An Immersive VR System for Sports Education

Peng Song, Shuhong Xu, Wee Teck Fong, Ching Ling Chin, Gim Guan Chua and Zhiyong Huang
Institute for Infocomm Research, Singapore
E-mail:{psong,sxu,wtfong,clchin,ggchua,zyhuang}@i2r.a-star.edu.sg

***Abstract*—The development of new technologies has undoubtedly promoted the advances of modern education, among which Virtual Reality (VR) technologies have made the education more visually accessible for students. However, classroom education has been the focus of VR applications whereas not much research has been done in promoting sports education using VR technologies. In this paper, an immersive VR system is designed and implemented to create a more intuitive and visual way of teaching tennis. A scalable system architecture is proposed in addition to the hardware setup layout, which can be used for various immersive interactive applications such as architecture walkthroughs, military training simulations, sports game simulations, interactive theaters, and telepresence exhibitions. Realistic interaction experience is achieved through accurate and robust hybrid tracking technology, while the virtual human opponent is animated in real time using shader-based skin deformation. Potential future extensions are also discussed to improve the teaching/learning experience.**

I. Introduction

Recently with the proliferation of personal computers, projectors and speakers, more and more classrooms are equipped with multimedia capabilities for teacher presentation and student interactions. With the emergence of networking and Computer Supported Collaborative Work (CSCW), virtual classrooms become popular in the research community in late 1990s. Combined with stereoscopic projected display technology, virtual reality (VR) applications are more attractive for their ability of immersive experience to the users, and thus have been explored [26] since the late 90s.

Winn [24] identifies three types of immersive experience VR would allow but can not be available in the real world. First, VR allows the changes of the user sizes and objects in the virtual environment. Second, immersive VR can make multisensory cues to present information that is not available to human senses in a direct and clear manner. Third, VR allows the creation and visualization of objects and events that have no physical form in the real world.

In most academic areas, the success of a student largely depends on his/her ability to envision and manipulate the abstract information [6]. Scientific visualization, as a sub-area of VR research provides the opportunity to help people recognize patterns, qualitatively understand physical processes, move among different frames of reference freely and manipulate micro-objects easily. The applications of VR in visualization are very useful in many classroom education situations. However, there are not many attempts in applying VR for sports education.

Comparing with sports education in the real world, there are 4 significant benefits using VR technologies:

1) The weather conditions in the real world can not be controlled or easily switched whereas the weather in a VR environment can be manipulated easily in simulation;
2) In a controlled VR environment, the physical properties for playing a game can be simulated and changed, such as force, gravity, ball speed, spin, etc. whereas in a real environment these cannot be changed;
3) In a controlled VR environment, everything can be precisely measured and replayed for sports analysis and repetitive training;
4) In a VR environment, there would be more fun playing with virtual objects that do not exist in the real world such as playing golf with a Tiger Woods or playing baseball with a Teddy Bear.

In this paper, we proposed an immersive VR system for tennis sports education. The gameplay can be recorded and re-play at a later stage for analysis. The user can be tracked on his/her head and racket, and haptic feelings are generated from hitting the tennis ball. More importantly, the user can choose to play with different types of virtual players to train himself/herself comprehensively and having fun at the same time.

II. System Design

We proposed an immersive projection-based display and interaction system. The projection system was composed of two back projected HD Infitec projectors, to provide the user 3D (stereoscopic) viewing with wide angles. A hybrid tracking system was developed to track the user's head and racket movements, while a haptics module was devised to generate force feedback and vibration for the tennis racket. As shown in Figure 1, mirrors are used to reduce the projection space constraint. The hybrid tracking system was composed of ultrasonic-inertial tracking devices and infra-red (IR) cameras and markers.

The motivations of this system design [25] are:

1) To provide the necessary immersive experience with enough viewing angles but not restricting the player's movement;
2) To provide the best stereoscopic and smooth viewing experience in movement, without the nauseated feelings or flickering views when using traditional stereoscopic display technologies;
3) To provide robust and accurate real time tracking for the high speed movement in tennis games;
4) To provide the vibration and force feedback feelings that were mostly missing from most tennis simulations.

Proceedings of the Second APSIPA Annual Summit and Conference, pages 587–592,
Biopolis, Singapore, 14-17 December 2010.

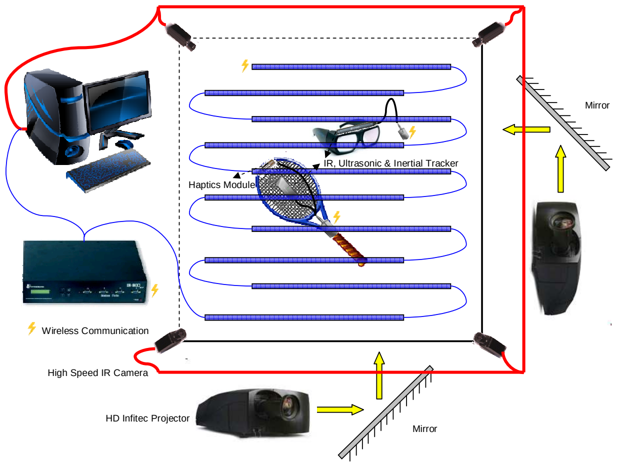


Fig. 1. The Basic System Setup. Two HD Infitec projectors are projecting onto screens with the aid of mirrors; IR, ultrasonic and inertial transmitters are attached to the infitec glasses and the racket; four high speed IR cameras and ultrasonic tracking receivers are overseeing the player's movement area.

To complement this design, a scalable system architecture will be presented in the next section. The key modules of the system will be discussed respectively in its implementation. This paper is the continuous work of [25] with more technical details emphasizing on the software architecture and implementation including display, tracking, animation and haptics. Some new preliminary results will also be presented on the system performance and visual simulation result.

III. Software Architecture and Implementation

Our immersive VR tennis education system was developed based on the scene-graph architecture, and can be easily scaled up to take in more modules when needed. An illustration of our system architecture is shown in Figure 2. We classified the different 3D models under the navigation node, while lighting is specified for the scene separately. In this way, the navigation can be done through a hardware controller to present adjustable views of the scene/simulation for the player.

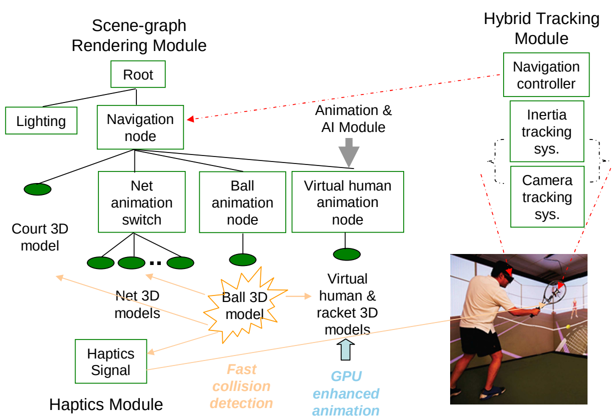


Fig. 2. The System Software Architecture.

Under the broad navigation node, there will be 4 categories of nodes taking care of independent items within the simulation. A 3D court model will be one of the components for animation. A few 3D net models of different shapes are designed under net animation switch to show the continuous movement when it is hit by the ball. Additionally a few other 3D net models are also used to display some gentle net movements when there is a breeze. A 3D ball model is used under the ball animation node. With input from virtual human animation and artificial intelligence (AI) module, the virtual player can be skin-deformed and be able to hit a ball back. More 3D models such as a ball serving machine or watching audience can be added to extend the scene. In addition, these models can be replaced for different educational simulations such as military simulation, and driving simulation, etc.

During our game play, the ball 3D model will interact with other models such as the court, net, virtual player, and the racket through fast collision detection. The fast and accurate collision detection of the virtual ball and the real racket is through our hybrid tracking module, which combines an ultrasonic-inertial tracking system with an optical tracking system. The player's head and racket movements are constantly tracked accurately through this tracking module.

With the stereoscopic display (in section III-A), the player is able to see the ball flying in the air. Together with the haptics feedback on the racket, when the virtual ball is detected to have collided with the real tennis racket through fast collision detection, the player would be able to feel the vibration and force feedback when playing real tennis (in section III-D). As the stereoscopic rendering takes twice the computing resources than a normal 2D display and in order to display super-real animations, a shader-based method is developed in the implementation to speed up the virtual human skin deformation, thus occupying little CPU time to ensure a high frame rate game play (in section III-C1).

High-definition stereoscopic display, real-time hybrid tracking, animation and haptics are the key components in the implementation to make the VR system feels real and differentiate it from the existing immersive simulation systems. In the following sections, these key modules will be discussed with emphasis on the real-time hybrid tracking and animation modules.

A. High Definition Stereoscopic Display

To create the immersive feelings using projection-based displays, stereoscopic technologies are usually used. Traditionally, passive stereo technologies is based on either anaglyph imaging or light polarization. Although it has low cost, high light output and lightweight glasses advantages, the generated ghosting effect inherent with these two technologies degrades the viewing experience. Therefore these two technologies are not suitable for immersive simulation use. Active stereo technology makes use of shutter glasses to show different images from projectors of a different refresh rate for each eye. Besides the cumbersomeness of using battery packed shutter glasses, the user may feel nauseated when moving in a fast speed playing tennis wearing them. In addition, for each display screen, two projectors are needed in synchronization with the computer and the shutter glasses to show frames at different refresh rates, which doubles the cost of using a single projector as in passive stereo technologies.

In our system, we used the latest Infitec stereo technology [10] which splits the color spectrum of each frame with two optical filters for each eye. Only one projector is needed for each display screen. Additionally, there is no ghosting effect for Infitec technology when the user is moving in fast speed, which makes it suitable for our immersive simulation.

B. Robust Real-time Hybrid Tracking

In a traditional immersive environment such as a CAVE which requires 6 DOF tracking, magnetic tracking systems such as Flock of Birds [5], and ultrasonic-inertial tracking systems such as Intersense IS-900 [9] were usually used for viewing and navigation. However, the magnetic tracking quality is severely affected by the existence of metal materials within that environment, and the ultrasonic tracking which provides position estimation in a ultrasonic-inertial tracking system is often unstable, and lost track frequently, especially in the wireless mode. Thus the only reliable fast yet stable trackers are the inertial trackers which provide the 3 DOF orientation tracking data.

Infrared-based optical tracking technology such as A.R.T [1] or PPTH [19] was developed as a vision-based tracking method using multiple IR cameras. Although this technology is sensitive to sunlight or incandescent light, it is suitable for an indoor dark or florescent lighting environment. As most VR simulations are projection-based and situated indoor, this technology can be widely used to provide the position tracking. The orientation tracking result of this technology is not accurate and unstable due to the short IR marker distance for camera triangulation. Since other IR-based tracking methods [1], [17] only run at a low rate (under 100Hz), PPTH is the only high-speed IR tracking technology (175Hz) currently that is capable of tracking fast movement objects such as tennis rackets.

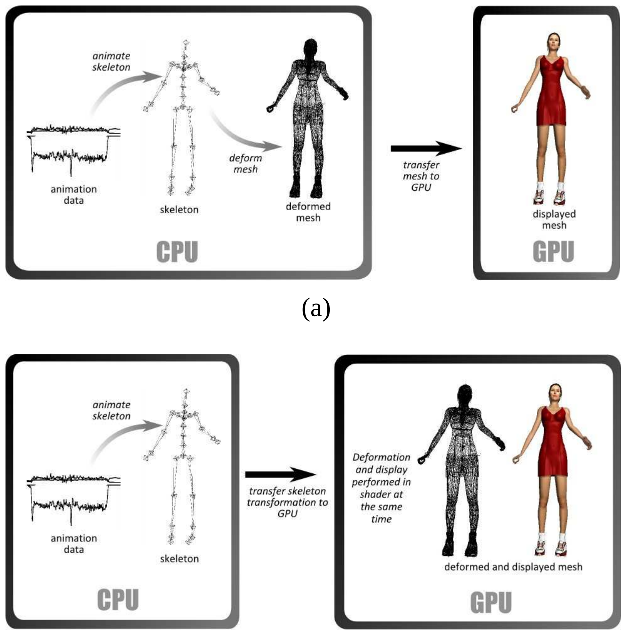
Pure vision-based tracking technology was widely used in augmented reality (AR) [11] and mixed reality (MR) [12] applications. Normally, visual markers need to be provided and the orientation estimation is not very stable or accurate. Some applications integrate vision-based tracking with magnetic/inertial tracking, however, due to the low tracking rate of vision-based tracking technology, this type of hybrid tracking [18], [21] is not suitable to be applied in our system which requires high speed (over 150Hz) tracking.

As optical tracking only provides good quality 3 DOF position tracking data, and may be occluded every now and then due to the inherent nature of vision-based tracking, we use inertial tracking to provide the 3 DOF orientation tracking data and ultrasonic tracking to compliment the optical tracking when occlusion occurred. Sensor fusion algorithms with Kalman filtering [3], [8] was used to smooth the combined position tracking with data inputs and produce more accurate result from ultrasonic and optical tracking. Through this approach, the 6 DOF robust and smooth tracking in real time can be achieved with high speed movements in our system.

C. Animation

In this section, we will first introduce the shader-based skin deformation that has been implemented in our system and is crucial to the real-time rendering of the virtual player. The intelligent animation control will then be briefly described for controlling the virtual player's actions.

1) Shader-based Skin Deformation: The virtual player is driven by data-driven animations; this offers life-like and realistic animations that are still not achievable from procedural methods. In the classical way of animation using data-driven approach, the steps involve animating a skeletal structure using a database of animation data and then deforming the target mesh based on binding information between the skeleton and the mesh. The deformed mesh is then copied over to the graphics card to be rendered, as illustrated in Figure 3(a). This approach is later referred as CPU-based animation.



(a)

(b)

Fig. 3. CPU-based Animation (a) vs. GPU-based Animation (b).

Given a polygon count of roughly 64,000, animating the skeleton and deforming the mesh are achievable under 10 milliseconds. However, transferring over 64,000 polygons' worth of vertex data from CPU memory to graphics memory becomes the bottleneck. As a simple example, 64,000 polygon's worth of data is equal to $64,000 \times 4 \times 3 \times 3 = 1.5$MB. This is assuming the data consists of only vertices and normals, and each floating point value is 4 bytes. We need to send 1.5 MB data from CPU memory to GPU memory every frame. Given PCI-e 16x transfer rate of 4,000 MB/s [16], that would be 1.33 MB per 0.03 seconds, or 30 frames per seconds. Therefore, for a sustainable frame rate of 30 frames per second, the maximum amount of data transferred between the CPU and graphics card should not be more than 1.33 MB. Clearly, transferring deformed mesh data over to the graphics card every frame is not an efficient method of animating a high-polygon count virtual character. Fortunately, programmable shaders allows for mesh deformation to be performed on the GPU side, as illustrated in Figure 3 (b). This approach is called shader-based skin deformation [22], and referred later as GPU-based animation. The GPU is better suited to handle mesh deformation than CPU because it is designed to process lots of vertices very efficiently. It does so by using a large number of parallel processing cores that are specialized for vector (SIMD) operations. For example, the Nvidia Quadro FX 4600 has 112 Processing Cores that are capable of processing 250 million triangles per second [13]. This, in effect, is like using a massively parallel algorithm for mesh deformation. This has the advantages of freeing up more CPU cycles to perform other tasks such as AI, and avoiding the inefficient CPU-GPU memory transfers. For shader-based deformation, only a few changes need to be made:

- Increasing the size of a vertex data to include bone indices and weights (typically an addition of 8 floats per vertex)
- Transferring the animated skeleton to the graphics card in the form of matrices. 1 bone consists of 16 floats, and typical skeletons have less than 50 bones, which would translate to 800 floats per frame.
- The scene graph needs to have shader support. For our application, OpenGL Performer [20] supports GLSL [15], and we can do the deformation entirely in vertex shader.
- The animation library needs to support GPU-accelerated animation. This means that the vertex data has to have the correctly assigned binding information of bone indices and weights, and the animated skeleton has to be in the form of matrices. Havok Animation [7], which we used as our underlying animation engine, has all of these qualities.

In addition, this framework allows more possibilities to be explored:

- CPU can spend more time on application related tasks such as AI or dynamic simulation of ball physics;
- Allows the use of higher polygon count models; without regard to polygon budget;
- Multiple characters may be animated, such as crowd or more playing characters;
- More efficient design in utilizing parallelism between CPU and GPU;
- Moving the whole animation process to GPU through the use of CUDA [4] or OpenCL [14].

2) Animation Control: Our system contains a set of animations related to tennis, such as service, running (locomotion), hitting (shot), idling etc.. To control the animation of the virtual player, a block diagram is drawn to direct the actions of player under different game events. (see Figure 4) When the game starts, one of the service animations will be played. Special events such as waving and walking around the court can also be triggered. More importantly, the virtual player should be able to hit the ball when the opponent returns the ball. To achieve this, a best pair of locomotion and shot animation for hitting a certain target position is chosen for playback. The choice of the best pair is a result of a set of pruning and score functions that eliminate and reward candidates based on their naturalness and logic. When no best pair can be found, a miss animation will be executed. More details will be reported in separate papers due to the space constraint here.

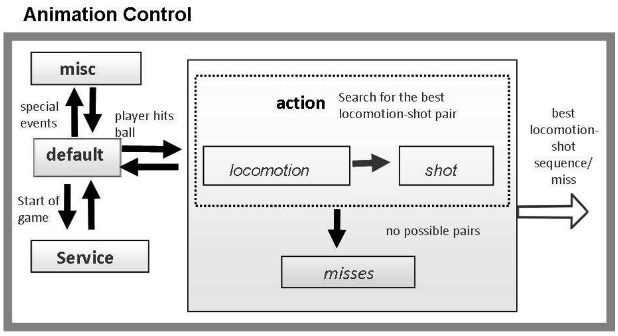


Fig. 4. A Block Diagram of Our Animation Control.

D. Haptics

For an immersive tennis sports training system, haptics module is crucial as proved by the success of Wii remote [23] tennis. However, Wii only provides the vibration feelings, whereas in a real tennis game, the player can clearly feel the force/torque feedback with tactile feelings when he/she hits the ball back. In our implementation, we used a real tennis racket to provide the basic tactile feedback. An actuator attached to the racket can provide the vibration feedback when the virtual ball collides with the real tennis racket. A printed circuit board (PCB) was designed, implemented and integrated in the racket with the USB/RS232 converter output. The vibration solution is the first step towards the 6 DOF force feedback haptics, which will be further investigated at a later stage. More details will be reported in a separate paper due to the space constraint here.

IV. Preliminary Results and Discussion

Our rear-projection 3D display system (a front screen and a right screen at an angle of 90 degrees) has been implemented with each screen of 4.00m wide and 2.25m high. One Barco Galaxy NH-12 DLP projector (12,000 ansi Lumens and 1920 $\times$ 1080 native resolution) [2] is used for each screen to generate Infitec stereo images. The wireless hybrid tracking system is comprised of Intersense IS-900 (ultrasonic and inertial) and WorldViz PPTH (4 IR cameras) to provide robust high speed tracking. A snapshot of a real player in this system is shown in Figure 5.

Fig. 5. A Snapshot of Our Prototype Tennis Simulation System with a Real Player.

The simulation were run on Dual-Core AMD Opteron Processor 2220 (2.8 GHz) with 8 GB RAM and running Windows XP 64-bits. The graphics card is a Nvidia Quadro FX4600. The display resolution is 3840 by 1080 pixels with the scene rendered in stereoscopic display mode. We have conducted two tests to compare the performance of CPU-based animation with GPU-based animation (shader-based skin deformation). In the first test, a high-polygon model of 62,160 triangles and 31,968 vertices was rendered in stereoscopic display mode, and the performance measurements of these two approaches are shown in Table 1. The processing time measures the CPU processing time of that particular application per frame; the rendering time measures how long GPU takes to draw a scene; the total latency refers to the time between the user input and the display of a new image computed from that input, which typically is the sum of the CPU processing time, GPU rendering time, and some additional overhead.

TABLE I
PERFORMANCE COMPARISON OF ANIMATING A MODEL WITH 62,160 TRIANGLES AND 31,968 VERTICES IN STEREOSCOPIC DISPLAY MODE

	Processing Time (ms)	Rendering Time (ms)	Total Latency (ms)
CPU	30	40	70
GPU	0.5	0.7	1.4

Table 1 shows that for CPU-based animation, the total latency is very high. The overhead of transferring vertex data to the GPU is a huge contributing factor. Additionally, the CPU usage is also very high. This would be unacceptable as the CPU is needed for other more crucial computations such as AI and dynamics. Rendering also takes up more time due to re-setting up of triangles every frame. With GPU-based animation, the numbers are dramatically improved: the total latency is down with negligible CPU processing time and rendering time.

In Table 2, the performance comparison is presented between animating a low-polygon model with 2,162 triangles and with 1,083 vertices in the stereoscopic display mode .

TABLE II
PERFORMANCE COMPARISON OF ANIMATING A MODEL WITH 2,162 TRIANGLES AND 1,083 VERTICES IN STEREOSCOPIC DISPLAY MODE

	Processing Time (ms)	Rendering Time (ms)	Total Latency (ms)
CPU	0.9	1.5	2.5
GPU	0.2	0.4	0.7

Though the improvements made by GPU-based animation are not as great, but the savings are evident even for this case.

In Figure 6, animation control results are shown for the virtual player to successfully hit back the returning ball from the real player. When the real player hits a ball, the trajectory of the flying ball can be calculated and the virtual player will run to the ball. The speed and path to the ball will be calculated based on our animation module. When the virtual player reaches to the target position, its pose will change from locomotion to shot, followed by a hit-back action.

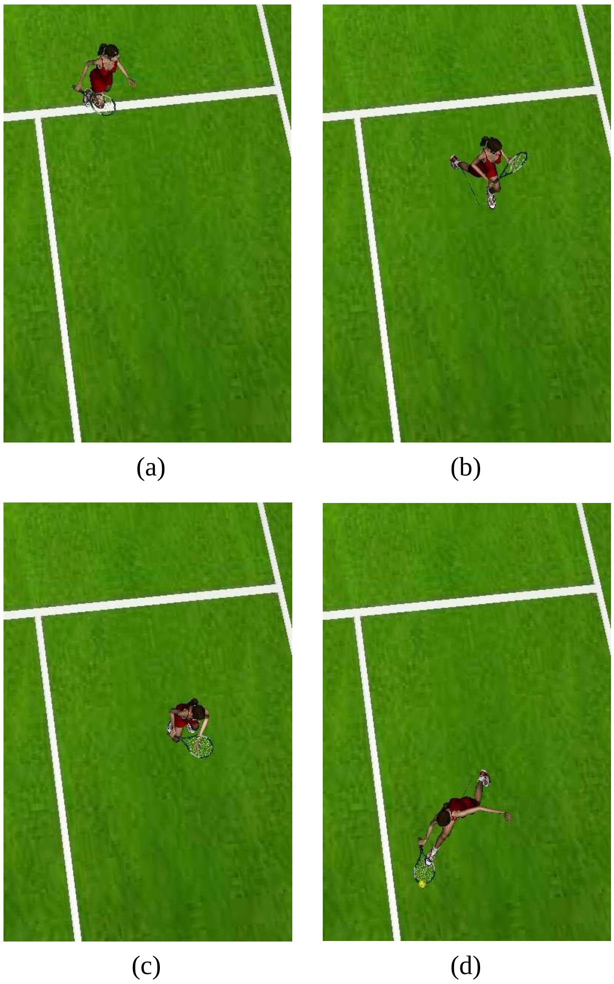


Fig. 6. Animation Control Results: (a) the real player hits the ball; (b) the virtual player runs towards the ball; (c) the virtual player's pose changes from locomotion to shot; (d) the virtual player hits back the ball.

With our scene graph rendering based system architecture, this system can be easily extended to other applications such as military simulations, other sports games and trainings, immersive interactive theaters or telepresence exhibitions with only extra 3D models to be added for customization. Shader based real-time skin deformation, robust high speed hybrid tracking, advanced animation control, and haptics feedback have satisfied the highest requirements of a simulation system such as a military simulation. Therefore, the system can be scaled up or down in terms of both hardware and software to other interactive and immersive applications.

V. Conclusion and Future Work

In this paper, we discussed the benefits of applying VR for education and especially for sports education. Further, we reported design and implementation of an immersive VR tennis education system using stereoscopic display walls. The system architecture is modular and scalable, with software based on the common scene graph and hardware on projector-based system. Through key technologies such as high-definition stereoscopic display, robust and accurate hybrid tracking, shader-based skin deformation, intelligent animation control, and haptics feedback, real-time immersive tennis playing experience has been achieved. With its high performance, the system can be scaled to suit various immersive interactive applications such as architecture walkthroughs, military training simulations, other sports game simulations, interactive theaters, and telepresence exhibitions.

There are still some future work to be done to enhance our VR system. A 6 DOF haptics feedback module needs to be investigated and integrated into the simulation to increase the realism and better measurements of the force feedbacks. Virtual players needs more artificial intelligence to play better with human players. In addition, haptics flooring need to be investigated to give the player feeling of playing on different grounds. Other sensory cues are also needed to be implemented such as airflows to simulate different weather conditions of the tennis court.

VI. Acknowledgement

This work has been funded by the Personal 3D Entertainment Systems Programme, Institute for Infocomm Research, A*STAR, Singapore.

References

[1] A.R.T. http://www.ar-tracking.de/.
[2] Barcho Galaxy NH-12. http://www.barco.com/virtualreality/en/products/product.asp?element=3978.
[3] J. Caarls, P. Jonker, and S. Persa. Sensor fusion for augmented reality. In *Proceedings of EUSAI*, pages 160–176, 2003.
[4] CUDA. http://www.nvidia.com/object/cuda_home.html.
[5] Flock of Birds. http://www.ascensiontech.com/realtime/rtflockofbirds.php.
[6] D. N. Gordin and R. D. Pea. Prospects for scientific visualization as an educational technology. *The Journal of the Learning Sciences*, 3(4):249–279, 1995.
[7] Havok. Havok physics. http://www.havok.com/.
[8] J. Hol, T. Schon, F. Gustafsson, and P. Slycke. Sensor fusion for augmented reality. In *Proceedings of 9th International Conference on Information Fusion*, 2006.
[9] Intersense IS-900 System. http://www.intersense.com/is-900_systems.aspx.
[10] H. Jorke and M. Fritz. INFITEC - a new stereoscopic visualization tool by wavelength multiplex imaging. In *Proceedings of Electronic Displays*, Wiesbanden, 2003.
[11] H. Kato and M. Billinghurst. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *Proc. 2nd International Workshop on Augmented Reality (IWAR)*, pages 85–95, San Francisco, CA, 1999.
[12] NUS Mixed Reality Lab. MXR Toolkit. http://mxrtoolkit.sourceforge.net/.
[13] Nvidia Quadro FX 4600. http://www.nvidia.com/object/io_11761.html.
[14] Open Computing Language. http://www.khronos.org/opencl/.
[15] OpenGL Shading Language. http://www.opengl.org/documentation/glsl/.
[16] PCI Express. http://www.directron.com/expressguide.html.
[17] W. R. Pensyl, D. K. Jernigan, T. C. T. Qui, H. P. Fang, and L. S. Ping. Large area robust hybrid tracking with life-size avatar in mixed reality environment: for cultural and historical installation. In *VRCAI '08: Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pages 1–5, New York, NY, USA, 2008. ACM.
[18] W. Piekarski, B. Avery, B. H. Thomas, and P. Malbezin. Hybrid indoor and outdoor tracking for mobile 3d mixed reality. In *ISMAR '03: Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, page 266, Washington, DC, USA, 2003. IEEE Computer Society.
[19] PPTH. http://www.worldviz.com/products/ppt/ppth.html.
[20] SGI OpenGL Performer. http://www.sgi.com/products/software/performer/.
[21] H. Su, B. Kang, and X. Tang. Research and implementation of hybrid tracking techniques in augmented museum tour system. In *Edutainment '08: Proceedings of the 3rd international conference on Technologies for E-Learning and Digital Entertainment*, pages 636–643, Berlin, Heidelberg, 2008. Springer-Verlag.
[22] J. Weber. Run-time skin deformation. In *Proceedings of Game Developers Conference*, 2000.
[23] Wii Remote. http://wiibrew.org/wiki/wii_remote.
[24] W. Winn. A conceptual basis for educational applications of virtual reality. *Technical Report: TR93-9, University of Washington, Human Interface Technology Laboratory*, 1993.
[25] S. Xu, P. Song, G. G. Chua, C. L. Chin, Z. Huang, and S. Rahardja. Tennis space: an interactive and immersive environment for tennis simulation. In *Proceedings of the 5th International Conference on Image and Graphics*, Xi'an, China, 2009.
[26] C. Youngblut. Educational uses of virtual reality technology. *Institute for Defense Analyses*, 1998.