Towards an Immersive Virtual Environment for Medical Team Training

Chang Ha LEE [a,1], Alan LIU [a], Sofia DEL CASTILLO [a], Mark BOWYER [a], Dale ALVERSON [b], Gilbert MUNIZ [a] and Thomas P. CAUDELL [c]

[a] *The National Capital Area Medical Simulation Center*
[b] *School of Medicine, University of New Mexico*
[c] *Department of Electrical and Computer Engineering, University of New Mexico*

Abstract. Many computer based medical simulators focus on individual skills training. However, medical care is frequently rendered by teams. In addition, the conditions under which care is provided can be a crucial factor in training. For example, mass-casualty events can involve the management and triage of large numbers of victims under austere environments. Learning to care for the injured warfighter during combat requires realistic simulation of battlefield conditions. Current simulation systems do not adequately address team training requirements within lifelike environments. This paper describes our work toward the development of an immersive virtual environment that meets these needs.

Keywords. Immersive virtual environment, medical training, CAVE, display wall

Introduction

Traditional methods of medical team training include apprenticeship, rehearsals, and role playing. As medical interns, physicians not only improve their technical skills, but learn the roles and responsibilities of the medical team when treating patients. First responders improve team coordination through repeated exercise and rehearsals. Elaborate scenarios involving many volunteers acting as casualties permit multiple levels of the healthcare system to practice and coordinate their efforts, and to identify weaknesses in operational procedure.

These approaches have several shortcomings. Interns can receive different experiences due to variations in institutional culture. Re-learning may be required when they graduate to become full practitioners. Rehearsals and role playing can require elaborate preparation prior to the event. Realism can be limited. For example, it can be difficult to realistically recreate the kind of damage caused by weapons of mass destruction. Feedback is available only after the event. There is also limited ability to re-run portions of a scenario if errors or weaknesses are identified.

Virtual environments are gaining prominence as simulators for medical team training. They can increase the realism of the training scenario. It has been demonstrated that

[1]Corresponding Author: Chang Ha Lee, The National Capital Area Medical Simulation Center, Bethesda, MD, USA; E-mail: clee@simcen.usuhs.mil.

individuals trained in sterile classroom settings perform poorly when placed in significantly different surroundings [1]. Immersive environments can generate conditions that are difficult or impossible to recreate, such as a mass-casualty event, or a combat zone. They can also be paused in mid-training, and portions repeated to incorporate changes in procedure or to improve team response. Wiederhold and Wiederhold [2] developed a virtual environment for training combat medics, and to provide stress inoculation as a means of preventing or reducing the severity of post traumatic stress disorder. Their system uses game-like environment to help trainees to control fear and anxiety in combat situations. Johnston and Whatley [3] have developed an interactive virtual training system for civilian and military health care practitioners. The system emphasizes experiential learning, and mirrors the complexities and conflicting demands of an operational healthcare facility. Alverson *et al.* [4] developed an immersive virtual environment for network-based medical team training. Their system allows trainees to interact over the network to treat virtual patients with an evolving epidural hematoma following a vehicular accident. Their system facilitates group and collaborative learning. Individuals that are otherwise geographically separated are brought together within the same virtual space. Kaufman *et al.* [5,6] developed an virtual environment for training medical first response in CBRNE (Chemical, Biological, Radiological, Nuclear and Explosive) events. Multiple trainees can collaborate each other and interact with live actors over the network.

While these methods are an improvement over traditional techniques, the level of immersion is still limited. Trainees generally interact with the virtual environment via computer monitors or head-mounted displays. The former provides very little immersive effect, while the latter can be cumbersome. In both cases, trainees generally do not interact directly with team members. In fact, direct visual contact with each other can be impossible with head-mounted displays.

This paper describes our work toward the development of an immersive virtual environment for medical team training. Our system is based on the CAVE [7], a fully immersive display modality that can accommodate teams of individuals. Unlike head-mounted displays, team members can interact with each other in a natural fashion. In addition, equipment and other gear, such as protective clothing, can be carried within the environment, thus improving realism.

1. Methods

1.1. Hardware Components

A CAVE-like system is used to display our immersive environment. A CAVE consists of 3 walls upon which stereoscopic images are displayed. An observer standing in the enclosed space perceives the illusion of being immersed in a 3D environment.

To accommodate larger teams, the adjacent walls in our implementation are angled at 135°. Figure 1 shows our screen setup. Stereoscopic images are displayed using a paired DLP projectors. Passive stereo projection is used. Users wear lightweight polarized glasses to view the scene. To handle very complex scenes and interactive framerates, a scalable hardware configuration is adopted. Each projector is driven by an Alienware Aurora ALX computer, with dual nVidia 7800 GPUs in an SLI configuration. A total of six display computers and six projectors are used for three screens. Additional improvements in display resolution and rendering capability can be obtained by tiling the display to increase the number of rendering computers used. Figure 2 shows the rendering/projection hardware for one screen. Our system also accommodates a 5.1 channel sound system for acoustic presence.

Figure 1. Screen setup.

Figure 2. Projectors with polarized filters for stereoscopic display.

1.2. Rendering

Rendering software generates images for display. The software takes a 3D model of the virtual environment, then generates the virtual scene with appropriate lighting and texture models. Visual effects, such as smoke, and fire are incorporated. Animated elements, such as virtual characters and vehicles, are added for greater realism.

We have adopted Flatland [8] as our rendering platform. Flatland is an immersive 3D environment that permits multiple networked individuals to interact, explore, examine, and manipulate objects in real-time. Flatland was originally designed for use with tracked head-mounted displays. We have adapted the code to run with our CAVE display system.

The use of multiple rendering computers requires the images to be synchronized. Synchronization across all computers is essential for maintaining the illusion of immersion. Even subtle variations in rendering speed between computers can cause unacceptable image jitter for the observer. Two forms of synchronization are required: event synchronization and frame synchronization. Changes in the virtual environment require event synchronization. The movement of virtual characters, smoke, and flame effects must be synchronized across displays. Flatland incorporates mechanisms for event synchronization. In our application, Each display computer runs its own instantiation of Flatland. Events within each instantiation is synchronized across the network so that all events occur at the same time across all instantiations.

In addition to synchronizing events within the virtual world, successive frames for each display must be displayed synchronously. Failure to accomplish this results in unacceptable jitter between images on each screen, and even between images for each eye. We have developed a network frame synchronization algorithm that reduces jitter down to visually imperceptible levels. Our method consists of a server and n rendering clients ($n = 6$ in the current implementation), and takes advantage of the double buffering mechanism in the graphics display pipeline. The server drives the locomotion and scenario events. For each frame, the server sends locomotion parameters and events to the rendering clients r_i ($0 \leq i < n$). The renderers generate the scene and write to the back buffer, then wait for a synchronization packet from the server. When the server sends this packet, all renders swap display buffers simultaneously. To compensate for variations in hardware performance, the server adaptively changes frame rates to ensure all rendering clients are ready to swap buffers before they receive the synchronization packet. While intended to eliminate minor variations in rendering speed across different machines, we have found that the algorithm is sufficiently robust that it successfully synchronizes computers with widely differing CPU and graphics capabilities. Our tests used three rendering computers. One had dual nVidia 7800 GPUs in an SLI configuration, another had an nVidia 6800 GT, and the final computer used a 3D Labs 6100 Wildcat. These three machines represent three generations of graphics hardware. When tested, rendering was synchronized within 50 frames after initialization, and the entire configuration ran at a consistent 18 frames per second.

1.3. 3D Modeling and Rendering

Mass-casualty and combat scenarios require realistic environmental effects, such as smoke, fire, and explosions. We have implemented a fire model using a sprite-based particle system with animated textures [9]. Sprites are instantiated, moved about, and removed based on the parameters such as initial position, velocity, and decay time. Animated textures are displayed on each sprite to simulate the appearance of a fire. This way, we can simulate realistic effects with fewer particles, which results in the performance enhancement. A smoke model has been implemented in a similar fashion.

Comprehensive training requires the development of a suite of scenarios and virtual environments. We have streamlined scenario development by developing a bridge between 3D modeling tools, such as 3D Studio Max and Flatland. 3D environmental models are created in 3D Studio Max, then exported to Flatland. Plug-ins were developed for 3D Studio Max for our smoke and fire models. Information such as the position, direction and intensity of these effects are then used for rendering within the immersive environment. The Cal3D library [10] is used for rendering skeleton based animations. Human characters, vehicles, and other animated objects are modeled by using 3D Studio Max, and exported in Cal3D file format for display in Flatland.

2. Results

Figure 1 shows our screen setup and synchronized rendering results. The wide angle of screens provides a spacious effective area for large teams. Our system provides synchronized rendering across screens. The fire and smoke effects shown in Figures 3 and 4(a) give more realistic presence in mass-casualty or battlefield scenarios. Figure 4(b) shows animated characters.

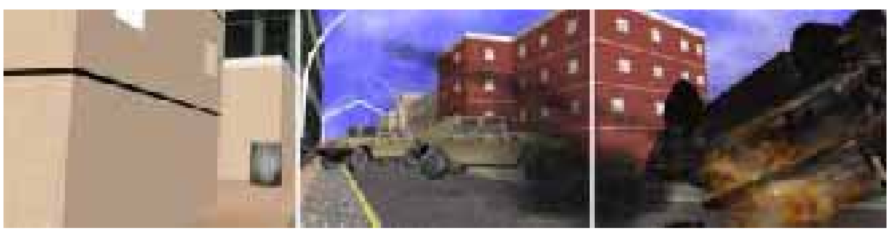

Figure 3. Synchronized rendering across three screens.

(a) (b)

Figure 4. (a) The fire and smoke model adds realism, and (b) an animated running man modeled by using 3D Studio Max is plugged in Flatland.

3. Discussion and Conclusion

We are currently developing an immersive virtual environment for medical team training. A CAVE-like display is used to physically accommodate team members. Our approach differs from systems using computer monitors and head-mounted displays in that all members of the team are in physical proximity, yet are still able to interact within a virtual space. A scalable, network-based rendering approach permits highly complex scenes to be rendered in real-time with minimal temporal mismatch between displays. We have also developed a framework for 3D model development that simplifies the transfer of scenarios between the 3D modeling tool and the immersive environment. Work is currently underway to develop educational case content that will use our immersive environment.

Acknowledgments

This work is supported by the US Army Medical Research Acquisition Activity, Contract W81XWH-04-1-087. The views, opinions and/or findings contained in this report are those of the authors, and should not be construed as the official positions, policies, or decisions of the US Army unless so designated by other documentation.

References

[1] M.W. Scerbo, Weireter L.J., Bliss J.P., Schmidt E.A., and Hanner H. An examination of surgical skill performance under combat conditions using a mannequin-based simulator in a virtual environment. In *NATO RTO Human Factors in Medicine, St. Pete Beach, FL*, 2005.

[2] M. D. Wiederhold and B. K. Wiederhold. Training combat medics using VR. In *Proceedings of CyberTherapy*, 2004.

[3] C. L. Johnston and D. Whatley. Pulse!! - a virtual learning space project. *Studies in health technology and informatics (MMVR14)*, 119:240–242, 2006.

[4] D.C. Alverson, S.M. Saiki Jr, T.P. Caudell, K.L. Summers, Panaiotis, A. Sherstyuk, D. Nickles, J. Holten, T. Goldsmith, S. Stevens, S. Mennin, S. Kalishman, J. Mines, L. Serna, S. Mitchell, M. Lindberg, J. Jacobs, C. Nakatsu, S. Lozanoff, D.S. Wax, L. Saland, J. Norenberg, G. Shuster, M. Keep, R. Baker, R. Stewart, K. Kihmm, M. Bowyer, A. Liu, G. Muniz, R. Coulter, C. Maris, and D. Wilks. Distributed immersive virtual reality simulation development for medical education. *Journal of the Association of International Medical Science Educators*, 15:19–30, 2005.

[5] M. Kaufman, P. Dev, and P. Youngblood. Application of multiplayer game technology to team based training of medical first responders. In *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*, 2005.

[6] M. Kaufman. Team training of medical first responders for CBRNE events using multiplayer game technology. In *Proceedings of Medicine Meets Virtual Reality*, 2006. http://www.forterrainc.com.

[7] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *Proceedings of ACM SIGGRAPH*, pages 135–142, 1993.

[8] T.P. Caudell, K.L. Summers, J. Holten, T. Hakamata, M. Mowafi, J. Jacobs, B.K. Lozanoff, S. Lozanoff, D. Wilks, M.F. Keep, S. Saiki, and D. Alverson. Virtual patient simulator for distributed collaborative medical education. *The Anatomical Record (Part B: New Anat.)*, 270B:23–29, January 2003.

[9] H. Nguyen. Fire in the vulcan demo. In R. Fernando, editor, *GPU Gems*, chapter 6, pages 87–105. Addison Wesley, 2004.

[10] Cal3D – 3D character animation library. http://cal3d.sourceforge.net/.