

A Framework for Virtual Reality with Tangible Augmented Reality-Based User Interface

Dongpyo HONG[†], Student Member and Woontack WOO^{†a)}, Nonmember

SUMMARY In this paper, we propose a framework for virtual reality, I²-NEXT, which enables users to interact with virtual objects by tangible objects in immersive networked virtual environment. The primary goal of this framework is to support rapid development of immersive and interactive virtual reality systems as well as various types of user interfaces. The proposed framework consists of user interface for interactions, immersive virtual environment, and networking interface. In this framework, we adopt several design patterns to guarantee that either developers or users (artists) can easily implement their VR applications without strong knowledge of VR techniques such as programming, libraries etc. One of the key features of this framework is the presence of the device module which supports a natural user interaction in a virtual environment. For example, the proposed framework provides users with tangible objects so that the users are able to manipulate virtual objects by touching real objects. The proposed framework also supports large scale stereoscopic display through clustering technique. To realize the effectiveness of the proposed framework, we have been developing an application for digital heritage reconstruction. 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 a digital space. Detailed explanations of each component and system architecture are presented.

key words: tangible user interface, interaction, immersion, virtual reality, augmented reality, framework

1. Introduction

With the rapid progress of technologies in the areas of computers and communications, the future computing environments will support “seamless support from ubiquitous computers and pervasive networking,” that is, users could do just-in-time access to any (invisible) computers at anytime and anywhere [1]–[3]. Consequently, this environment will require users to interact with computers through more natural and comfortable interfaces. In addition, it will be an important issue to deliver the user’s intensions or emotions over the network. In this regard, Virtual Reality (VR) technology is one of promising technologies to provide users with natural immersion and interaction through the network. VR technology, for example, enables to provide a full immersion to the participants through a large stereoscopic display and various interactive user interfaces, such as gestures, tangible objects, and so on. On the other hand, many VR applications have been proposed in various areas to show its effectiveness of realism with computer generated objects or scene. However, we believe that the reconstruction of cultural heritage sites is an appropriate VR application. Fur-

thermore, there have been also many research oriented virtual heritage reconstruction systems in order to present new experience to the public [4]–[8].

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 direct work in VR. 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 [7]. In fact, many VR related toolkits have been reported and commercially available, but they are too big or general for artists or non-professional developers to exploit them [9]–[13]. In addition, well-defined VR toolkits require cost whereas open source VR toolkits still require developers’ efforts and time. Meanwhile, one of the advantages of VR applications is the interaction 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. However, most of the interactions in the previous systems were based on navigations [4]–[8]. In aspects of visual immersion, a large stereoscopic display is indispensable in VR applications. For this purpose, multiple pipes supported workstations with high computational power such as SGI Onyx are used [4]–[6]. However, the recent trend in VR technology for large scale stereoscopic display is being replaced by clustered personal computers with low costs instead of expensive and high computational powered workstations.

In this paper, we propose an immersive and interactive VR application framework based on I-NEXT [14]. The primary goal of this framework is to support rapid development of an immersive and interactive virtual reality system as well as various types of user interfaces. Main contributions over the past work are as follows. In the first place, in the development of VR application, we adopt component-based developing approach, where VR application developers simply follow the sequence of rules in the proposed framework and make virtual objects and environment. In this regard, we concentrate on a mechanism for rapid VR development by implementing VR applications as a combination of components. Therefore, the proposed framework enables not only VR developers, but also non-technical users to construct and experiment VR applications. In the second of place, we also refine device module for natural user interactions by pro-

Manuscript received March 11, 2005.

Manuscript revised June 21, 2005.

[†]The authors are with GIST U-VR lab, Gwangju, Korea.

a) E-mail: wwoo@gist.ac.kr

DOI: 10.1093/ietisy/e89-d.1.45

which is a subclass of *C12_NEXT*. *I²-NEXT* supplies vision-based and haptic user interfaces as well as conventional VR user interfaces such as Wand and Joystick.

2.2 User Interface

The user interfaces in *I²-NEXT* support conventional VR interaction interface like Wand as well as haptic and vision-based interaction interface through the component of device client. 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 virtual environment.

As for the vision-based user interface in *I²-NEXT*, we adopt vision-based augmented reality (AR) technique for user interaction method rather than gesture-based user interaction used in the previous work. We have been developing various methods to provide natural interactions with users such as vision techniques [15]. Through the experience of vision-based user interface, we believe that AR technique is relatively easier to implement tangible user interface (TUI) [16] because of ARToolkit [17]. 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 in VR. The difference from the previous version in vision-based user interface is the exploitation of infrared (IR) camera instead of 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 *I²-NEXT*. Figure 3 shows the proposed vision-based user interface. Table 1 shows the proposed markers and the corresponding interactions.

As shown in Fig. 3, the proposed vision-based user interface enables users to interact with various tangible objects wrapped by retro-reflective markers in order to manipulate (shown in Table 1) various Buddha towers. As shown in

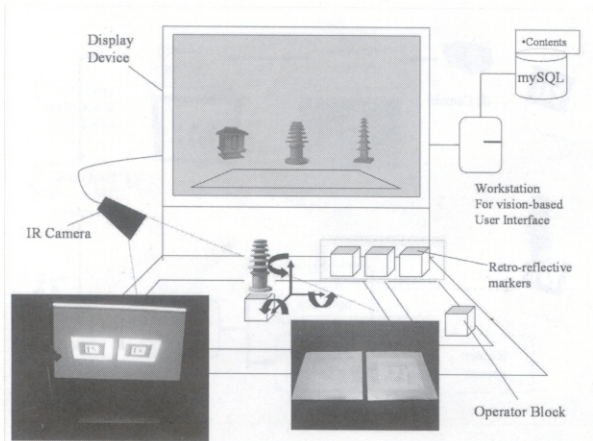


Fig. 3 The proposed vision-based user interface.

Table 1, the proposed user interface provides general interactions in virtual environment such as objects manipulations and navigations. That is, users are able to manipulate virtual objects and navigate in virtual environment through simply rotating or translating the markers from the camera.

2.2.1 Marker Design

In this paper, we exploit both retro-reflective marker and the camera installed Infra-red (IR) emitter and IR filter in order to avoid the interference of lighting effects, which is one of key factors in the limitations of vision-based augmented reality. Figure 4 shows the designed marker which is able to be detected in various lighting conditions.

The designed marker like the one in Fig. 4 (a) is hard for people to recognize with bare eyes. However, the designed marker is easily detected when IR emitter and IR filter are used due to the fact that the marker is made by retro-reflective materials as shown in Fig. 4 (b). Thus, the designed marker is able to be used in various lighting conditions as well as to release the obtrusive marker interface because the markers are obviously visible in general augmented reality. The addressed points enable us to construct immersive virtual environment at any place.

2.2.2 3D Information Retrieval from Marker

The proposed user interface enables us to interact with virtual objects or virtual environments by exploiting 3D information between a marker and the camera, that is, translation

Table 1 Tangible makers and the corresponding interactions.

Marker	Interaction	Operations
	Loader	With this marker, a user can load virtual objects (Buddha towers) into the proposed user interface space from the contents database while he/she is seeing which virtual object is loaded. In this interaction, we simply use the angle of x-axis' rotation and the depth of z-axis's distance from the camera.
	Navigator	With this marker, a user can navigate himself/herself in virtual environment by tilting the marker while he/she is still watching the augmented virtual object.

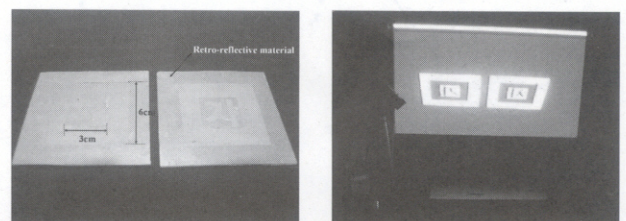


Fig. 4 The designed marker.

and rotation information. Thus, we have to calculate the relative rotated angle and translated distance to each X, Y, and Z axis of marker from the camera. In order to support accurate interactions with virtual objects and virtual environments in the proposed user interface, we have to do basic camera calibrations to extract 3D information between a marker and the camera [18]. In this paper, however, we utilize camera calibration method which is provided in ARToolkit. Through the camera calibration, we can acquire 3D information of a marker from the camera [17]. Figure 5 illustrates the relationship between a marker and a camera in various coordinate systems so that we can infer Eq. (1) from Fig. 5.

In Eq. (1), $R_{3 \times 3}$ is a rotation matrix and $T_{3 \times 1}$ is a translation matrix, respectively. As shown in Eq. (1), the computation of the translation vector from camera to marker is obvious and straight forward comparing to the computation of the rotation matrix of it. Therefore, we need another equation to calculate the rotation of each axis of a marker. With help of Euler's angle representation (Roll, Pitch and Yaw) [19], we can compute the rotated angles to each axis X, Y, and Z from Eqs. (2) and (3).

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_M \\ Y_M \\ Z_M \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_M \\ Y_M \\ Z_M \\ 1 \end{bmatrix} \quad (1)$$

$$R_{3 \times 3} = R_{z,\psi} R_{y,\theta} R_{x,\psi}$$

$$= \begin{bmatrix} c_\phi c_\theta & -s_\phi c_\psi + c_\phi s_\theta s_\psi & s_\phi s_\psi + c_\phi s_\theta c_\psi \\ s_\phi c_\theta & c_\phi c_\psi + s_\phi s_\theta s_\psi & -c_\phi s_\psi + s_\phi s_\theta c_\psi \\ -s_\theta & c_\theta s_\psi & c_\theta c_\psi \end{bmatrix} \quad (2)$$

where $c = \cos$, $s = \sin$

$$\psi = \sin^{-1} \left(\frac{R_{32}}{\cos \theta} \right)$$

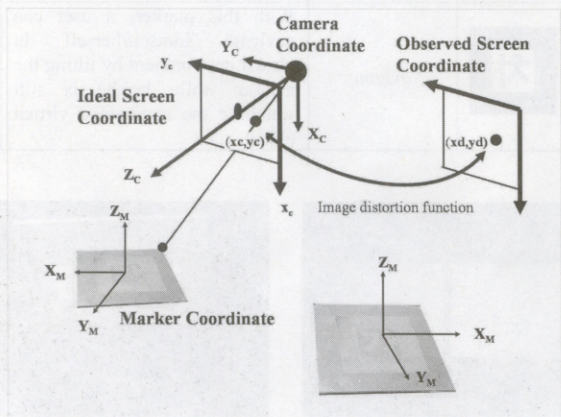


Fig. 5 The relationship between a marker and a camera in various coordinate systems.

$$\theta = -\sin^{-1}(R_{31}) \quad (3)$$

$$\phi = \sin^{-1} \left(\frac{R_{21}}{\cos \theta} \right)$$

In fact, the rotation matrix of $R_{3 \times 3}$ is implemented as *YZY* representation of Euler Angle in ARToolkit. Thus, *arGetAngle()* function provided in ARToolkit is not appropriate to be used in the proposed user interface because we have to compute the rotation for each axis of X, Y, and Z from a marker. As shown in Eq. (3), the proposed method restricts the angle of Y-axis to $-90^\circ < \theta < 90^\circ$ in order to satisfy the constraint of $\cos \theta \neq 0$ in Eq. (3). This reason is obvious because a marker is not recognized by ARToolkit when the rotated angle is either below -90° or over $+90^\circ$. Furthermore, this constraint reduces the complexity of the computation of the rotation matrix $R_{3 \times 3}$ because we do not have to calculate all the components.

2.2.3 Interactions Algorithm

The proposed user interface makes users to interact with virtual objects and virtual environments based on the computed 3D information of relationships between a marker and the camera through Eqs. (1)–(3). Figure 6 shows the overall interaction algorithm in the proposed user interface. As shown in Fig. 6, the proposed interaction algorithm is similar to the augmentation algorithm of ARToolkit, but we modify the flow of interaction when a certain specific marker is detected and recognized.

Figure 7 depicts the situation of selecting a virtual object from the contents database while watching the wanted virtual object.

In Fig. 7, a user can use a *Loader* marker to select the wanted virtual object from Fig. 7 (a) to Fig. 7 (b). This manipulation is done by checking whether both the rotated angle to X-axis of a marker is over the threshold angle and the translated distance to Z-axis from the camera is over the threshold distance. Thus, users are able to manipulate virtual objects or navigate virtual environments while they are

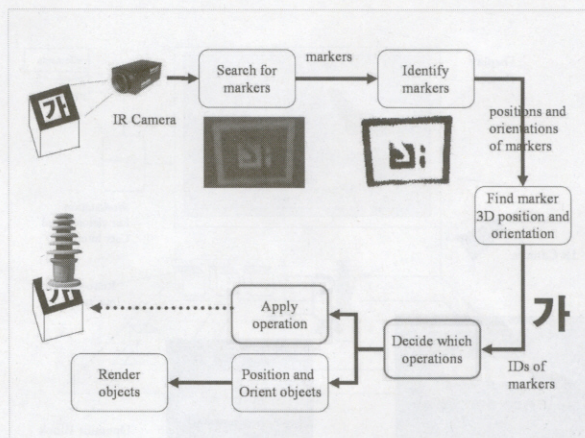


Fig. 6 The diagram of interaction processing in the proposed vision-based user interface.

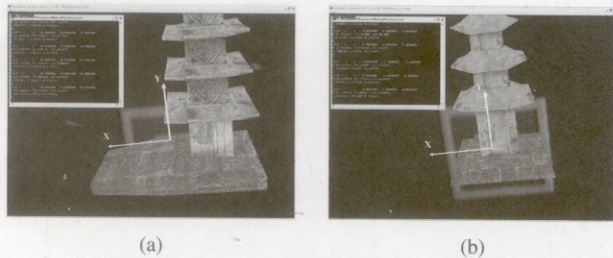


Fig. 7 Virtual object selection by manipulating marker.

still observing the augmented virtual object onto the tangible object.

2.3 Virtual Environment Management

In I²-NEXT, virtual environment manager (VEManager) integrates all the information of a local user and a remote user through the networking interface component and controls a virtual environment to provide users with real-time interactions over the network. Compared to the previous work, the major difference is the ability to support immersive stereoscopic display with CAVE library [20]. In order to utilize immersive display functions in CAVE library, we implement *CCAVELib* class as shown in Fig. 2. With *CCAVELib* class, developers are easily able to display stereoscopic VR contents in a VR application. Meanwhile, we enhance virtual environment management functions from the previous work. For example, we can utilize Wand and head tracker as the user interaction device in I²-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 of 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 [21]. 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 have been 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 I²-NEXT, we are able to experience full 3D sound effects in the virtual environment through a sound server and client. We implement a sound

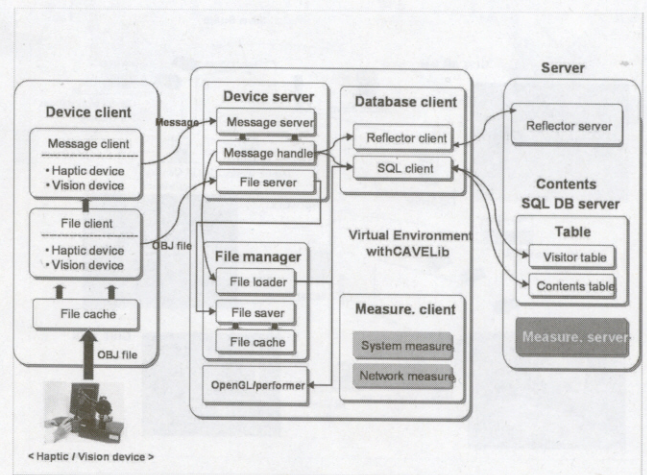


Fig. 8 Flows of messages and events in I²-NEXT.

server by using Windows Media Library. However, we need DirectX to work properly without negligible probability of error.

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 in the networking interface component, we exploit QUANTA library [22], the same as in the previous work. In I²-NEXT, the networking interface is composed of several server and client components as shown in Fig. 8. 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.

3. Experimental Setup and Results

To show the effectiveness of proposed framework, we have implemented an application of I²-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 9 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 displays and interactive user interfaces.

As shown in Fig. 9, VR equipments are deployed into the NMR and MClass. For example, 3 workstations (Dell™ 650) for cylindrical 3D stereoscopic display, 2 worksta-

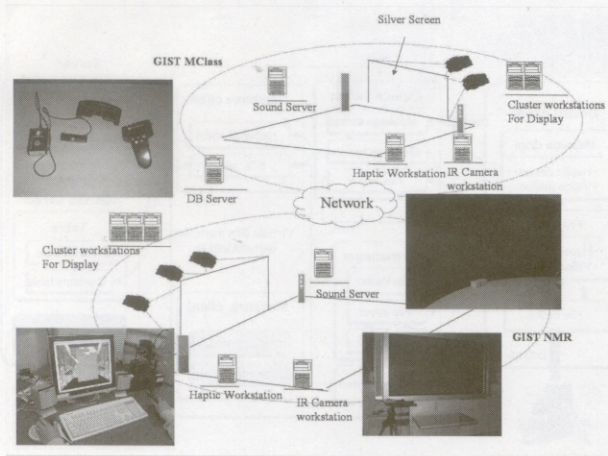


Fig. 9 The experimental setup.

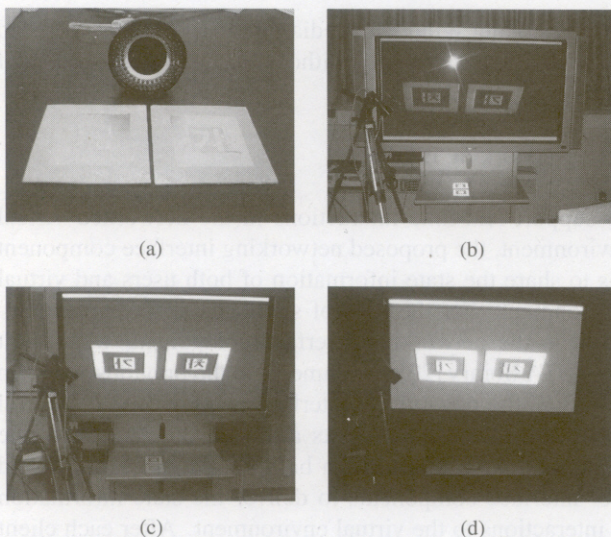


Fig. 10 Marker recognition under various lighting conditions. (a) The used marker and IR filter/emitter (b) bright lighting (c) dimmed lighting (d) lightless.

tions (Dell™ 650) for vision interface and haptic interface, and 1 workstation for sound and DB server. In addition to workstations, we have 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.

Figure 10 illustrates the robustness of the designed marker under various lighting conditions. Thus, the proposed user interface can be invisible but yet tangible. In addition, the user interface can deal with unknown lighting condition dynamically like in public exhibitions.

Figure 11 shows the mean error graph of the rotated angle to each axis of marker.

As shown in Fig. 11, we have experimented on the error of rotation from -45° to $+45^\circ$ because it is useless to check whether the rotated angles to each axis of marker is over $\pm 45^\circ$ in ARToolkit. In fact, the detection is not ro-

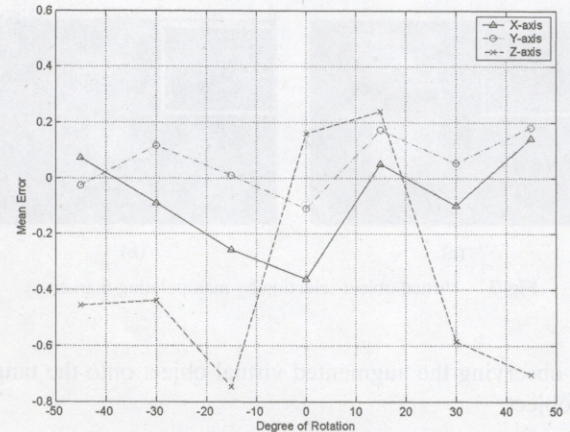


Fig. 11 Mean error graph of rotated angle to each axis of marker.

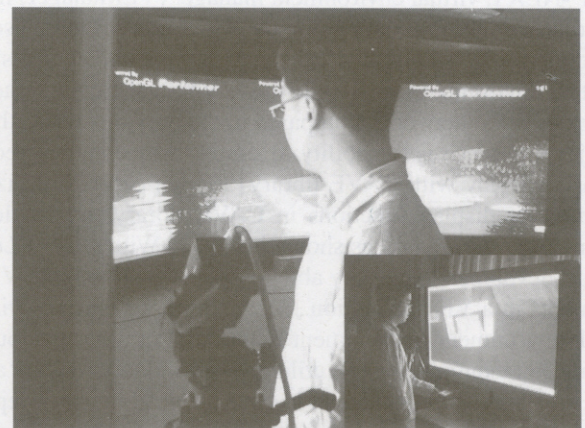


Fig. 12 Interaction with virtual environment and virtual object.

bust over the angles. From the result of the experiment in Fig. 11, the error range of the rotated angle to each axis falls under $-0.75 \leq \text{mean error}(X, Y, Z) \leq +0.24$. Although it reveals that there is some error on the rotation angles to each axis with varying on time, the error rate is relatively low ($\pm 1^\circ$) and the fluctuation is not significant because we use a threshold value based on this experiment. Thus, the result is enough to be applied to interactions like our user interface. Moreover, we found through the experiments that the error is significantly high in a certain range of angles. We figured out that this phenomenon is highly related to the strength of IR emitter and the angle of retro-reflective. However, we could not analyze the error itself for a given $R_{3 \times 3}$ rotation matrix and $T_{3 \times 1}$ translation vector because we had exploited camera calibration which provided by ARToolkit. Thus, we need to develop a novel camera calibration algorithm to calculate more accurate rotation and translation.

The proposed framework is designed to provide various types of user interaction devices such as vision-based, haptic, and tracker-based user interfaces. Moreover, the proposed framework allows users to travel through virtual environment while they are watching full immersive VR contents. Figures 12–14 show that users interact with virtual

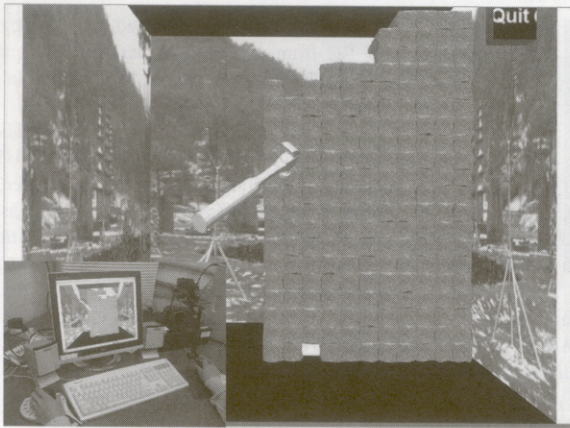


Fig. 13 Haptic interaction.

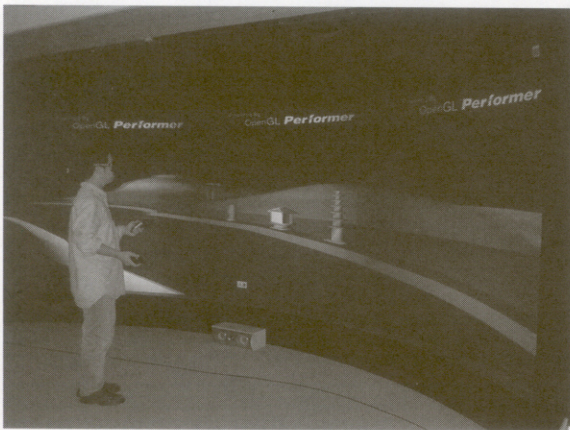


Fig. 14 Tracker-based interaction.

objects and virtual environments by using various types of user interface devices in an application of I²-NEXT.

Users can select a proper virtual object by manipulating tangible operator blocks like shown in Fig. 7. As shown in Fig. 12, they are able to navigate the virtual environment after selecting an interesting one (Buddha tower). Through this experiment, we noticed that participants were amused while they were manipulating tangible objects within augmented virtual objects. As we have mentioned, we utilized IR camera and retro-reflective markers because lightless environment gives users more immersion. Thus, the proposed user interface can be used in an immersive virtual reality system.

In haptic interaction, users can experience force feedback with the haptic device when they actually touch virtual objects. As shown in Fig. 13, users break the wrapped blocks around a Buddha statue using PANTOM[®] and trackball.

As shown in Fig. 14, a user who can wear shutter glasses with a head tracker and a Wand is able to navigate virtual environment in a large stereoscopic display.

In fact, the application of I²-NEXT is designed to provide immersive experiences to the public. At this moment,

however, we have 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 fact that the tracker-like navigation devices required are expensive infrastructures (INTERSENSE[™] IS-900).

4. Discussion and Future Works

In this paper, we proposed a framework for virtual reality, I²-NEXT, which enables users to interact with virtual objects by tangible objects and haptic device in the immersive networked virtual environment. The primary goal of this framework is to support rapid development of an immersive and interactive virtual reality system as well as various types of user interfaces. Through the proposed framework, users are able to have immersive experiences with tangible interactions and visual feedback. To show the effectiveness of the proposed framework, we have also been developing an application for digital heritage reconstruction to show to the public. However, we need to investigate more natural interactions with intelligent virtual objects or environments, called I³-NEXT (Intelligent, Interactive and Immersive VR framework). In addition, we need to experiment on the effectiveness of the proposed framework in public places, such as exhibitions not in the laboratory.

Acknowledgments

This work is supported by GIST and ICU DML.

References

- [1] M. Weiser, "The computer for the 21st century," *Scientific American*, vol.265, no.3, pp.94-104, Sept. 1991.
- [2] A.K. Dey, D. Salber, and G.D. Abowd, "A context-based infrastructure for smart environments," *Proc. 1st International Workshop on Managing Interactions in Smart Environments (MANSE '99)*, pp.114-128, Dublin, Ireland, Dec. 1999.
- [3] S. Shafer, B. Brumitt, and B. Meyer, "The EasyLiving intelligent environment system," *CHI Workshop on Research Directions in Situated Computing*, April 2000.
- [4] H.-J. Lee, J. Kim, H. Ahn, S.C. Ahn, I.J. Kim, H.-G. Kim, and H. Ko, "VR experience design in tangible space: Heritage alive!," *Proc. XVth Triennial Congress International Economics Association, (IEA 2003)*, Seoul, Korea, Aug. 2003.
- [5] Y.-M. Kwon, I.-J. Kim, S.C. Ahn, H. Ko, and H.-G. Kim, "Building a 3D virtual museum," *Proc. Virtual Systems and MultiMedia (VSMM 2002)*, pp.291-299, Kyongju, Korea, Sept. 2002.
- [6] Y.M. Kwon, I. Kim, S. Ahn, H. Ko, and H. Kim, "Virtual heritage system: Modeling, database & presentation," *Proc. Virtual Systems and Multimedia Conference (VSMM 2001)*, Berkeley, USA, pp.137-146, Oct. 2001.
- [7] M. Song, T. Elias, W. Müller-Wittig, and T.K.Y. Chan, "Interacting with the virtually recreated peranakans," *International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, pp.223-228, Melbourne, Australia, Feb. 2003.
- [8] A. Gaitatzes, D. Christopoulos, and M. Roussou, "Reviving the past: Cultural heritage meets virtual reality," *Proc. VAST 2001: Virtual Reality, Archaeology and Cultural Heritage (VAST'01)*, ACM SIGGRAPH, ed. S. Spencer, pp.103-110, 2001.

- [9] WorldToolkit, <http://sense8.sierraweb.net/>
- [10] CAVELib, <http://www.vrco.com/>
- [11] Virtools, <http://www.virtools.com/index.asp>
- [12] A. Bierbaum, C. Just, P. Hartling, K. Meinert, A. Baker, and C. CruzNeira, "VR Juggler: A virtual platform for virtual reality application development," *Virtual Reality*, 2001, Proc. IEEE, pp.89-96, March 2001.
- [13] C. Park, H.D. Ko, H.-K. Ahn, and J. Kim, "NAVER: Design and implementation of XML-based VR framework on a PC cluster," *Proc. Virtual Systems and MultiMedia (VSMM 2002)*, pp.967-975, Kyoungju, Korea, Sept. 2002.
- [14] D. Hong, W. Lee, J. Jeong, J. Kim, and W. Woo, "I-NEXT: An interactive networked expression experience testbed," *Ninth International Conference on Virtual Systems and MultiMedia (VSMM03)*, pp.455-462, 2003.
- [15] D. Hong and W. Woo, "SpaceSensor: Real-time gesture tracking for I-NEXT," *International Conference on Artificial Reality and Telexistence (ICAT03)*, paper ISSN 1345-1278, pp.95-99, 2003.
- [16] H. Ishii and B. Ullmer, "Tangible bits: Towards seamless interfaces between people, bits and atoms," *Proc. Conference on Human Factors in Computing Systems (CHI'97)*, pp.234-241, ACM Press, Atlanta, March 1997.
- [17] <http://www.hitl.washington.edu/artoolkit/download.htm>
- [18] H. Kato and M. Billinghurst, "Marker tracking and HMD calibration for a video-based augmented reality conferencing system," *Proc. 2nd Int. Workshop on Augmented Reality*, pp.85-94, 1999.
- [19] M.W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, pp.45-46, pp.103-105, John Wiley & Sons, 1989.
- [20] D. Pape, "A hardware-independent virtual reality development system," *IEEE Comput. Graph. Appl.*, vol.16, no.4, pp.44-47, 1996.
- [21] M. Garland and P. Heckbert, "Simplifying surfaces with color and texture using quadric error metrics," *Proc. IEEE Visualization 98*, pp.263-269, 1998.
- [22] S. Dharanikota and K. Maly, "QUANTA: Quality of service architecture for native TCP/IP over ATM networks," *Proc. 5th IEEE International Symposium on High Performance Distributed Computing*, 1996, pp.585-594, Aug. 1996.



Dongpyo Hong received his B.S. degree in Computer Engineering from Dong-A University, Busan, Korea, in 2001 and M.S. degrees in Information and Communications from Gwangju Institute of Science and Technology (GIST), Gwangju, Korea in 2004. Now he is a Ph.D. student in GIST since 2004. Research Interest: HCI, Vision-based User Interface/Interaction, Wearable Computing, and VR.



Woontack Woo received his B.S. degree in EE from Kyungpook National University, Daegu, Korea, in 1989 and M.S. degree in EE from POSTECH, Pohang, Korea, in 1991. He received his Ph.D. in EE-Systems from University of Southern California, Los Angeles, USA. During 1999-2001, as an invited researcher, he worked for ATR, Kyoto, Japan. In 2001, as an Assistant Professor, he joined Gwangju Institute of Science and Technology (GIST), Gwangju, Korea and now at GIST he is leading U-VR Lab.

Research Interest: 3D computer vision and its applications including attentive AR and mediated reality, HCI, affective sensing and context-aware for ubiquitous computing, etc.