Joint Platform for the Real-Scale 3D Visualization of Conceptual Design Incorporating Semantic

Gunjeet Hattar, M. Ali Mirzaei, Samir Garbaya, Frédéric Mérienne, Jean-Rémy Chardonnet
Arts et Métiers ParisTech, CNRS, Le2i, Institut Image, France
{*mirzai142.nri,gunjeet.hattar,samir.garbaya*}@*gmail.com*,{*frederic.merienne,jean-remy.chardonnet*}@*ensam.eu*

***Abstract*—The need for a collaborative approach that can easily manifest the basic ideas into a semantic enriched Virtual Environment is felt. To realize the same, a clear combination of two individual processes is applied. Beginning with the manipulation and semantic addition to 2-Dimensional geometric shapes, a concise 3D model is generated. The latter serves as an input to a much interactive system, e.g., a CAVE, for better visualization and interaction with the model. The idea became completed by joining table-touch and real-scale 3D visualization system. However, on question remained unknown "what is the effect of this kind of visualization on an end-user?" This paper tries to find out an appropriate answer to the question only in terms of gender difference. The results show significant difference between female and male populations.**

***Keywords*-Multi-touch table; real-scale visualization; joint-platform; conceptual design and semantic**

I. Introduction

The visualization of a basic idea which strikes to the mind while mixing the idea with the semantics is very interesting to the managers, the designers and the researches. On the other hand, the current demand to reduce the time, cost, and mistakes involved in taking a product from has forced companies to employ the novel emerging technologies in the area of designing. One such technology is virtual reality (VR) [1].

However, the adoption of such on growing technologies with specific application in turn has added an extra complexity to the development process. VR is not such an exception. Complex features such as collision detection, artificial intelligence etc., are being integrated with it, thereby widening the technical gap between the end user and the application programmer.

There have been tools that were presented as an aid to ease the design and development process of VR applications. They were broadly classified into user interface enriching units (e.g., Java3D [2], VRML [3]) and authoring tools (e.g., Blender [4]). However, to visualize the final output with the said tools, the VR designer was required to have a considerable working knowledge of one or more of these tools/software thereby limiting him/her from being an integral part in the design and the development process. The main task of VR however, is to allow users to step through the computer screen into a three-dimensional (3D) world. They can look at, move around, and interact with these worlds as if it is real. It is also the new medium of information and knowledge demonstration, and representing concepts and ideas in ways not previously possible [5]. With the advancement in real-time image processing and cloud rendering technology, and the graphic processors, VR systems could contribute efficiently in various applications. Virtual design is one of the applications of applying VR technology in manufacturing applications [6] [7] [8].

Contribution: a Multi-Touch-CAVE joint platform will be introduced to simplify the design, modification and the real-scale visualization of the products.

Paper organization: the architecture and the development platform of a multi-touch table for small scale visualization will be introduced in the first section. This section will describe how a small scale 3D model is generated from simple concept while the semantics are added to the model. In the next section, a scale one (real-scale) CAVE platform will be introduced. This platform is used to navigate inside and interact with a scale one model. Finally, two platforms are joint to complete the design and real-scale visualization chain.

II. Multi-touch table

Conventional VR development process is a tedious task which caters for unwanted knowledge sharing iterations between the domain expert/end user and the VR application programmer. The latter could be minimized and further enhanced so as to realize the end result on a much interactive platform such as CAVE system (Figure 1). However, before its display on the said complex interactive systems, the approach is realized on an interactive multi-touch table.

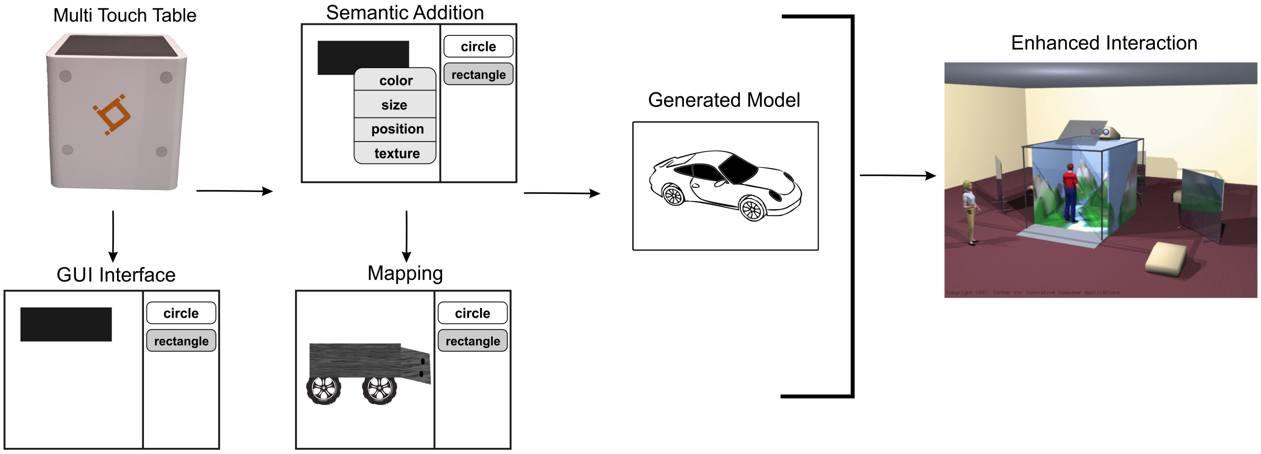

Figure 1: Collaborative VR development environment

III. Development platform for Multi-touch table

A. Knowledge sharing

Initial knowledge sharing is an important process and should be carried out with optimum interaction between the end user and the domain expert as this helps minimize delays and rework [9]. Interviews, data collection and other improvement tools are implemented to achieve the same [10]. The main idea behind this step is to make aware of the limitations and modeling specifications to the end user and a better understanding of the domain to the VR programmer.

B. Conceptual design through GUI approach

The approach begins by introducing the users to a GUI based environment on a multi touch tactile table. The multi-touch serves as a great conceptualize tool to generate ideas, for two or more users (current configuration supports 120 fingers). The users begin by selecting a number of geometric shapes that are two dimensional in orientation. These defined shapes clearly depict the actual VR primitive shapes that will eventually populate the virtual environment. As explained in [11], in order to reach at the final stage of generating a conceptual model it is required to make a number of assumptions and simplifications. This helps to reach a point which results in a believable simulation model since not every aspect of the real world problem can be included into the simulation. Furthermore, for any given simulation model [12], lays down four of the most essential components, i.e., the objectives, that defines the purpose of building the model, the input elements derived from objectives that can be manipulated for better understanding, the output that defines the results obtained and the content that defines the elements involved and their interconnections.

C. Semantic addition

Adding semantics is achieved through a menu, sub-menu system. The end user maneuvers through the system in the touch interface to interactively set the semantics defined for the object. Certain attributes such as color, texture overlay, height, width, etc., can easily be set for any given object [13]. Furthermore, for an individual object the user could set a different set of semantic properties as opposed to the ones already defined for a group of similar objects. He/she could also enter the exact values for the object's semantic properties in the given value text boxes in addition to interactively manipulating it using the touch gestures defined for the object. Semantics for a simulation model can also be defined through various concepts, i.e., objects and their relationships. These again, are easily accessible using the GUI environment through the selection of two objects and defining semantics for one with respect to the other selected object.

D. Mapping

The purpose of mapping is required to bind the two-dimensional shapes and their respective semantics, defined during the GUI interaction process onto the actual VR primitives. To achieve the same the use of ontologies is incorporated [13]. As described in [14] ontology provides a pedantic abstraction of a given structured vocabulary usually extending to the suited domain knowledge. The use of an existing domain ontology or an implicitly created one with the help of the domain expert are both suitable requiring the mapping of the source onto the target. Having adopted the mapping technique used in [14] which defines the use of three different types of mapping, i.e., meta, domain and world, we can ensure that the objects are firstly mapped from the conceptual vocabulary onto the one defined by a formal VR specification, moving onto the mapping of domain concepts onto VR object type and finally defining a mapping to alter the state of default domain mapping.

E. VR generation

Now that the concepts and their relationships have been defined and that they are mapped onto the VRML primitive type, a VR environment can be easily generated. An automatic conversion program generates the appropriate code for the VR application. The final code can be obtained in different files formats, e.g., an *.obj file. This file can thus be fed into the CAVE system and used for further enhanced interactivity [2].

IV. Hardware architecture and development platform for real-scale 3D system (CAVE)

Real-scale virtual-reality system was designed to visualize any kind of 3D model with the real dimension. The 3D models can be the model of an industrial product (car, engine), building, urbane landscape, and so on. The aim of the system is to facilitate the interaction with the model while changing some properties of the model such as color, texture, material, lighting and sound. Moreover, the designer should be able to navigate inside and inspect different parts of the model such that can modify part of the design or correct the error before the fabrication and the construction. This decreases the cost of design and implementation consequently. More detailed system architecture is shown in Figure 2. It is made up of navigation/interaction devices, VRPN (Virtual-Reality Peripheral Network) server, measurement unit, mapping unit, cloud computing and graphic engine, head tracking and display subsystems. These subsystems construct the skeleton of the system; however there is another part which uses the bio-feedback to study the effect of immersion in virtual environment on the end-user.

A. Devices and interfaces

VRPN, which is proposed and implemented by Russell M. Taylor [15], has been used to connect the infrared cameras of head tracker system, other navigation and interaction peripherals to the system. So far, Kinect Xbox 360, Fly-stick, game-pad, mouse, keyboard, 3D mouse, and 5DT glove has been connected and tested successfully. Among the devices, Kinect provides sound (using array of microphones) and video (infrared and RGB) resources. A new navigation/interaction metaphor was developed by using speech processing [16] along with a dictionary (for example: “turn to the right/left”, “move forward/ backward”, “show menu”) to map words to action. Infrared images provide information about the depth which is used with RGB image for gesture and locomotion analysis. VRPN server developed by [17], widely known as FAAST, were employed to extract the skeleton and record the different signals for further gesture analysis. FAAST VRPN server provides the coordinate and orientation of 24 joints of the user’s body. This information helps design very complicated interaction and navigation tools based on user gesture. For example, the person can walk inside the virtual environment while maintaining his given position and moving the 3D scene. The platform provided a secure way to stream information from iPhone and iPad to the VRPN network through the wireless network. In addition to any other devices (e.g., iPhone/iPad, Smart phone, laptop, Mac) with a UDP connection can connect and stream data over this infra-structure.

B. Measurement and parametric mapping function

Device and the VRPN server only provide streaming of the data from the devices to the client application and not the action. Measurement unit makes a bridge between server data and mapping unit. Mapping unit is a parametric function which specifies appropriate action to the analog signal (after the signal processing was completed), the orientation and the position tracker, and the digital buttons. Any action requires some inputs and bias values to be set up to function well. For instance, traveling inside the model requires speed control and acceleration adjustment. Moreover, the model should be re-projected in the new position of the user. For that, appropriate messages should be sent to the graphic engine and other parts of the system. All these tasks are managed in the mapping unit.

C. Display system

A development platform called PeTRIV was developed to manage the connection and data communication between the VRPN network, head tracking system, and display projectors. The platform uses OpenSceneGraph on the top of OpenGL to render a 3D model. Then the model is projected into the display system by MPI and four NVidia Quadroplex GUPs. The display system consists of four walls, 2 projectors per wall (one image per eye, totally two images per face is generated for projection) [18]. The system uses laser triangulation to visualize real-scale virtual immersive 3D. An infrared head tracking system (ARtracker [19]) is established in a multi-view configuration to find the user location. PeTRIV uses java-scripts for the application development. However, VR-Juggler platform [20] was tested on this architecture too. So the user has this possibility to select the development platform.

D. 3D Model plug-in

DTEntity is a game and simulation engine using OpenSceneGraph for visualization [21]. DTEntity employs different entity systems (or component systems) to manipulate a 3D model and integrate environmental properties such as lighting, sound and a new object in the model. The entity systems are an almost perfect solution to massively speed up the development time for most of the VR applications, and allow almost unrestrained re-writing of fundamental virtual reality features post-launch with very little effort. Devices,

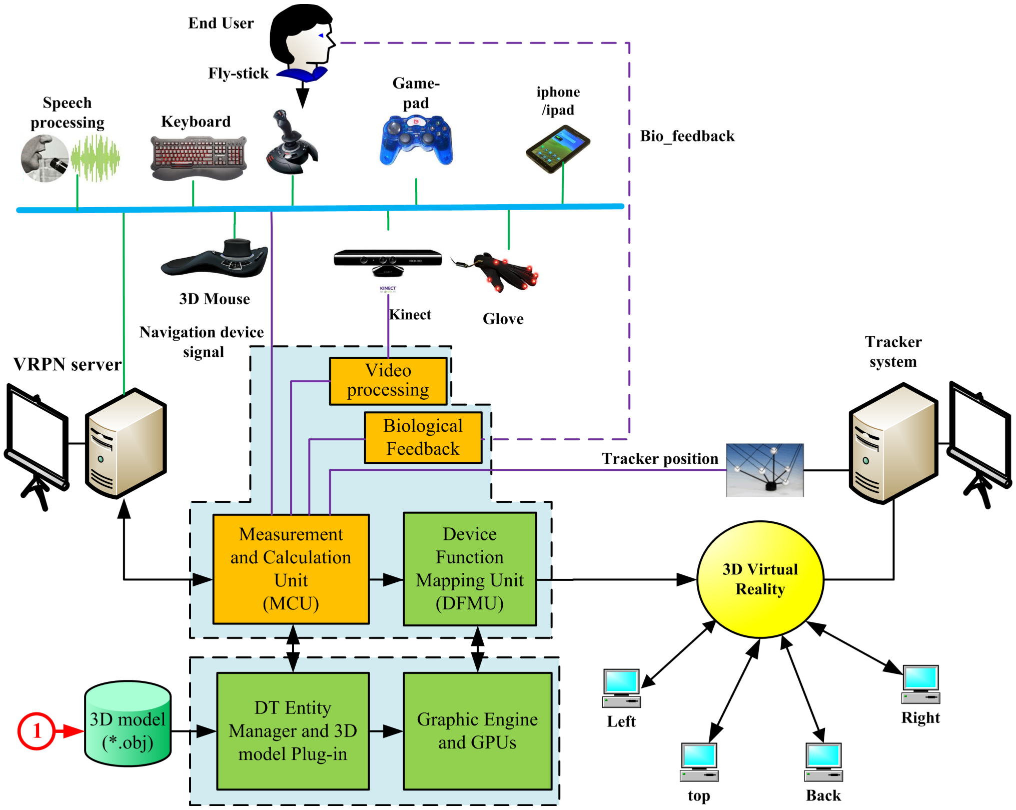

Figure 2: Real-scale (scale-one) visualization system architecture

virtual environment, head tracker, properties of the environment (sound, lighting) have entity system which will be used by entity manage and the client code to have access to the data of that system. For instance, a Map System in the client code is accessible by using "EntityManager.getEntitySystem ("Map")" command. Similarly, a fly-stick can be called in the application by "mapSystem.getEntityIdByUniqueId ("ImmersiveFlystick")".The target 3D model by the third party platform or software (3DMAX) can be plugged in the platform using entity manager. *.AVI and *.OBJ models can be loaded by the plug-in. More information about other 3D file extension is explained in [22]. Plug-in point is shown in Figure 2 by number 1 at the left bottom and highlighted in red.

V. Joint platform and connection protocol

The overall approach is a clear and clever combination of two distinct approaches that if used together could help generate a quick and collaborative virtual reality application whose output can then be used to display the same model in a more interactive system such as the CAVE.

A. *The joint visualization and modification process*

The first part of the approach (Figure 3) defines how conventional virtual reality application development process can be a lengthy and tedious task when it does not involve a collaborative input of the domain expert and the application programmer. To address the latter, a collaborative approach with easy to use GUI based environment is implemented. Following through a series of quick and easy steps the final 3D model can be obtained.

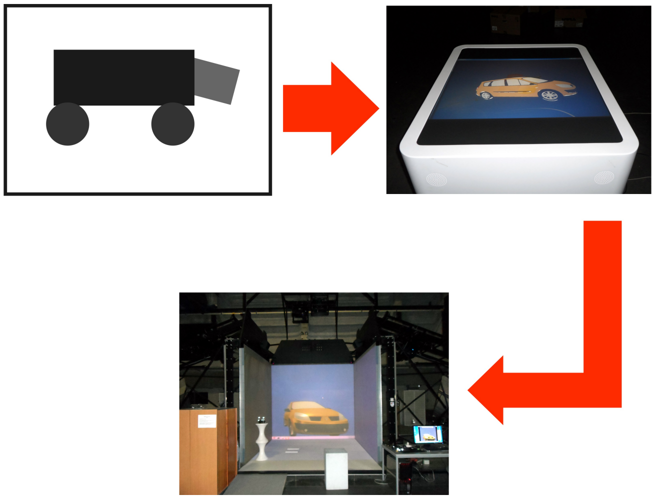

Figure 3: Overall process from concept to real-scale 3D model

The first half of the approach serves as input to the second half. The input model can be plugged in the second platform (CAVE) easily by the DTEntity interface and using common *.OBJ format. A designer, programmer and end-user can interact and see the model in the real scale. If there is a problem in the model or some parts need to be modified the group will come back to the multi-touch table and correct the model and generates a new 3D model. The overall process is shown in Figure 3. Figure 3 shows the application of joint platform in a car design.

VI. Experiment and Result

Since these two platforms can work jointed or independent from one to another only one platform will be studied in this paper. Besides, because the second platform affect the user more than the first one due to full immersion only second platform will be studied. This paper evaluate the effect of immersion inside a real-scale 3D model (the second platform) on how well end-user can work without any cyber-effect (e.g., cyber sickness). For that, 10 subjects were randomly selected (6 males and 4 females: 31.58 ± 12.69 years old, 74.65 ± 15.22 kg). They were informed via university email address and communicator. There was a briefing to give enough information about the test procedure and its objective. All the subjects participated voluntarily in the experiments.

Pre-exposure questionnaire (Q1) was obtained from each of the subjects and each was asked about their backgrounds and health conditions. The result of the questionnaire showed that there was no test subject whose participation in the experiment would be unsuitable due to health issues. They were exposed to 3D model and asked to navigate, touch, rotate and go inside a 3D model for roughly 30 minutes. After exposure to the experiment, they filled out the Kennedy SSQ (Simulator Sickness Questionnaire) [23].

The difference between female and male participants was compared using SSQ total score in this study. The results indicated that women get more affected (higher sickness score) with the immersion than men. Statistical analysis shows significant difference (T-test: p = 6.7196e-04). Average level of sickness for female and male are $\mu_f = 147.19$, $\mu_m = 40.26$ respectively (see Figure 4). the SSQ requested to be filled after 10 and 30 minute after immersion in VR. As figure 4 shows, the level of SSQ total score for females is 40 while at the same time this value is 25 for female after 10 minute of immersion. These value after 30 minutes (post-exposure) approached to 148 and 115 for female and male group respectively. As seen, the level of score grows exponentially which in turn means the more participants stay in the model the more they will be affected from the virtual environment.

This finding indicates that navigation and interaction with virtual mock-up has less effect on male gender. Therefore, they can stay more time inside a real-scale simulation system. The emergence of the cyber-effect is due to either the conflict between visual and end-vestibular organs or appears because of difference between sensory organs and proprioceptive knowledge [24]. The reason of cyber-effect in this application is the second conflict.

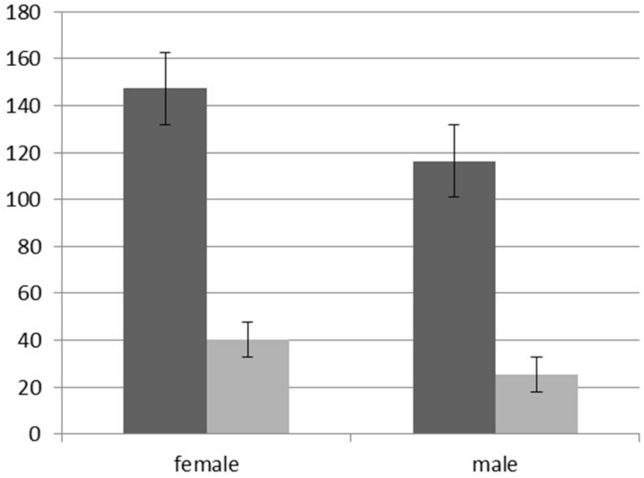

Figure 4: Comparison of SSQ score between male and female 10 (gray) and 30 (black) minutes after exposure

VII. Conclusion

A joint multi-touch table and CAVE platform was proposed to facilitate the design and visualization of an idea for a group of designers, programmers and end-users. A 2D sketch of an industrial idea is drawn in 2D and converted into a concise 3D model while integrating it with semantics. The model is projected into a virtual environment and displayed as real-scale model. Some property such as texture, color, material needs to be changed to look more realistic. These properties can be modified in real-scale but essential changes in the shaper and components of the model need to done on the multi-touch table. After modification again the new model can be projected and inspected in the real-scale. Because this process is performed fully in VEs no money needs to be spent and the process save a lot of extra cost design.

References

[1] D. J. Sandin, T. Margolis, J. Ge, J. Girado, T. Peterka, and T. A. DeFanti, "The varrier tm autostereoscopic virtual reality display," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 894–903, 2005.

[2] S. M. Aaron E. Walsh, Doug Gehringer, *Java 3D API Jump-Start*. Prentice Hall PTR, 2002.

[3] G. B. Rikk Carey, *The Annotated VRML 2.0 Reference Manual*. Addison-Wesley, June 1997, vol. 2.

[4] C. Wartmann, *The Blender GameKit: Interactive 3D for Artists*, 2nd ed., C. Wartmann, Ed. No Starch Press, 2009.

[5] R. Takeuchi, G. Chen, and D. P. Lepak, "Through the looking glass of a social system: Cross-level effects of high-performance work systems on employees' attitudes," *Personnel Psychology*, vol. 62, no. 1, pp. 1–29, 2009.

[6] S. Brown, B. Squire, and K. Blackmon, "The contribution of manufacturing strategy involvement and alignment to world-class manufacturing performance," *International Journal of Operations & Production Management*, vol. 27, no. 3, pp. 282–302, 2007.

[7] S. L. Lai, "The construction and realization process of the virtual manufacturing technology in package design," *Applied Mechanics and Materials*, vol. 271, pp. 539–543, 2013.

[8] P. F. Wang and X. H. Diao, "Network manufacturing technology based on cloud computing," *Advanced Materials Research*, vol. 601, pp. 390–393, 2013.

[9] L. F. Alarcón and D. A. Mardones, "Improving the design-construction interface," in *6th Annual Conference of the International Group for Lean Construction. Proceedings Guarujá, Brasil: IGLC*, 1998.

[10] K. Kotiadis and S. Robinson, "Conceptual modelling: knowledge acquisition and model abstraction," in *Proceedings of the 40th Conference on Winter Simulation*. Winter Simulation Conference, 2008, pp. 951–958.

[11] S. Robinson, "Conceptual modelling for simulation part i: definition and requirements," *Journal of the Operational Research Society*, vol. 59, no. 3, pp. 278–290, 2007.

[12] K. Otto *et al.*, "Towards semantic virtual environments," in *Workshop Towards Semantic Virtual Environments (SVE05)*, vol. 2, no. 2. Citeseer, 2005.

[13] D. Wiebusch and M. E. Latoschik, "Enhanced decoupling of components in intelligent realtime interactive systems using ontologies," in *Software Engineering and Architectures for Realtime Interactive Systems (SEARIS), 2012 5th Workshop on*. IEEE, 2012, pp. 43–51.

[14] W. Bille, B. Pellens, F. Kleinermann, and O. De Troyer, "Intelligent modelling of virtual worlds using domain ontologies," in *Proceedings of the Workshop of Intelligent Computing (WIC), Mexico City, Mexico*, 2004, pp. 272–279.

[15] R. M. Taylor II, T. C. Hudson, A. Seeger, H. Weber, J. Juliano, and A. T. Helser, "Vrpn: a device-independent, network-transparent vr peripheral system," in *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, 2001, pp. 55–61.

[16] J. Benesty, M. M. Sondhi, and Y. A. Huang, *Springer handbook of speech processing*. Springer, 2007.

[17] E. A. Suma, B. Lange, A. Rizzo, D. Krum, and M. Bolas, "Faast: The flexible action and articulated skeleton toolkit," in *Virtual Reality Conference (VR), 2011 IEEE*. IEEE, 2011, pp. 247–248.

[18] 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 the 20th annual conference on Computer graphics and interactive techniques*. ACM, 1993, pp. 135–142.

[19] Ar-tracker: http://www.ar-tracking.com/home/.

[20] A. Bierbaum, C. Just, P. Hartling, K. Meinert, A. Baker, and C. Cruz-Neira, "Vr juggler: A virtual platform for virtual reality application development," in *Virtual Reality, 2001. Proceedings. IEEE*. IEEE, 2001, pp. 89–96.

[21] Dtentity: http://code.google.com/p/dtentity/wiki/dtentity.

[22] D. Burns and R. Osfield, "Open scene graph a: Introduction, b: Examples and applications," in *Proceedings of IEEE Virtual Reality*. Published by the IEEE Computer Society, 2004.

[23] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness," *The international journal of aviation psychology*, vol. 3, no. 3, pp. 203–220, 1993.

[24] J. Reason, "Motion sickness adaptation: a neural mismatch model." *Journal of the Royal Society of Medicine*, vol. 71, no. 11, p. 819, 1978.