tangible operator blocks with makers and the corresponding operations.

marker	operator	features
가	Loader	Lading objects from database
나	Select	Select a virtual object from the loaded objects from database
다	Send	After selection of a virtual object, sends the selected object to database
하	Scale	Scale the selected objects
차	Map	Show the entire virtual environment as a map

2.3 Virtual Environment Management

In I^2-NEXT, virtual environment manager (VEManager) integrates all the information of a local user and a remote user through the networking interface component. Furthermore, it controls a virtual environment to provide users with real-time interactions over the network. Compared to the previous work, the major difference is to support the immersive stereoscopic display with

CAVE library [12]. In order to utilize immersive display functions in CAVE library, we implement *CCAVELib* class as shown in Figure 2. With *CCAVELib* class, developers are easily able to display stereoscopic VR contents. Meanwhile we enhance several virtual environment management functions from the previous work. For example, we can utilize wand and head tracker as the user interaction device in I^2-NEXT. Head tracker keeps tracking the rotation of viewing direction of a user and wand provides the user with navigation method in virtual environment. In addition, we add other functionalities of VEManager that are object morphing routine, object simplification routine, and 3D sound generation routine. In object morphing routine, we utilize the function of OpenGL Performer™ such as pfFlux and pfEngine. However, we are still facing constraint of morphing from one object to another object because the implemented morphing algorithm requires the same number of vertices, normals, and triangles in the two objects. Through the object morphing technique, we are able to construct dynamic virtual environment. In networked virtual environment, there is a need for simplifying complex VR contents. These VR contents often have material properties such as colors, textures, and surface normals. In object simplification routine, we exploit both vertex and texture information in VR contents through Mixkit library [13][14]. The object simplification routine provides more robust simplified objects after simplification because it utilizes not only vertex but also texture (color) information. In most of the previous VR frameworks, sound effects are less focused as compared to graphics rendering and user interactions. However, it is important to provide sufficient sound effects to the users in order to give more immersion along with visual feedback. In I^2-NEXT, we are able to experience full 3D sound effects in the virtual environment through the sound server and client. We implement sound server by using Windows Media Library. However, we need DirectX to work properly without any error [15].

2.4 Networking Interface

To support real-time interactions in the networked virtual environment, the proposed networking interface component has to share the state information of both users and virtual environment. For the sake of simplicity of the implementation, we exploit QUANTA library [16] as the same as the previous work. In I^2-NEXT, the networking interface is composed of several servers and clients components as shown in Figure 5. Device client module is imported in both haptic and vision-based user interface components to deliver the state information of interactions to the virtual environment. After each client send out its state information, Reflector server intermediates the delivered state information between the remote virtual environments. In each virtual environment, VRE module is embedded to process the delivered state information through the registered callback functions. To keep processing the transmitted state information, we adopt thread mechanism in Device server and Reflector client.

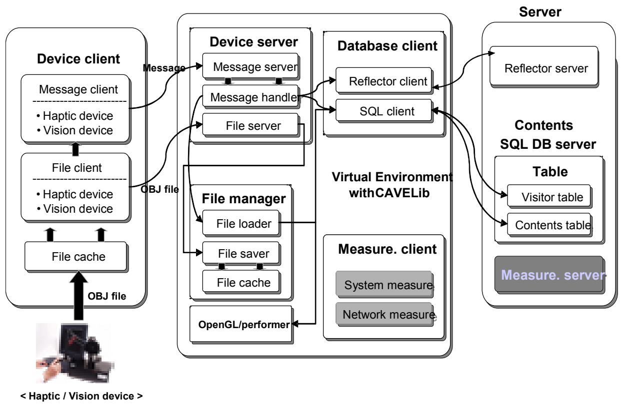

Fig 5. Flows of Message and Events in I^2-NEXT

3. Experimental Setup and Results

To show the effectiveness of proposed framework, we implemented an application of I^2-NEXT where users are able to experience immersive networked virtual reality and interact with virtual objects by vision-based user interface and haptic user interface. Figure 6 illustrates our experimental setup in which we utilized our department facilities such as multimedia class room (MClass) and nomadic meeting room (NMR) because they are already equipped with stereoscopic display and interactive user interfaces.

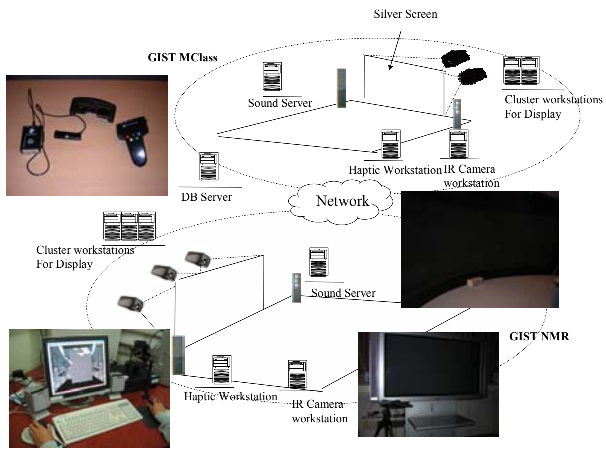

Fig 6. The Experimental Setup

As shown in Figure 6, VR equipments are deployed into the NMR and MClass. For example, the clustered 3 workstations (Dell™ 650) for cylindrical 3D stereoscopic display, 2 workstations (Dell™ 650) for vision interface and haptic interface, and 1 workstation for sound and DB server. In addition to workstations, we also installed IR camera (Fire-i400™) for vision based user interface, Ethernet card for networking interface, wand/head tracker for navigation, and PANTOM® for haptic interaction.

The proposed framework is designed to provide various types of user interaction devices such as vision-based, haptic and tracker-based user interface. Moreover, the proposed framework allows users to travel through virtual environment while they are watching full immersive VR contents. Figure 7-9 shows that users interact with virtual objects and virtual environment by using various types of user interface devices in an application of I^2-NEXT.

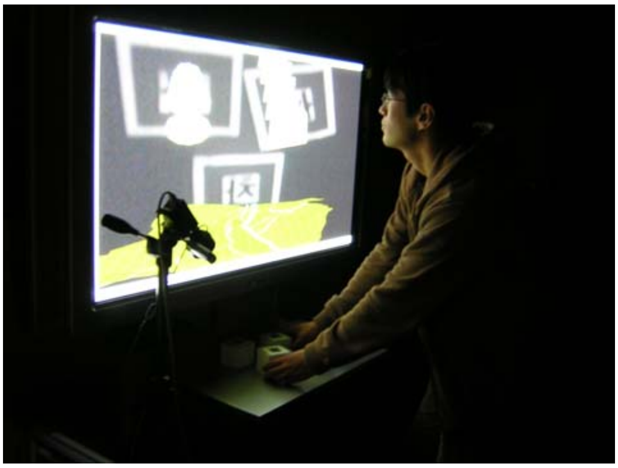

Fig 7. Vision-based Interaction

As shown in Figure 7, users can select a proper virtual object by manipulating operator blocks. Through this experiment, we noticed that participants were amused while they were manipulating tangible objects within augmented virtual objects. As we mentioned, we utilized IR camera and retro-reflective markers because lightless environment gives users more immersion.

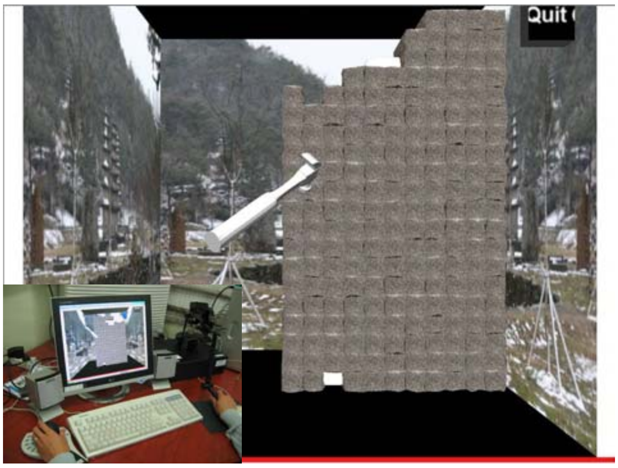

Fig 8. Hatpic Interaction

In haptic interaction, users can experience force feedback with the haptic device when users actually touch virtual objects. As shown in Figure 8, users break the wrapped blocks around a Budda statue using PANTOM® and trackball.

Fig 9. Tracker-based Interaction

As shown in Figure 9, a user who can wear shutter glass with a head tracker and wand is able to navigate virtual environment in a large stereoscopic display.

In fact, the application of I^2-NEXT is designed to provide immersive experiences to the public. At this moment, however, we only conducted experiments on the application in the laboratory. To represent this work to the public, we need a portable 3D stereoscopic display devices and workstations for user interaction devices. Moreover, we need to develop a new type of navigation interface due to the tracker-like navigation devices require expensive infrastructure (INTERSENSE™ IS-900).

4. Discussion and Future Works

In this paper, we proposed I^2-NEXT which enables users to interact with virtual objects by tangible objects and haptic device in the immersive networked virtual environment. The main enhancement of I^2-NEXT over the previous work is immersion. Through the proposed system, users are able to experience immersive interactions and visual feedback. In order to support the immersion, we designed the framework for VR application. As to show the effectiveness of the proposed system, we have been developing an application for the reconstruction of cultural heritage. However, we need to investigate on more natural interactions and experiment on the effectiveness of the proposed system in public places such as exhibitions not in the laboratory.

References

1. Hyun-Jhin Lee, Jinwook Kim, Heekap Ahn, Sang Chul Ahn, Ik Jae Kim, Hyoung-Gon Kim, Heedong

Ko, “VR Experience Design in Tangible Space : Heritage Alive!” In Proc. XVth Triennial Congress International Eronomics Association, (IEA 2003), August, Seoul, Korea.

2. Yong-Moo Kwon, Ig-Jae Kim, Sang Chul Ahn, Heedong Ko, Hyoung-Gon Kim, “Building a 3D Virtual Museum”, In Proc. Virtual Systems and MultiMedia (VSMM 2002), pages 291 - 299, September, Kyoungju, Korea.
3. YongMoo Kwon, Igjae Kim, Sangchul Ahn, Heedong Ko, Hyounggon Kim, “Virtual Heritage System: Modeling, Database & Presentation”, In Proc. Virtual Systems and Multimedia Conference (VSMM 2001) October 2001, Berkeley USA.
4. Meehae Song, Thomas Elias, Wolfgang Müller-Wittig, Tony K.Y. Chan, “Interacting with the Virtually Recreated Peranakans”, International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia, 11-14 February 2003,Melbourne, Australia.
5. Gaitatzes, A., Christopoulos, D., & ROUSSOU, M., “Reviving the Past: Cultural Heritage Meets Virtual Reality.” In S. Spencer (Ed.) Proc. of VAST 2001: Virtual Reality, Archaeology and Cultural Heritage (VAST'01), ACM SIGGRAPH, ISBN 1-58113-447-9, pp.103-110.
6. Dongpyo Hong,Wonwoo Lee, Jaeheon Jeong, Jongwon Kim, Woontack Woo, "I-NEXT: An Interactive Networked Expression eXperience Testbed," Ninth International Conference on Virtual Systems and MutilMedia(VSMM03), pp. 455-462, 2003.
7. Bierbaum, A., Just, C., Hartling, P., Meinert, K., Baker, A., CruzNeira, C., “VR Juggler: a virtual platform for virtual reality application development”, Virtual Reality, 2001. Proceedings. IEEE , 13-17 March 2001. pp89 -96
8. ChangHoon Park, HeeDong Ko, Hee-Kap Ahn, Jinwook Kim, “NAVER : design and implementation of XML-based VR Framework on a PC cluster”, In Proc. Virtual Systems and MultiMedia (VSMM 2002), pages 967 - 975, September, Kyoungju, Korea.
9. D.Hong and W.Woo, "SpaceSensor:Real-Time Gesture Tracking for I-NEXT," International Conferece on Artificial Reality and Telexisitence(ICAT03), paper ISSN 1345-1278, pp. 95-99, 2003.
10. Ishii, H. and Ullmer, B., Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, in Proceedings of Conference on Human Factors in Computing Systems (CHI '97), (Atlanta, March 1997), ACM Press, pp. 234-241.
11. http://www.hitl.washington.edu/artoolkit/download.htm
12. D. Pape, "A Hardware-Independent Virtual Reality Development System", IEEE Computer Graphics and Applications, vol 16, no 4 07/01/1996 - 07/01/1996, pp. 44-47.
13. M. Garland and P. Heckbert. Simplifying Surfaces with Color and Texture using Quadric Error Metrics. In Proceedings of IEEE Visualization 98.
14. http://graphics.cs.uiuc.edu/~garland/software/qslim20.html
15. http://www.microsoft.com/windows/directx/default.aspx
16. Dharanikota, S.; Maly, K., “QUANTA: quality of service architecture for native TCP/IP over ATM networks”, High Performance Distributed Computing, 1996., Proceedings of 5th IEEE International Symposium on , 6-9 Aug. 1996. pp585 -594.