vConnect: Connect the Real World to the Virtual World

Yifeng He, Ziyang Zhang, Xiaoming Nan, Ning Zhang, Fei Guo, Edward Rosales, and Ling Guan
Department of Electrical and Computer Engineering
Ryerson University
Toronto, Canada

Abstract—**The Cave Automatic Virtual Environment (CAVE) is a fully immersive Virtual Reality (VR) system. CAVE systems have been widely used in many applications, such as architectural and industrial design, medical training and surgical planning, museums and education. However, one limitation for most of the current CAVE systems is that they are separated from the real world. The user in the CAVE is not able to sense the real world around him or her. In this paper, we propose a *vConnect* architecture, which aims to establish real-time bidirectional information exchange between the virtual world and the real world. Furthermore, we propose finger interactions which enable the user in the CAVE to manipulate the information in a natural and intuitive way. We implemented a *vHealth* prototype, a CAVE-based real-time health monitoring system, through which we demonstrated that the user in the CAVE can visualize and manipulate the real-time physiological data of the patient who is being monitored, and interact with the patient.**

I. INTRODUCTION

Virtual Reality (VR) refers to computer-generated Three-Dimensional (3D) environments that allow the user to perceive the virtual objects and interact with them. Depending on the degree of user immersion, the VR systems can be classified into partially immersive VR system and fully immersive VR system. A partially immersive VR system supports the feeling of “looking at” a virtual environment, while fully immersive VR system supports the feeling of “being in” that environment. The Cave Automatic Virtual Environment (CAVE) [4] is a fully immersive VR system. The framework of CAVE consists of three main systems: 1) the projection system, 2) the tracking system, and 3) the computer system. In the projection system, multiple stereoscopic projections are used to project a 3D environment into a room-sized cube [4], which consists of three walls and a floor. The tracking system is used to track the real-time position and orientation of the eyes of the user inside the CAVE. The computer system accepts the inputs of the real-time tracking data from the tracking system, and then controls the outputs of the images from the projectors. The user can perceive the true 3D environment through a pair of stereoscopic shutter glasses, which alternately block the left or right eye such that each eye only sees the corresponding images. The user inside the CAVE can see objects apparently floating in the air, and can walk around them, getting a proper view of what they would look like in reality. Such fully immservie experience provided by the CAVE is not achievable with desktop-based computers.

CAVE systems have been widely used in many applications, for example, 3D simulation and training, architectural and industrial design, medical training and surgery plan, museums and education, oil and gas exploration, scientific research, and topographical analysis. The first CAVE was built at the University of Illinois at Chicago in 1992 [4]. CAVEs have been advanced significantly during the last two decades. The current CAVEs can support a 120Hz refresh rate, a 1:1 aspect ratio, and a resolution up to 1920 x 1920 pixels per facet [1]. Though the standard CAVEs have a very high cost, the mini-CAVEs or micro-CAVEs have a much lower cost in maintenance and ownership, and they can fit into a space as small as a cubicle or small office [1]. The mini-CAVEs or micro-CAVEs can cater for special-purpose applications, such as 3D home theater.

One limitation for the current CAVE systems is that they are typically separated from the real world. First, the CAVE is a virtual environment, and everything shown in the CAVE is not real. The user in the CAVE cannot sense the real world around him or her. For example, an Avatar can be used to represent a person in the real world, and be shown in the CAVE. However, the Avatar is just a virtual object, and it does not has the real-time characteristics, such as body temperature, heart rate, that the person in the real world has. Therefore, the user in the CAVE cannot access the information of the real person. Second, the user in the CAVE cannot interact with the real world. Any instructions generated in the CAVE cannot directly be applied to the real world.

In this paper, we propose a *vConnect* architecture, which aims to establish real-time bidirectional information exchange between the virtual world and the real world. The contributions of this paper are listed as follows. First, we propose a *vConnect* architecture to enable the user in the CAVE to perceive the real world and interact with it. Second, we propose to use finger interactions such that the user in the CAVE can manipulate the information in a natural and intuitive way. Third, we implemented a health monitoring prototype, *vHealth*, through which we demonstrated that the user in the CAVE can visualize and manipulate the real-time physiological data of the patient who is being monitored, and interact with the patient.

The remainder of this paper is organized as follows: Section II discusses the related work. Section III presents the overview of the proposed *vConnect* architecture. In Section IV, we describe information manipulation in the CAVE with finger interactions. In Section V, we present the implementation of the *vHealth* prototype. Finally, the conclusions are drawn in Section VI.

II. RELATED WORK

In terms of what is the most natural interaction with the CAVE, various tools have been proposed and tested.

978-1-4799-2614-5/14/$31.00 ©2014 IEEE

Traditionally, a so called Flystick$^{\text{TM}}$, a wand type remote control is used with various buttons. Individual button and a combo of buttons corresponding to certain actions such as menu selection, directional up/down, etc. Abramyan *et al.* used two types of wands (Nintendo Wii controller and Nunchuk joystick) in angle viewing and manipulation control [2]. Wand was also used in a virtual table tennis game as the hand tracker to mimic the racket in Li *et al.*'s work [6]. Koike and Makino proposed a 3D solid modeling system using wand to draw sketches on the screen, and 3D model was then converted from the basic sketch [5]. Intuitive and natural manipulation in the CAVE system with virtual objects should be as similar as in the real world with tangible objects. Initial works have been proposed in tracking hand and limbs in commanding the CAVE. Kapri *et al.* used marker-based hands and head tracking method for steering-by-pointing in directional control [10]. Virtual ball juggling using hand motion tracking technique was also reported in literature [7].

CAVE is also used in collaborative and distributed environments connected by networks. Schaeffer *et al.* proposed a network interaction in simulating multi-participant physical activities in dancing [9]. In this application, two dancers from two locations worked together in a way that the virtual Avatar of one dancer was created by motion capture. The motion was transmitted from one stage to the other one to form a collaborating dance. A similar example was also created, such that a dancer remotely controls the Avatar with real dance movement to interact with the audience, with a video camera setup to assist her/him in observing the audience response accordingly [12]. Collaborative work on volumetric medical data is also achieved using CAVE system [3]. In this work, a local CAVE user and a remote user, whose Avatar appears in CAVE, were able to collaboratively work on a volumetric model. Roberts *et al.* also proposed a virtual collaboration that the joint effort task is virtually displayed in the CAVE, while three remotely located partners have to cooperate in this collaborative virtual environment [8].

III. OVERVIEW OF THE PROPOSED VCONNECT ARCHITECTURE

The proposed *vConnect* architecture is shown in Fig. 1. It consists of two major components: 1) *cloud server* and 2) *clients*. *Cloud server* is the coordinator which aggregates the information from various sources, processes the information, and then transmit it to the clients. *Clients* can be classified into *bidirectional clients* and and *unidirectional clients*. The bidirectional clients, such as remote desktop users and mobile users, can receive and send information. The unidirectional clients can only receive information, such as actuators, or send information, such as sensors. Clients are heterogenous in terms of power, processing and display capacity. For example, the remote desktop user has a much higher processing capacity than a sensor. The CAVE is a special bidirectional client, which represents the virtual world. The other clients can capture, send, or receive the information of the real world.

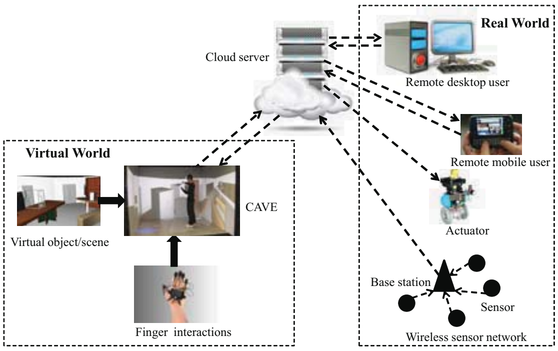


Fig. 1. The proposed *vConnect* architecture with bidirectional information exchange between the virtual world and the real world.

In the proposed *vConnect* architecture, the CAVE receives the real-time information, including the information captured by the sensors and the information provided by the remote users, from the real world. The received real-world information is then visualized in the virtual world, such that the user in the CAVE can perceive it. The user in the CAVE can also manipulate the information, and then make corresponding actions, which can be transmitted back to the real world for executions. For example, the user in CAVE can perform operations on the remote actuators, or place a video call with a remote user.

In the proposed *vConnect* architecture, there are three major technical challenges, including 1) QoS guarantee for real-time delivery of data streams in the wireless networks, 2) QoS guarantee for real-time processing of data stream at the cloud, and 3) natural interactions for information manipulation in the CAVE. In this paper, we focus on *natural interactions for information manipulation in the CAVE*.

IV. INFORMATION MANIPULATION IN THE CAVE VIA FINGER INTERACTIONS

We propose to use finger interactions for information manipulation in the CAVE. In this section, we first define our own specifications on the finger interactions in the CAVE, then justify why finger interactions provide a natural way of information manipulation in the CAVE.

A. Specifications of Finger Interactions in the CAVE

Finger interactions in the CAVE are achieved by the markers worn on both hands of the user. Each marker in the CAVE is tracked in real time by the tracking system. At any time, we can get the 6 Degrees-Of-Freedom (DOF) tracking data in the format of $(x, y, z, \eta, \theta, \phi)$, for any marker. The coordinates (x, y, z) represent the position of the marker in the 3D space, and the Euler angles (η, θ, ϕ) represent the rotation of the marker around its local coordinate system. We use the left marker to represent the left index finger and the right marker to represent the right index finger. That is why we call such marker-based interactions *finger interactions*. The trigger of an action is determined by the positions of the markers and the position of the virtual object to be manipulated. For example, a menu item is selected when the distance between the marker on the right hand and the center of the menu item is less than a threshold.

We define the specifications of five basic actions, which are: 1) activating the main menu, 2) selecting a menu item, 3) moving an object, 4) rotating an object, and 5) scaling an object. The activation of main menu is triggered by the pull-down action performed by the right hand. The selection of a menu item is trigger by touching the menu item with the right index finger. The moving action is controlled by the midpoint between the two markers. In other words, the object is moved along a path which is parallel to the moving path of the midpoint. The rotating action is determined by four factors: *rotation plane*, *rotation axis*, *rotation direction*, and *rotation angle*. The object manipulation is performed in a discrete-time manner. The *rotation plane* is determined by the two intersected lines: the line (denoted as L_c) passing through the two markers at the current time, and the line (denoted as L_p) passing through the two markers at the previous time. The *rotation axis* is the line perpendicular to the rotation plane and through intersection point. The *rotation direction* is the same to the rotation direction of the two markers. The *rotation angle* is the angle through which line L_p is rotated to coincide with line L_c around the rotation axis along the rotation direction. Scaling refers to *uniform scaling*, which means that the object is enlarged or shrunk with the same *scaling factor* in all directions. The *scaling factor* is defined as the ratio between the distance between the two index fingers at the current time and that at the previous time.

B. Information Manipulation in the CAVE

Compared with the desktop-based 3D visualization, such as Maya$^{\text{TM}}$, CATIA$^{\text{TM}}$, Unity etc$^{\text{TM}}$, CAVE has several advantages in visualizing 3D models. First, the nature of the volumetric data requires a big stage for visualization. For instance, in the automobile design, a pervasive conventional approach is to have a real 1:1 model for examination before finalizing the design. The reason is that viewing a 3D model on desktop-based software is not enough in presenting both overall and detail information at the same time. In the CAVE, this is achievable since the 3D glasses the user wears have a tracking marker so that with physical movement, the 3D projected images will change accordingly. This simulates the natural user interaction with the real model, while desktop-based counterpart is not capable in providing such a platform for viewing the scalable data. Second, CAVE provides an immersive experience, which ultimately presents user a real feeling and better judgement in decision making. This immservie feeling is not achievable with desktop-based computers. Third, CAVE is capable in providing several layers of information, this is especially important for data mining [11] and medical data visualization [3].

The dominant remote controller approach requires a learning process. All the buttons and their functions need to be familiarized with and memorized after a period of usage. On the contrary, finger-based interaction is intuitive and natural, such that a user can manipulate a virtual object just as a real but weightless one. Another advantage to have finger interactions is due to the 3D space the CAVE system provided, compared with a 2D screen on the desktop computer which only allows 2D movement. Thus, 3D position and angle data is easily captured by the tracking system which can support the 3D gesture tracking and recognition precisely and effectively. Last but not the least, users are more engaged with 3D finger interactions such as they are in the real world. Object manipulations with fingers in the CAVE create an illusion as if the user works with real object in reality. This proves to be useful in serious games and patient rehabilitation.

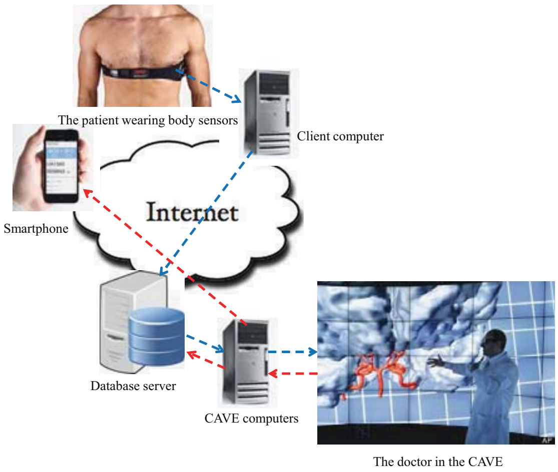


Fig. 2. Illustration of the *vHealth* prototype.

V. CASE STUDY: VHEALTH PROTOTYPE

A. Overview of vHealth Prototype

We developed a *vHealth* prototype to validate the proposed *vConnect* architecture. The *vHealth* prototype is a CAVE-based real-time health monitoring system with the bidirectional information exchange between the virtual world and the real world. The relationship between the *vHealth* prototype and the *vConnect* architecture is clarified as follows. The *vConnect* architecture is a generic framework proposed in this paper. The major contribution of the framework is that it introduces the mixed reality experience in the virtual environment through real-time bidirectional information exchange. The proposed *vConnect* architecture can be applied in a wide range of applications, such as entertainment, tourism, education, healthcare, and others. The *vHealth* prototype is an application of the *vConnect* framework in the healthcare field.

The *vHealth* prototype is illustrated in Fig. 2. The patient who is monitored is required to wear a set of body sensors, which collect the physiological parameters including electrocardiogram (ECG), heart rate, respiration, body temperature, posture, and acceleration. The physiological data of the patient is transmitted to a client computer, which forwards the data to a database server over the Internet. The CAVE computers, which are the computers used to drive the projectors, can retrieve the physiological data of the patient, and then visualize the data in the CAVE. The doctor in the CAVE can interactively manipulate the information via fingers. The patient information, such as phone number, can be updated in the CAVE, and then stored back to the database. Also, the doctor can send instant messages to the patient's smartphone over the Internet.

The data flows in the *vHealth* prototype can be divided into three sequential steps: 1) collection of real-time physiological data from the real world, 2) visualization and manipulation of physiological data in the CAVE via Finger Interactions, and 3) notifications to the real world. The three steps will be elaborated in the following three subsections, respectively.

B. Collection of Real-Time Physiological Data from the Real World

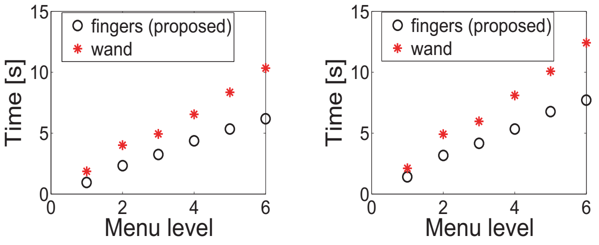
The patient wears a set of body sensors, which capture the real-time physiological data, such as ECG, heart rate, and body temperature. The physiological data is first transmitted to a client computer via wireless channels. The client computer receives the physiological data, and then forwards the data to the database server over the Internet. The database server is used to store the real-time information from the real world. We set up two tables for each patient. One is the health table which records the real-time physiological data. The other one is the personal information table which contains the patient's personal information such as weight, height, date of birth, contact number, and the name of the physician who takes care of the patient.

C. Visualization and Manipulation of Physiological Data in the CAVE via Finger Interactions

In the *vHealth* prototype, the user in the CAVE can perform a series of information manipulations, including information retrieval from the database, information visualization and processing in the CAVE, and information feedback to the patients. All information manipulations are done with finger interactions. In this section, we will first present the performance evaluation on finger interactions, and then describe the visualization and manipulation of physiological data in the CAVE.

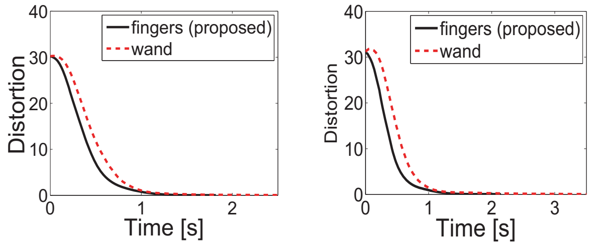
1) Performance Evaluation on Finger Interactions: In order to evaluate the performance of finger interactions, we conducted user tests on a task, which is to *manipulate an object from the initial state to the target state*. The initial state represents the position, the orientation, and the size of the object before any manipulation, while the target state represents the expected state after object manipulations. The test task can be divided into two sub-tasks: 1) menu navigation to select the object, and 2) object manipulations on the object. We conducted user tests to compare the performance between the proposed finger interactions and the conventional wand interactions. In wand interactions, the action of selecting a menu item or scaling an object is performed by pressing the button, the action of moving or rotating an object is performed by moving the wand. We invited two participants to participate in the test. One is an expert, who had been using wand and finger interactions quite frequently. Another one is a novice, who had used wand occasionally and had not used any finger interaction before the test. The novice was trained on the usages of the wand and finger interactions for a half hour before the test. Both participants performed the same test task 20 times with fingers and 20 times with wand.

We first compare the time spent on sub-task 1 between the wand interactions and finger interactions by the two participants. Sub-task 1 is a 6-level menu navigation. Fig. 3(a) shows the average trigger time of each of the 6 menu levels for the operation by the expert. The expert completes the 6-level manu navigation using the proposed finger interactions within 6.19 seconds, reducing the time by 40.1% compared to the wand interactions. The average time for selecting a menu item is 1.03 seconds with the fingers by the expert, which is much lower than that (1.72 seconds) with the wand. The results for the novice are shown in Fig. 3(b), from which we can see that the proposed finger interactions reduce the time by 37.9% compared to the wand interactions. The time for 6-level menu navigation with the fingers by the novice is close to that by the expert, indicating that the proposed finger interaction technique is easy to learn.



(a) (b)

Fig. 3. Comparison of the time for menu navigation: (a) by the expert, and (b) by the novice



(a) (b)

Fig. 4. Comparison of the time-distortion relationship for the manipulations of the object: (a) by the expert, and (b) by the novice

We next compare the time-distortion relationship for sub-task 2 between the wand interactions and finger interactions by the two participants. We use distortion to represent the deviation from the target state. Let $\mathbf{V}$ denote the set of vertices of an object represented by a 3D polygonal mesh. The distortion D_t of the object at time t is represented by the Mean Squared Error (MSE), which is given by $D_t = MSE = \frac{1}{|\mathbf{V}|}\sum_{i=1}^{|\mathbf{V}|}(d_i^t)^2$ where $|\mathbf{V}|$ represents the number of the vertices in the set $\mathbf{V}$, and d_i^t represents the distance between the i-th vertex of the object at time t and the corresponding vertex of the object at the target state. In Fig. 4, we plot the distortion at different time when the object is manipulated from its initial state towards its target state. Fig. 4(a) gives the average time-distortion relationship among 20 tests performed by the expert, while Fig. 4(b) by the novice. We can see that the finger interactions can get a lower distortion at any time compared to the wand interactions by either the expert or the novice. We define the *satisfactory state* as the state of the object with a distortion less than 0.1. As shown in Fig. 4(a), the object manipulated by the expert with fingers can reach the satisfactory state within 1.35 seconds, which is lower by 30.8% compared to that with wand. As shown in Fig. 4(b), the novice can manipulate the object with fingers to the satisfactory state within 1.54 seconds, reducing the time by 51.4% compared to the wand.

2) Visualization and Manipulation of Physiological Data in the CAVE: A vHealth invigilating doctor who is stationed in the CAVE, can retrieve the information from the database

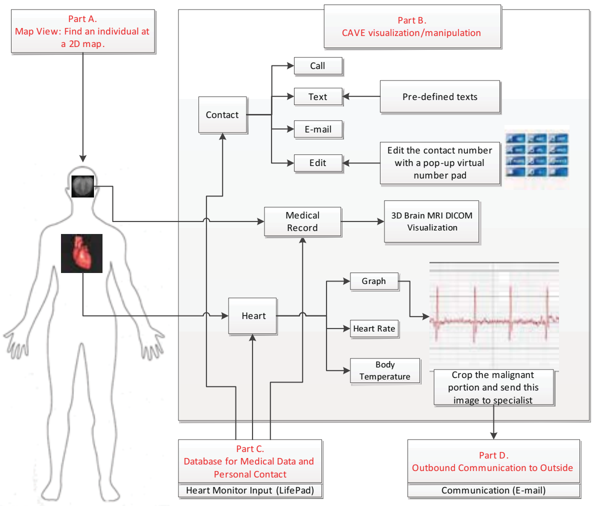


Fig. 5. An overall schematic of the information visualization and manipulation for a specific patient.

server, and then visualize and manipulate it in the CAVE. Fig. 5 presents an overall schematic of the information visualization and manipulation for a specific patient. It has four modules. Part A is a local Toronto map view, which shows the locations of active users who wear Biopac, a bio-sensor hardware collecting real-time heart related physiological data, heart rate, body temperature and ECG data. This map is demonstrated in Fig. 6. At normal monitoring level, every user with normal bio-status record are represented as a green dot. When a user has an abnormal bio status, such as heart rate or body temperature irregularity, his representation dot turns to red. Then an alert monitoring level is triggered and a doctor in the vHeath CAVE is notified.

Part B is a CAVE-based monitoring system. The doctor who is once notified for an alert monitoring level, is able to visualize and process the red dot patient's information in a detail level. He can examine heart rate, body temperature, and ECG data in real-time. The doctor can also access and process patient's personal information and other physiological record to ensure an effective and holistic medical strategy and treatment for this patient. For example, when a patient encounters a cardiac arrest without rapid emergency rescue, his brain is likely to get damaged because of a lack of blood and oxygen delivery. Moreover, it is utterly important for a doctor to have a complete knowledge of this patient's medical record before making a critical rescue plan. Among those physiological data, brain is probably the most important organ. This is because patients who are under the vHealth monitor are probably already have cardiac arrest history and likely brain injury at the same time. The vHeath system offers a visualization platform to examine brain Magnetic Resonance Imaging (MRI) data, by querying the database. This is because, currently, the MRI is the most efficient tool for examining the complicated brain anatomy. Scanning deck of MRI images is obviously slow and not quick enough to make a life-threatening decision. However, CAVE equipment provides an inside-out capability to quickly view the patient's 3D MRI model and catch potential time-bombing medical condition during the emergency rescue. Both the visualization and tracking capacity in CAVE enables the doctor a very quick examination and accurate rescue decision.

Fig. 6. A map monitoring the current locations of the patients.

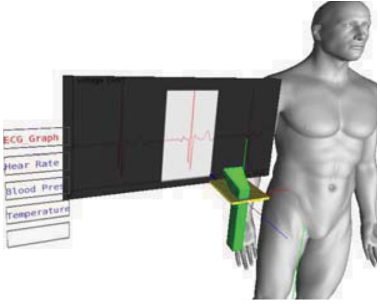


Fig. 7. An operation to crop the abnormal ECG segment and send it outbound via email.

Part C is the database connected with the CAVE system. Input queries from the CAVE user are processed, and fetched data is passed from the database to the CAVE for presentation. Part D is a real-time outbound communication module for the doctor to send alert phone calls, messages, and emails. Currently, we have the email communication set up. The doctor is able to send email alerts to the patient. He is also able to send the abnormal ECG graph to cardiologist, after an image cropping action done in the CAVE.

After the doctor in CAVE choosing the red dot from the monitoring map, an Avatar representing the patient appears in the front of the doctor. The personal information, such as the name and the age, and the real-time physiological data, such as the heart rate and the body temperature, of the patient are shown beside the Avatar. The system also provides a menu selection to visualize and manipulate the information such as personal contact, physiological record, and ECG data.

The real-time ECG signals of the patient can be displayed in a graph, as shown in Fig. 7. The scenario of this scene is that after examining the basic biological statistics, the doctor decides to carefully examine the ECG streaming data of this patient. After recognizing the abnormal ECG segment, the doctor is able to crop that segment using finger interaction, and saves as an image file in the database for record. This image can also be sent to a cardiologist via the outbound email communications for a further diagnosis.

While making a rescue decision for saving the patient's life. the doctor also needs to examine the physiological history of the patient. He can choose the medical record menu. The CAVE then will bring the doctor to visualize the previously stored physiological record. In our scenario, a 3D MRI *Digital Imaging and Communications in Medicine* (DICOM) image of the patient's brain is used for visualization. Fig. 8 depicts

Fig. 8. A 3D brain visualization in the CAVE.

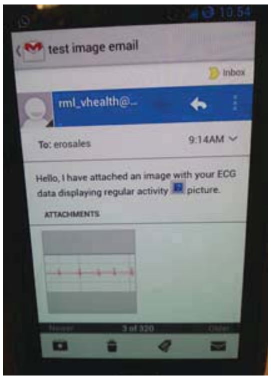
this scene. It can be observed that, with the assistance of the CAVE, the doctor is able to freely examine the 3D brain model without tedious remote control (as if on the desktop-computer). The doctor is also able to zoom in/out, rotate, move the 3D model using remote-control-free hand gestures, and even walk around inside a brain to do a careful examination. Such an approach combines the advantage of the conventional CAVE visualization and the information processing ability provided by the proposed *vHealth* system. This can save a lot of time and help the doctor to make an accurate and fast rescue plan.

D. Notifications to the Real World

In order to ensure that fully immersive interaction between the patient and the doctor is achievable in the *vHealth* system, communication between the doctor and the patient must be achieved. In the *vHealth* prototype, notifications to the real world are done through email. The doctor inside the CAVE is capable of sending the patient an email with text and an image attachment. Thus, without the use of a keyboard or any additional hardware, the doctor inside the CAVE is capable of notifying the patient of any updates or emergencies. Email transfer was implemented by creating a Simple Mail Transfer Protocol (SMTP) instance linked to a public SMTP server. Once a link is made, authentication is done to ensure only authorized users have access to the server. In the process of sending an email with an image, the location of the image must be specified, along with the file name and extension. This automatically adds the image to the email while also allowing the doctor to incorporate text into the email. Fig. 9 shows the text message and the ECG graph received on a smartphone. The email communications enable the user in the cave to interact with the real world.

VI. CONCLUSIONS

In this paper, we proposed a *vConnect* architecture to establish the real-time bidirectional information exchange between the virtual world and the real world. With such information exchange, the user in the CAVE can perceive the real world and interact with it. Moreover, we proposed finger interactions such that the user in the CAVE can manipulate the information in a natural way. We implemented a *vHealth* prototype, a real-time health monitoring system, to validate the proposed *vConnect* architecture. The *vHealth* prototype demonstrated that the user in the CAVE can perceive the real-time information from the real world and provide feedbacks to the real world.



test image email

rml_vhealth@...

To: erosales

9:14AM

Hello, I have attached an image with your ECG data displaying regular activity picture.

ATTACHMENTS

Fig. 9. The text message and the ECG graph received on a smartphone.

REFERENCES

- [1] Christie cave: [online] http://http://www.christiedigital.com/.
- [2] Lucy Abramyan, Mark Powell, and Jeffrey Norris. Stage: Controlling space robots from a cave on earth. In *Proc. of IEEE Aerospace Conference*, pages 1–6, 2012.
- [3] Ali H Al-Khalifah, Robin Woff, Vassil N Alexandrov, and Dave J Roberts. Case study: interacting with volumetric medical datasets in networked cave environments. In *Proc. of IS&T/SPIE Electronic Imaging*, pages 350–359, 2005.
- [4] Heather Creagh. Cave automatic virtual environment. In *Proc. of Electrical Insulation Conference and Electrical Manufacturing & Coil Winding Technology Conference*, pages 499–504, 2003.
- [5] Mikiko Koike and Mitsunori Makino. Crayon a 3d solid modeling system on the cave. In *Proc. of IEEE International Conference on Image and Graphics*, pages 634–639, 2009.
- [6] Yingzhu Li, L-K Shark, Sarah Jane Hobbs, and James Ingham. Real-time immersive table tennis game for two players with motion tracking. In *Proc. of IEEE International Conference on Information Visualisation*, pages 500–505, 2010.
- [7] Minato Mizutori, Koichi Hirota, and Yasushi Ikei. Skillful manipulation of virtual objects: Implementation of juggling in a virtual environment. In *Proc. of IEEE International Conference on Virtual Systems and Multimedia*, pages 79–86, 2012.
- [8] David J Roberts, Robin Wolff, and Oliver Otto. Supporting a closely coupled task between a distributed team: using immersive virtual reality technology. *Computing and Informatics*, 24(1):7–29, 2012.
- [9] Benjamin Schaeffer, Mark Flider, Hank Kaczmarski, Luc Vanier, Lance Chong, and Yu Hasegawa-Johnson. Tele-sports and tele-dance: full-body network interaction. In *Proc. of ACM symposium on Virtual reality software and technology*, pages 108–116, 2003.
- [10] Anette von Kapri, Tobias Rick, and Steven Feiner. Comparing steering-based travel techniques for search tasks in a cave. In *Proc. of IEEE Virtual Reality Conference*, pages 91–94, 2011.
- [11] Eileen Vote, Daniel Acevedo Feliz, David H. Laidlaw, and Martha Sharp Joukowsky. Discovering petra: archaeological analysis in vr. *IEEE Computer Graphics and Applications*, 22(5):38–50, 2002.
- [12] Qiong Wu, Pierre Boulanger, Maryia Kazakevich, and Robyn Taylor. A real-time performance system for virtual theater. In *Proc. of ACM workshop on Surreal media and virtual cloning*, pages 3–8, 2010.