AC 2011-2478: EDUCATIONAL USE OF VIRTUAL WORLDS FOR ENGINEERING STUDENTS

Tumkor Serdar, Stevens Institute of Technology

Serdar Tmkor is affiliated as a research scientist and adj. professor in Mechanical Engineering Department at Stevens Institute of Technology. He had been a full-time faculty member of Mechanical Engineering Department at Istanbul Technical University since 1996. Dr. Tmkor received his PhD in Mechanical Engineering from Istanbul Technical University in 1994. His teaching interests are Machine Design, Engineering Design, and Engineering Graphics. His current research interests include Design of MEMS devices with polymeric nano-composites, computer integrated design, process planning and manufacturing, online design catalogs, and web-based collaboration.

El-Sayed Aziz, Stevens Institute of Technology

Dr. El-Sayed Aziz holds a faculty position as assistant professor in the Production Engineering and Mechanical Design Department at Faculty of Engineering, Mansoura University, Egypt. Currently, he is working as research scientist at Stevens Institute of Technology, Hoboken, New Jersey, USA. He received B.S. and M.S. degrees in Mechanical Engineering from Mansoura University, Egypt, in 1991 and a Ph.D. in Mechanical Engineering from Stevens Institute of Technology in 2003. His research interests include knowledge-based engineering systems, computer-integrated design and manufacturing, Finite Element Analysis, software development and applications as well as remote and virtual laboratories.

Dr. Sven K. Esche, Stevens Institute of Technology

Sven Esche is a tenured Associate Professor at the Department of Mechanical Engineering at Stevens Institute of Technology. He received a Diploma in Applied Mechanics in 1989 from Chemnitz University of Technology, Germany, and was awarded M.S. and Ph.D. degrees from the Department of Mechanical Engineering at The Ohio State University in 1994 and 1997, respectively. He teaches both undergraduate and graduate courses related to mechanisms and machine dynamics, integrated product development, solid mechanics and plasticity theory, structural design and analysis, engineering analysis and finite element methods and has interests in remote laboratories, project-based learning and student learning assessment. His research is in the areas of remote sensing and control with applications to remote experimentation as well as modeling of microstructure changes in metal forming processes. He publishes regularly in peer-reviewed conference proceedings and scientific journals. At the 2006 ASEE Annual Conference and Exposition in Chicago, USA, he received the Best Paper Award for his article 'A Virtual Laboratory on Fluid Mechanics'.

Dr. Constantin Chassapis, Stevens Institute of Technology

Educational Use of Virtual Worlds for Engineering Students

Abstract

One possible way for increasing student motivation is using computer software and simulations, such as a game-based learning and training platform. There already exist many web-based learning games, which may be based on pre-defined interaction scenarios. In recent years, such interactions have occurred in virtual worlds, where the users of these virtual worlds (avatars) debate, negotiate, simulate the consequences of various scenarios and solve simplified real-world problems. This virtual setting may provide an effective learning experience for students, and it may also capture and hold their attention. Virtual worlds have been tailored for many subject areas, but more work on adopting this approach for engineering education is warranted. For instance, because of the technical restrictions of the programming for such virtual worlds, the implementation of realistic laboratory exercises has turned out to be very challenging.

In this paper, the development of a classroom and a laboratory space with a number of engineering experiments in Second Life/OpenSim 6.7 is described. The technical problems encountered during this development can be divided into three main groups: (a) insufficient visual appeal of the object models, (b) problems with the physics engine and (c) limited programming capabilities. The object models are low in graphical quality, giving them poor visual appeal. The physics engine of Second Life does not allow objects to undergo realistic motions. Limitations of Second Life create difficulties in implementing coordinated motions of multiple objects. Some methods for overcoming these technical obstacles in creating virtual laboratory experiments are discussed in this paper.

Introduction

Nowadays, college students spend more time than ever in front of a computer, and most students are members of online communities such as virtual environments and social networks, where they can play games, exchange information and share their experiences. Game-based learning in virtual environments is not a novel idea [1,2]. Traditionally, virtual learning environments had been two-dimensional in nature and based on web browsers. Over the last few years, three-dimensional (3D) virtual worlds have been extensively used for educational purposes. Many institutions have created virtual campuses inside of virtual worlds and have started to offer virtual lectures. However, most of the courses currently being delivered in such virtual environments are on non-engineering subjects related to social or language sciences. Many subjects have been adopted, but there has not been sufficient work done for engineering education [3,4,5]. New trends are tending towards laboratory experiences in immersive environments such as games or virtual worlds [6]. Some technical simulations [7] and test beds [8] with various interaction possibilities were implemented in Second Life, but technical limitations and difficulties were also reported. Game-based environments accessed via computer networks enable students to more easily collaborate and conduct experiments in teams without the limitations of physical space. 3D visualization helps students to remember what they are learning by providing them an experience in a virtual setting that they cannot get from reading textbooks. In addition to these benefits, the evaluation of the team members is easier in virtual environments than in real-life hands-on laboratories. However, in virtual environments like Second Life, there

is no system available to collect, control, store and share information between the clients' Web applications and the virtual world. Various systems were developed to address this issue [9].

A game-based laboratory environment was designed, implemented and piloted at Stevens Institute of Technology (SIT) [10,11]. The virtual laboratory environment at SIT consists of (i) a game-based virtual laboratory facility where the students obtain laboratory instructions, divide the tasks among themselves and assemble a virtual experimental setup, (ii) a remote laboratory module that the students connect to from inside the virtual laboratory facility to perform real-time experimental procedures based on actual physical hardware via the Internet, and (iii) a virtual laboratory module that the students use to simulate experimental procedures that go beyond those possible with the physical hardware.

Second Life is classified by some educators as a Multi-User Virtual Environment, a term for virtual worlds that lack the 'game' component [12]. As 3D virtual community, Second Life increased in popularity, teaching and meeting spaces were designed to compare 3D game with 3D virtual world communities. This paper describes some of the methods used to overcome the technical obstacles in creating virtual laboratory experiments in Second Life, a popular virtual environment that so far has mostly been used for entertainment and social interactions.

Overview of Second Life / OpenSimulator

Although Second Life looks like a 3D game, it is one of the most popular non-game, 3D multi-user virtual environments. Contrary to multi-user computer games, there is no story to follow and no level to accomplish. Instead, Second Life is a simulation of the real world where the citizens of this world can meet other residents, socialize and participate in individual and group activities [13,14,15]. In this sense, Second Life is an Internet-based 3D virtual community. This social network allows its users from various geographical locations to interact with one another in the same virtual land. Second Life acts as a bridge between the text-structured 2D Web and a 3D virtual world. The members of this social network can communicate by text-based or voice-based chat and use avatar gestures to show their impressions. They can create their own settings by customizing avatars, designing the terrain of the land and modeling objects in three dimensions. The avatar-customization, terrain-editing and object-building tools of the viewer allow for fast and easy modification of the content.

Free client programs, called viewers, such as the Second Life viewer or the Hippo OpenSimulator (often referred to as OpenSim) viewer [16], enable users to interact with each other through avatars. OpenSimulator is an open source server platform for hosting virtual worlds. While OpenSimulator is most recognized for its compatibility with the Second Life client, it is also capable of hosting alternative worlds with different feature sets with multiple protocols and operating systems (Figure 1).
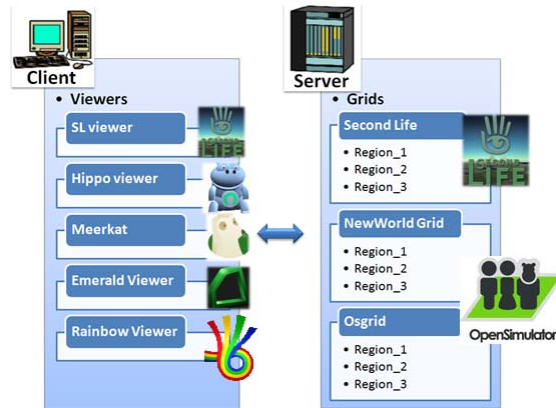
Figure 1: Client-server architectures of Second Life or OpenSimulator

The created models can be linked together and programmed using the Linden Scripting Language (LSL) to control the behavior of the objects. The interactive content creation from inside the viewer is very convenient because it is not necessary to adopt an expensive 3D modeling software to import the models. However, highly optimized primitive shapes are used to model the objects, and the visual appeal of these objects is sacrificed in order to reduce the required data traffic through the network.

Teaching in Second Life

It is pointed out that education in virtual worlds is a fast moving area [17]. The potential of using virtual worlds for distance engineering courses motivated the effort to create classrooms for demonstrations, simulations and hands-on experiments at SIT. A virtual model of the Edwin A. Stevens building (where SIT was originally founded) was created in Second Life and OpenSimulator (Figure 2). Each floor contains classrooms and laboratory spaces.

Figure 2: Virtualized Edwin A. Stevens building and virtual classroom

The students can attend classes, try different experiments and collaboratively work on their projects. The classrooms have a lectern or desk for the lecturer, a seating area for 20 to 40 students, a lecture slides viewer and a whiteboard. Collaborative classrooms also have large monitors with the functionality for displaying a variety of media including demonstrations, simulation videos and informative webpages. In Second Life, collaboration is an integral part of the environment. Educators and students can not only chat with each other via a text-based chat tool, but they can also use built-in video and audio functions to share their own webcam and voice media to converse verbally as well (Figure 3). Having the option to attend a presentation via a virtual world from home can help the user to be more at ease and comfortable.

Figure 3: A lecture in OpenSim

A pilot classroom was implemented for 'ME 358 Machine Dynamics and Mechanisms', a mandatory junior-level mechanical engineering course at SIT. In this course, the principles of kinematics and dynamics are introduced and applications such as linkages, cam systems, gear trains, belt and chain drives and others are covered. The three-credit course consists of lectures, weekly graded homework and a series of laboratory exercises [18,19].

Among other experiments, the first laboratory component of this class is an industrial plant emulator, which introduces the students to the modeling of inertia, friction, backlash and stiffness phenomena in machines (Figure 4).

Figure 4: Industrial plant emulator (right) and its virtual model in OpenSimulator (left)

The emulator consists of a drive disk and a load disk, which are connected via a speed reduction unit. The real emulator system is equipped with encoders connected to the drive and disturbance motors and a friction brake. With the virtual equipment created in OpenSimulator, students can assemble and disassemble loads, gears and belts to configure the experiments and run the simulation remotely over the internet using the virtual laboratory system [20].

The second virtual experiment created in Second Life was a vibration system. After the online lecture, students can build their own spring-mass systems of different sizes and apply loads to observe the resulting vibrations (Figure 5). Using a browser inside of Second Life, the students can input the parameters to customize their spring-mass systems. Then, the undamped circular natural frequency and the undamped natural frequency are calculated. The created vibration system is simulated and animated automatically using the user-specified data. This is similar to the virtual experiments that have been used at SIT for years [21,22]. The difference is that this Second Life experiment enables the students to explore and design the vibration systems collaboratively in teams from different locations at the same time.
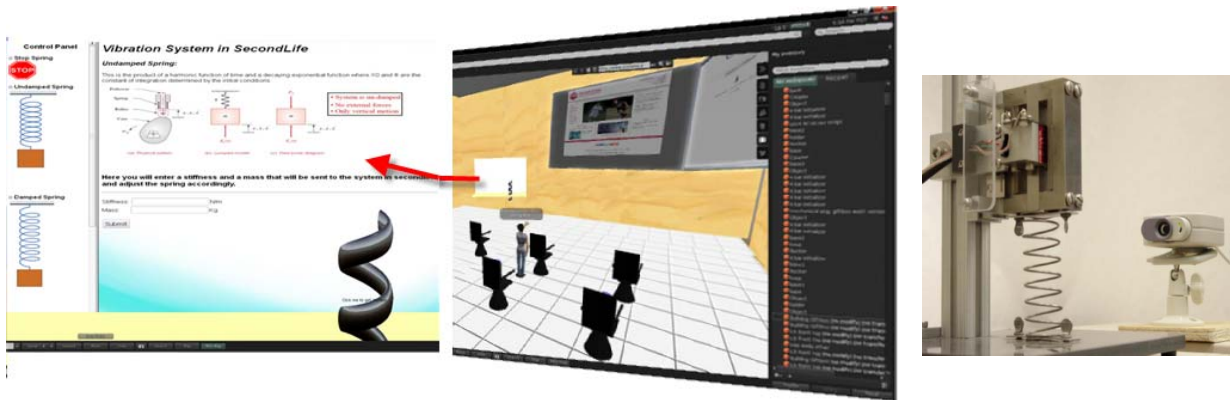
Figure 5: Vibration systems in Second Life and a one-degree-of-freedom remote setup

Also created in Second Life was a four-bar linkage experiment. A four-bar linkage is a multi-lever device used to convert rotary motion into reciprocating motion or vice versa. The virtual experiment is initiated by designing the linkage with user input. The input parameters are the link lengths (denoted as a, b, c, d in Figure 6) and the initial crank angle.
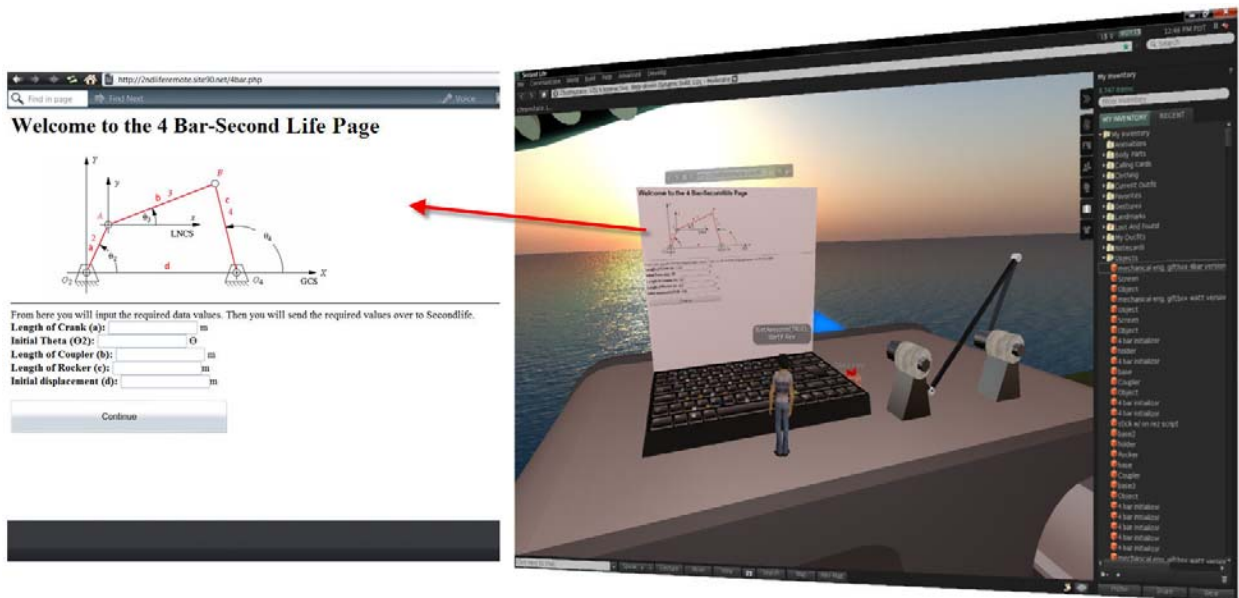

Figure 6: Four-bar linkage design in Second Life

Some four-bar linkage examples can also be found on the website, including for instance Watt's straight-line mechanism. If the students choose to try the Watt mechanism, they only need to provide the input link length or pick one of the preset links.

The linkages are assembled automatically by Second Life by inserting pins. At the same time, the input parameters are passed to a website that performs the calculation for the animation of the linkage. Once the calculations are completed, the linkage's Grashof condition is reported in the

text box at the bottom left of the screen. If the linkage is non-Grashof, the angles at which it reaches its toggle positions are calculated. The animation speed depends on the speed of the computer and mostly on the network and server speed. The animation may be jerky if the overall speed is too low. It gives the impression as if the links were broken as shown in Figure 7.

Figure 7: Jerky animation of four-bar linkages

Technical Problems and Limitations of Second Life

The technical problems and limitations of Second Life make it difficult to implement realistic virtual experiments. Second Life provides a high degree of collaboration, but it has limited capabilities for modeling objects with visual appeal because of the low geometric complexity supported and the lack of a surface mesh. Massive multi-user virtual environments probably require sacrificing visual appeal because of the network speed concerns.

Figure 8: Modeling using commercial CAD tools and importing models into Second Life

The highest degree of object fidelity is achieved by commercial CAD tools (Figure 8). However, it is only possible to import models to Second Life if they consist of primitive shapes, just like the objects created with the object builder inside of Second Life (Figure 9). One way to create realistic objects is to use textures or sculpties (sculpted primitive objects whose shapes are determined by textures) [23].
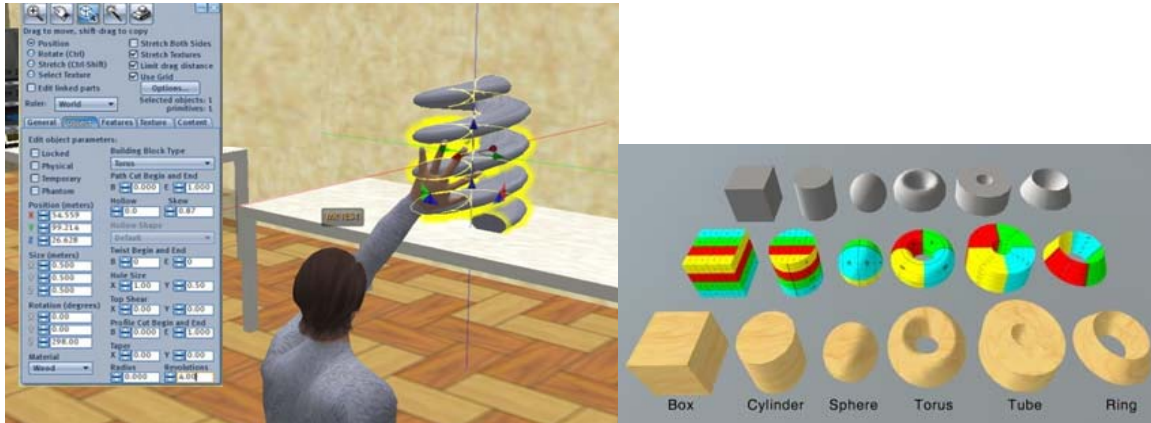

Figure 9: Built-in object editing tool and its primitive shapes

All physics simulations within Second Life are handled by the Havok physics engine. The function of a physics engine is to simulate gravity, elasticity and the conservation of momentum between colliding objects. The Havok physics engine is capable of simulating thousands of physical objects at once. However, the version of Havok used by Second Life is extensively optimized for multi-user internet access and has some limitations. The simulation of Second Life is not a complete physical simulation. While certain physical effects, such as gravity, are modeled realistically, the physics engine of Second Life does not allow objects to undergo coordinated motions. Therefore, in the implementation of the virtual laboratory, the motions of

the objects of the experimental setups were generated by scripts rather than by the physics engine.

Second Life provides an event-based scripting language called the Linden Scripting Language (LSL). It can be used to add action and interaction functionality/behavior to characters and objects. Scripts can be written that recognize button-click events and perform an action based on these events. The syntax of the language is similar to that of Java or C. Limitations of Second Life's programming capabilities create difficulties in implementing coordinated motions of multiple objects. The most significant disadvantage is the limit in data types and memory. Every LSL script is capped at 16 KB, while a Mono script (another open source scripting engine in Second Life)[24] can use up to 64 KB of total memory. This restriction represents a significant problem that makes Second Life a really limited programming environment. Another difficulty is the way the user interface accepts inputs. The user may only provide input data by a simple chat box (llTextbox) or by a limited dialog box (llDialog) with a fixed set of buttons (Figure 10). The reason for that may be that Second Life was not intended to be used as a software development environment, but it is merely a platform where users are enabled to create their own buildings and objects with simple steps.

Furthermore, LSL does not offer a built-in persistent data storage functionality, such as a file or database. However, LSL does provide communication functions via HTTP to outside servers. This mechanism for communicating via HTTP can be used to store the state of an object externally. The objects request data from the web server using the LSL function llHTTPRequest.
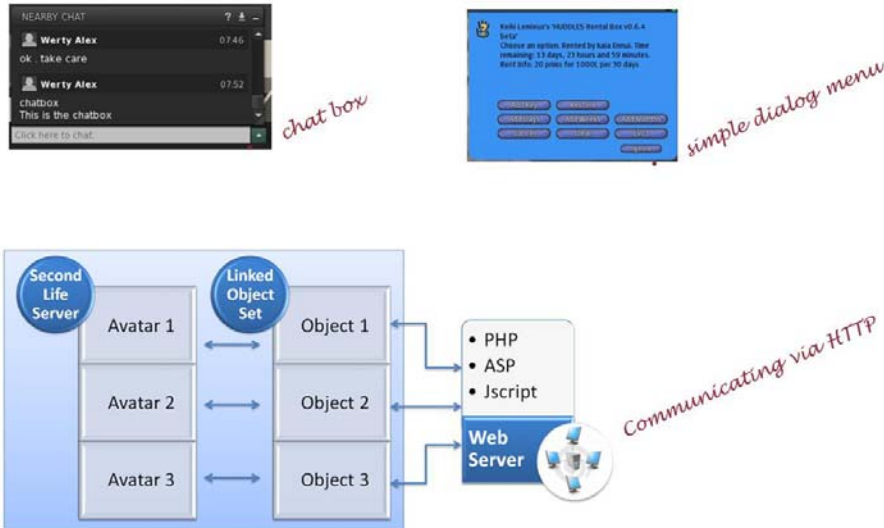

Figure 10: Limited user interface of Second Life and communication implemented via HTTP

The same communication method can be used to implement user interactions (Figure 10). The user input is facilitated by an external web server, but the website is accessed from inside of Second Life so that the users do not need to leave the environment.

How Effective Can Second Life Be in Student Learning

Second Life offers a useful medium for role playing. The findings of a pilot study in an English course showed that the students who enjoy console and computer games tend to like being in a virtual world and generally respond more favorably to Second Life than students who do not enjoy such games [25]. Students must also be willing to be active, become participatory learners, and to demonstrate collaboration, critical thinking and reflection. Challenging and collaborative characteristics of the environment create a better learning experience. Educators should provide tools that support both synchronous and asynchronous collaboration and challenge the students in a team setting that allows them to research and reflect on their own choices. In addition, students can also explore the different aspects of a topic by experiencing the available visual media (such as video content or simulations within or outside of Second Life) from a different perspective. The role of the educator in the Second Life virtual world should be to facilitate and to connect and advise students in their learning. The most effective uses of Second Life in engineering education can be centered around project-based assignments. Students can develop greater organization, critical thinking, communication and multitasking skills.

Conclusions

This paper provided an overview of continuing research at Stevens Institute of Technology on the use of 3D multi-user virtual environments for engineering education. Classrooms and laboratory exercises were developed in Second Life to demonstrate the feasibility of creating a highly interactive and collaborative educational environment for geographically dispersed students. However, because of its technical limitations, Second Life is not recommended for developing interactive laboratory setups for engineering students. 3D multi-user virtual environments have a tremendous potential for creating effective tools for education, training and research, but they are still in the early stages of development. Second Life is extensively optimized for multi-user internet access, but it also exhibits serious limitations. If future versions of Second Life can overcome those limitations, the next stage of the Second Life project should be to create more realistic objects, to integrate them such as to control real experiments that can be used in a real classroom situation, and to assess the usefulness of this approach with students.

Acknowledgments

References

1. Dede, C., Nelson, B., Ketelhut, D. J. & Bowman, C. (2004). Design-based research strategies for studying situated learning in a multi-user virtual environment. *Proceedings of the 6th International Conference on Learning Sciences*, Santa Monica, CA, USA.
2. Peachey, A. (2010). *Researching Learning in Virtual Worlds, Human-Computer Interaction Series*. 1st Edition, Springer.
3. Livingstone, D., Kemp, J. & Edgar, E. (2008). From multi-user virtual environment to 3D virtual learning environment. *Research in Learning Technology*, Vol. 16, No. 3, pp. 139-150.
4. Molka-Danielsen, J. & Deutschmann, M. (2009). *Learning and Teaching in the Virtual World of Second Life*. Tapir Academic Press, 2009.

5. Seng, J. L. K. & Edirisinghe, E. M. N. S. (2007). Teaching computer science using Second Life as a learning environment. *ICT: Providing choices for learners and learning. Proceedings ascilite Singapore 2007*, Nanyang Technological University, Singapore, pp. 583-586.

6. Tüzün, H., Yılmaz S., M., Karakuş, T., Inal, Y. & Kızılkaya, G. (2009). The effects of computer games on primary school students' achievement and motivation in geography learning. *Computers & Education*, Vol. 52, No. 1, pp. 68-77.

7. García-Zubia J., Irurzun, J., Angulo, I., Orduña, P., Ruiz-de-Garibay, J., Hernández, U. & Castro, M. (2010). Developing a Second-Life-based remote lab over the WebLab-Deusto architecture. *Proceedings of the Remote Engineering and Virtual Instrumentation Conference (REV2010)*, Stockholm, Sweden, pp. 171-176.

8. Brandherm B., Ullrich, S. & Prendinger, H. (2008). Simulation of sensor-based tracking in Second Life. *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Demo Papers*, May 12-16, Estoril, Portugal, pp. 1689-1690.

9. Mahmud, J., Huang, Y. W., Ponzo, J. & Pollak, R. (2010). Avara: A system to improve user experience in Web and virtual world. *Proceedings of the 14th International Conference on Intelligent User Interfaces*, New York, NY, USA, pp. 349-352.

10. Chang, C., Kodman, D., Chassapis, C. & Esche, S. K. (2007). Immersive collaborative laboratory simulations using a game engine. *Computers in Education Journal*, Vol. 17, No. 3, pp. 85-92.

11. Aziz, E.-S., Esche, S. K. & Chassapis, C. (2010). An interactive game-based engineering laboratory. *World Transactions on Engineering and Technology Education*, Vol. 8, No. 2, pp. 131-136.

12. Bell, L. & Trueman, R. B. (2008). *Virtual Worlds, Real Libraries: Librarians and Educators in Second Life and other Multi-User Virtual Environments*. Information Today Inc., New Jersey, USA.

13. Definition of Second Life. *Wikipedia page*, http://en.wikipedia.org/wiki/Second_Life, accessed on 03/11/11.

14. Callaghan, M. J., McCusker, K., Losada, J., Harkin, J. G. & Wilson, S. (2009). Engineering education island: teaching engineering in virtual worlds. *e-journal of the Higher Education Academy Subject Centre for Information and Computer Sciences (ITALICS)*, Vol. 8, No. 3, pp. 2-18.

15. Gough, R. & Dearnley, J. (2009). Engineering Education Island: Teaching Engineering in Virtual Worlds*, e-journal of the Higher Education Academy Subject Centre for Information and Computer Sciences ITALICS*, Vol. 8,No.3, pp.19-25

16. Hippo OpenSim Viewer. *Website*, http://mjm-labs.com/viewer, accessed on 03/11/11.

17. Carpenter, S. B. (2009). Virtual worlds as educational experience: Living and learning in interesting times, *The Journal of Virtual Worlds Research*, Vol. 2. No. 1, pp. 3-4.

18. Arango, F., Altuger, G., Aziz, E.-S., Esche, S. K. & Chassapis, C. (2008). Piloting a game-based virtual learning environment. *Computers in Education Journal*, Vol. 18, No. 4, pp. 82-91.

19. Aziz, E.-S. (2009). Teaching and learning enhancement in undergraduate machine dynamics. *Computer Applications in Engineering Education*, Vol. 17, No. 2, pp. 1-15.

20. Aziz, E.-S., Esche, S. K. & Chassapis, C. (2010). Design and implementation of a virtual laboratory for machine dynamics. *International Journal of Online Engineering*, Vol. 6, No. 2, pp. 15-24.

21. Aziz, E.-S., Esche, S. K. & Chassapis, C. (2007). On the design of a virtual learning environment for mechanical vibrations. *Proceedings of the 37th ASEE/IEEE Frontiers in Education Conference*, Milwaukee, WI, USA.

22. Aziz, E.-S., Esche, S. K. & Chassapis, C. (2009). Content-rich interactive online laboratory systems. *Computer Applications in Engineering Education*, Vol. 17, No. 1, pp. 61-79.

23. Linden Lab Research Inc., Second Life wiki: Sculpted Prims: FAQ. *Website*, http://wiki.secondlife.com/wiki/Sculpted_Prims:_FAQ, accessed on 03/11/11.

24. Linden Lab Research Inc., Second Life wiki: Mono, http://wiki.secondlife.com/wiki/Mono, last accessed 2011.

25. DIIA, Second Life: DIIA Investigates the Academic Viability of Virtual Worlds, *Case Study Series*, The University of Texas at Austin, http://www.utexas.edu/academic/ctl/about/postcards/casestudy3_second_life.pdf, accessed on 03/11/11.