Technical Aspects in Using X3D in Virtual Reality Mathematics Education (EViE-m Platform)

ATHANASIOS G. MALAMOS[1], GEORGIOS MAMAKIS[1], PARASKEVI SYMPA[1], ELENI KOTANITSI[1], ALFREDO JAVIER GONEL CRESPO[2], ALVARO ZUBIZARRETA LOPEZ[2]
[1]Department of Applied Informatics & Multimedia
Technological Educational Institute of Crete
Stavromenos, TEI Crete Campus, Herakleion Crete, 71004
GREECE

[2] Department of Computer Engineering
University of Valladolid
Plaza de Sta. Cruz, nº 8 47002. Valladolid
SPAIN

Abstract: Mathematics is “a world of variables” of nature that lies between philosophy and reality. Teaching mathematics has always been an intriguing task, especially in areas of mathematics where visualization is overcome by conceptualism. In such cases, it is vital for the learner to understand the effects of each concept and its application to the real world. Thus, an educational platform aiming at mathematics education should, besides offering an intriguing environment, where easily visualized concepts may be applied, be capable of producing the results of less perceptual models to the user. An approach in this area would be utilizing 3D technologies, in order to provide an intriguing environment for the learner to acquire mathematical experience through problem solving imposed by the system. Instead of utilizing 3D representation of mathematical concepts, our approach utilizes 3D technologies as the basis for the user to get acquainted with mathematical problem solving and gradually build up knowledge through solving itself. In this scope, we have developed a VR educational platform for mathematics named Educational Virtual Environment – Mathematics (EViE-m). EViE-m platform offers an environment where the user tries to construct an organized city through answering to simple or complex mathematical questions that offer the possibility of gaining access to buildings and technology trees. The greater the level the user has reached, the more advanced the questions they have to answer in order to evolve their city. Thus, this imposes certain technological requirements that need to be dealt with in developing such approaches. These requirements stem also from the capabilities of the platform in the areas of networking, scenarios (problem solving) and random event generation – all the unique modules that provide user interaction and motivating user environment. In this paper we present the technological requirements – and potential solutions - imposed by the technologies engaged in our approach, as X3D technology, as well as the architectural procedures we have identified in the development of EViE-m.

Key-Words: collaborative virtual environments, X3D technology, real world simulation

1. Introduction

Computer assisted learning is one of the most interesting areas of modern educational research. Terms as e-learning or its modern variation m-learning have been widely researched, both from an educational and from a technological aspect. However, computer-assisted learning is more than information presentation; it can become a constant search for information through inner motivation. Therefore, assisting user motivation in knowledge, in our opinion is even more important than offering an area of user information. This approach, however, is the outcome of modern technological advances such as collaborative virtual environments. Collaborative virtual environments are, according to Redfern and Naughton [1], common areas where people interact with each other, exchange aspects, information and knowledge, and in utmost cases act as motivating environments utilized in distant learning. Modern technological advances especially in the area of virtual reality (VR) and 3D technologies may offer more than a common area where people interact; they offer a unique real world simulation where people may not only cooperate, but also visualize the after-effects of their actions, thus providing an intriguing feature to modern education. That being the case, it has been stated by McLellan in [2] that virtual environments assist in extending knowledge through two of its most special attributes: visualization and spatial memory. In addition to that, learners can build up to their knowledge ("cumulative knowledge") as they build on their world. In this paper, we target on the area of mathematics education. It has been proposed and verified by Gittler and Gluck [3] that virtual environments are ideal for the education of Geometry, as this area of mathematics is easily visualized, through improvements of the learners' spatial knowledge. We extend this approach by proposing a system capable of extending user knowledge in areas that cannot be easily visualized as calculus. This is accomplished by focusing on the outcome of the learners' calculations, providing a link between these calculations and the seemingly irrelevant real world problem solving. In this paper, we present a system (Educational Virtual Environment Mathematics - EViE-m) [4] capable of accomplishing the aforementioned, focusing on the technologies utilized in its development, the technological requirements imposed by the system and the architectural procedures identified. The rest of the paper is organized as follows: in section 2, we discuss work and research already accomplished in virtual environment assisted education, presenting projects that have utilized 3D technologies in the area of virtual environments and can be potentially extended to act as educational tools. In section 3, we provide a brief description of our approach, along with the identification of the technologies utilized in achieving such an effort, while in section 4, we discuss the utilization of these technologies and their requirements and propose a potential architecture based on these requirements. In section 5, we conclude with extensions and solutions in tasks that have not been dealt with in the current development status of EViE-m. The last section deals with our implementation approach and an identification of the problems we handled in the development of the platform, while the paper concludes with potential extensions to the system.

2. Background

Jackson and Fagan [5], who have studied collaboration and learning within immersive virtual reality (IVR), use a head mounted display technology and with the support of the College of Education and Human Interface Technology Laboratory at the University of Washington and they came to the conclusion that: "… VLEs can provide beneficial educational experiences unobtainable by any other means if they focus strongly on the affordances that are specific to the technology…" . Students also stated, that such an experience was very fertile from an educational approach. Lunce [6] administered a pre-test and post-test to students in a computer graphics course to gauge the effectiveness of a VR-enhanced simulation. According to the test results, students who used the VR-enhanced computer simulation scored higher on practical examinations (post-test). Lunce has also reported that students' response to the VR-enhanced computer simulation was positive as demonstrated by willingness to devote more time working in the simulation than to more traditional study methods. Moreover, he reported that "…computer simulations provide important benefits to the learning experience in distance education…". Additional work in the area has been undertaken by Naeve and Nilsson [7], who developed a platform named CyberMath comprised by a 3D graphical environment where people from different physical locations can exchange information and take part in lectures, demonstrations etc. The virtual reality environment provides the users with sound and the visitors are represented as avatars that enter the virtual space, which gives them the chance to observe and experience other users' actions. Yeh and Nason [8] whose aim is to design and evaluate a VRLE(Virtual Reality Learning Environment) to facilitate the construction of knowledge about 3D geometry concepts, developed VRMath which is a 3D space as well as a mathematical micro-world where users can move by using the mouse and the navigation keys. This application consists of three components:

- topological: is the virtual reality interface that allows real-time navigation within the 3D virtual space.
- typological: is the Logo-like programming interface that links to the topological component when students are programming to manipulate and build objects within the 3D microworld and refers to any meaning by kind discrete representations of geometry such as language, texts, numbers, icons and buttons.
- social action component: which includes the hypermedia and forum

interface that aggregates information and scaffolds discourse.

Another graphical environment named AquaMOOSE [9], [10] was designed to support free exploration of 3D mathematical concepts using primarily parametric equations and trigonometric functions. Motion in AquaMOOSE can be specified mathematically, by using parametric equations. An example they present is "Swimming in a sine wave in x and a cosine in y creates a spiral. At this application both Cartesian and polar coordinate systems are supported. According to their notes at the end of the school year indicated that very few of the students remembered anything about the unit on polar coordinates. Some of the answers of the students when asked if they enjoyed their experience with AquaMOOSE were: "I mean, I really didn't understand it overall. It was ok but like just to do, I wouldn't do it. Not just to have fun. I didn't think it was fun, if anything, it confused even more. Another, student responded: "AquaMOOSE was awful [sic]. I didn't learn a thing, my mind just got confused and un-oriented". As it occurred from the experiments of the AquaMOOSE platform, students appear to request a more challenging, competitive and larger environment with more potential for exploration. Furthermore, they are requesting more structured goals to win, more community involvement. They also refer that it would more interesting if they had the chance to collaborate and play with their friends. Furthermore, an approach for web-based mathematics education (WME) was considered by Wang, Kajler, Zhou and Zou [11]. The WME framework designs a distributed system to enable web-based mathematics education. WME empowers the teacher and eliminates many technical difficulties of on-Web mathematics education. Kaufmann, Schmalstieg and Wagner [12], [13] presented a three dimensional geometric construction tool called Construct3D designed for high school and university education. This setup consists of a menu system with large, textured 3D buttons with meaningful 3D icons floating above the buttons to allow easy and fast selection of the menu elements. A stereoscopic head mounted display (HMD) is also used and students can observe three dimensional objects in order to administer in a better way complex three-dimensional spatial problems and relationships. According to the authors "…The use of Construct3d is easy to learn and encourages experimentation with geometric constructions…" Alice [14], a 3d graphics programming environment designed for undergraduates with no 3D graphics or programming experience. It's a Windows 95/NT tool for describing the time-based and interactive behaviour which occurs by manipulating 3D objects.

3. Our Approach

In our approach, we are dealing with mathematical education, both in areas of high visualization (geometry) and in areas of lower visualization (calculus). Our platform focuses on presenting the user the outcomes of their calculations, rather than the visualization these concepts. Thus, it performs more in the manner of a game, rather than an educational platform. Its scope is to present the user with an environment, where the user can construct a series of buildings based on their current knowledge of mathematics, while in the same time offering the necessary motivation to extend their knowledge with the aim of extending their world. The platform is constituted by a series of modules and parameters:

- User affected parameters as education, economy and civil protection. These parameters are affected by user actions, such as infrastructure buildings and location of buildings.
- System parameters as catastrophic events and time
- Virtual units or points assisting in world evolution and user ranking
- Building and technological tree, assisting in world evolution
- Three levels of knowledge. In the first level of knowledge the system utilizes a simple statistical model in applying the catastrophic events, while the questions imposed to the user are of simple nature, both in calculus and in geometry. Along with the progress of the world, the user extends their knowledge as the question subsystem extends accordingly. In the second level the user gains access to second level buildings, while the system engages a more advanced statistical model in controlling catastrophic events. Location of buildings becomes more important as catastrophic events become targeted on areas of interest, while building construction becomes more difficult as advanced knowledge of mathematics is required. The third level of difficulty engages advanced statistical models for catastrophic events, expert knowledge in building construction and access to all the buildings available.
- Scenarios in the form of eXtensible Markup Language (XML files) where the level a building is available, the points it

contributes to the world and its prerequisite buildings and points needed to construct are verified.

In order to develop our platform, we utilized modern web technologies including X3D technology, a Web3D consortium standard for 3D representation of objects on the World Wide Web, Xj3D browser, which is a Java-capable X3D browser, Sun Java object oriented programming language and XML data representation technology.

4. Technological requirements

Our main scope in this attempt was to develop a highly modular and effective platform based on problem solving, while at the same time offering an intriguing environment for the user to continue interacting with this platform. In addition to that, we targeted on keeping the platform as generic as possible, implying that we desired to differentiate between the logic modules of the platform, the 3D objects and the areas of interest of the platform (scenarios). Thus, EViE-m is capable of being extended to others areas of interest, e.g. physics, just by maintaining everything but the scenarios the same. A scenario with physics questions can be used instead of the one for mathematics if it abides by the development requirements of the platform (specific syntax).

In order to accomplish this task we decided to utilize widely-available open source technologies. The core of the system is based on X3D technology, while each scenario and property file for the buildings that may be constructed is stored as an XML file. The logic module of the system was developed using Sun Java technology and is independent of the scenario and buildings loaded each time. The world is loaded using a Java-capable X3D browser (Xj3D) which deals with the representation of the virtual world and executes the logic associated with each file. In this paper we will focus on X3D technology explain how each of the aforementioned technologies contributes to the system in greater detail, validating the architecture followed.

4.1. X3D Technology

Extensible 3D (X3D) is an open standard proposed by Web3D consortium. It was developed to extend the World Wide Web with real-time 3D capabilities. X3D technology is based on the extension of Virtual Reality Markup Language (VRML) that was proposed on previous decades. This extension offers great capabilities to the developer as the classic VRML syntax, which highly resembled a scripting language, has been supplemented with a corresponding XML syntax, for representing VRML objects. Thus, it has become more extensible and easier to comprehend, while maintaining the powerful capabilities provided by XML data representation. In addition to that, X3D technology enables the use of Java programming language surplus to Javascript technology, becoming more complete and enabling a series of new features, from an algorithmic aspect. A typical X3D file in XML syntax resembles the one depicted in Fig. 1.

```
<?xml version="1.0" encoding="UTF-
8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD
X3D 3.0//EN"
"http://www.web3d.org/specifications/
x3d-3.0.dtd">
<X3D profile='Immersive' >
<Scene>
 <Shape>
   <Appearance>
    <Material DEF='Blue'
diffuseColor='0 0 1'/>
   </Appearance>
   <Box size='1 1 1'/>
 </Shape>
</Scene>
</X3D>
```

Fig. 1 X3D sample file

The X3D depicted in Fig.1 represents a blue box of size 1*1*1 m located in the default location (centre) of our world.

In our platform, X3D is used to as the data representation means of each building. Each building the user may construct is a set of an X3D file along with its textures. It is internally connected with its properties through a Java logic module. The properties file is stored in XML data format.

X3D is also utilized in the main world loaded initially in the platform. The main world holds all the necessary information for the location of the buildings to be loaded, the main logic and its own properties and status XML files. Using the X3D capabilities each building constructed is appended to the main world, along with its properties.

4.2 Extensible Markup Language (XML)

XML technology contributes to the platform by storing the necessary information that describes the usability and the requirements of each building. Usability in this case refers to the knowledge level it becomes available, its contribution to the world in points and to the various areas available in the world (quality of life, economy, technology, culture, health, environment, civil protection, and construction) and the effects catastrophic events have on it. The requirements of the building are typically the level the user has to reach prior to constructing this building, the number of necessary points required as well as the prerequisite buildings that need to be constructed beforehand. A typical property XML file is depicted on Fig. 2.

```
<?xml version="1.0" encoding="utf-
8"?>
<!DOCTYPE properties SYSTEM
"properties.dtd">
<properties>
      <property>
            <points>12000</points>
            <name>houseC</name>

      <preconds>houseB</preconds>

      <quality_of_life>0</quality_of_
life>
            <culture>0</culture>
            <education>0</education>

      <technology>0</technology>
            <economy>0</economy>
            <health>0</health>

      <constructions>0</constructions
>

      <environment>0</environment>

      <civil_protection>0</civil_prot
ection>
            <time>4</time>

      <point_rate>0</point_rate>
            <cost>40</cost>
      </property>
</properties>
```

Fig. 2 Property File Example

4.3. Xj3D browser

Xj3D browser is a Java-capable X3D browser available for download at http://www.xj3d.org. The browser is used to load X3D files both in XML and in VRML syntax, offering at the same time all the necessary libraries for interaction with the X3D files through Java interfaces. There are two potential approaches of interaction between the browser and Java: either embedding Java classes inside the X3D to be loaded through the browser, or loading the browser directly through a Java application. In our case we followed the first approach, in order to become more generic in future developments of other existing browsers as Octaga browser and Flux Player.

4.4. Sun Java technology

Java programming language, in our case, is used to develop middleware applications that connect an X3D building with its properties, the world with the buildings the users want to construct, the world with its status, the world with the corresponding scenario and finally the logic inside the world through the control of system parameters.

4.5. System architecture

The technologies described in previous sections are the minimum identified for the system to operate, since they only act as complementary to one another. We have identified that the most efficient way of storing properties for the buildings is through an external XML file, as little use can be made of the capabilities of storing application specific parameters inside an X3D file. In addition to that, during development it became clear that we needed to store the world itself as there are limitations in the dynamic handling of consequent X3D objects, thus requiring reloading the world with its current status. Keeping in mind that the main scope of the application is to maintain the highest possible technology independence, we propose the following architecture for our platform, as depicted in Fig. 3.

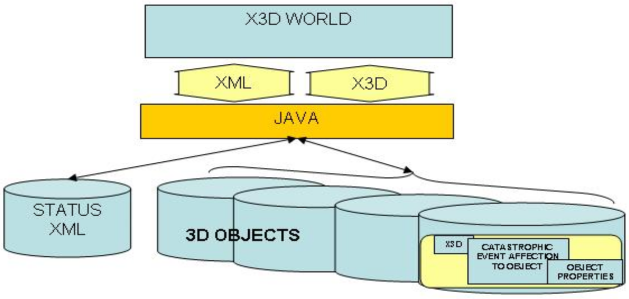

Fig. 3 EViE-m Architecture

As it may be seen in Fig. 3, on top of everything is the X3D world. The X3D world is comprised of two main features: X3D files representing the models and XML scenario files that represent the questions imposed by the system on the user in order to construct a building. These are connected using Java classes as middleware; classes that are embedded in each X3D file that link to the according XML file. Java middleware is also used to calculate and maintain the status of the world (user points, buildings constructed, level etc) while also it is used to interact with the 3D objects that may be loaded in the world. Java middleware is also responsible for handling system parameters as time and catastrophic events, which interfere with every building constructed on the virtual world.

5. Implementation Issues

Our approach in developing EViE-m deviates from the standard X3D application development. Usually, developing X3D applications is based on deploying the X3D browser (in our case Xj3D browser) inside the system; that is building up your application, loading the browser through appropriate methods. In our case, we selected to load the class implementing the logic module through our browser as a script node. Therefore, we became rather independent of the browser utilized, as in this case it is only required to have a browser capable of interpreting Java classes.

Our implementation approach defines four basic modules in the system:

- Basic Map Interface (BMI) responsible for the world and the menu objects.
- Dynamic Interaction Manager (DIM) responsible for the dynamic loading of the objects (buildings) in the main scene.
- Question Manager (QM) responsible for the Q&A module of the system.
- X3D File Manager (XFM) responsible for holding the state and status of the X3D world. This module is highly dependent to the QM and the DIM as it acts as the middleware application interconnecting both.

In the following UML diagram, the interaction between the modules in terms of user perception may be seen. This is the result of the interconnection between the modules, while user interaction with the system is also depicted

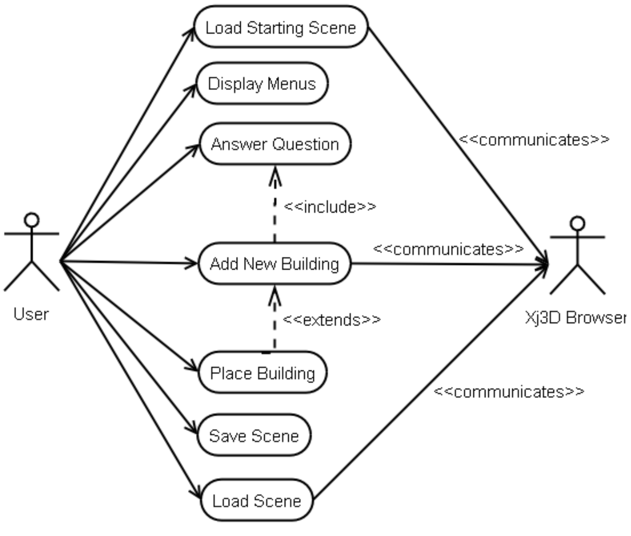

Fig. 4 UML diagram

In the following Figure (Fig. 5), the class diagram connecting all four modules may be depicted. The main class in our application is the Main Scene (MS). MS is directly connected with the Xj3D browser and the buildings added in the scene, while it also relates to the BMI responsible for loading the menu of the system. On a second level, it interacts with the DIM, which is responsible for loading the questions imposed on the user through the browser, by interacting with the question manager. The XFM in this case, loads and saves every state of the world, on each user - system interaction.

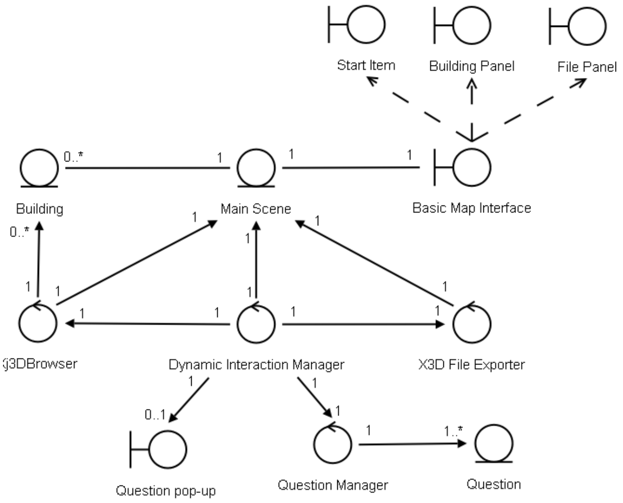

Fig. 5 Application Class Diagram

The main user interface of the application is depicted in Fig. 6. As it may be observed, the user is presented with the map where the objects (building) may be loaded. Inside the world an object menu is displayed that enables constructing buildings in the world, saving and loading world state.

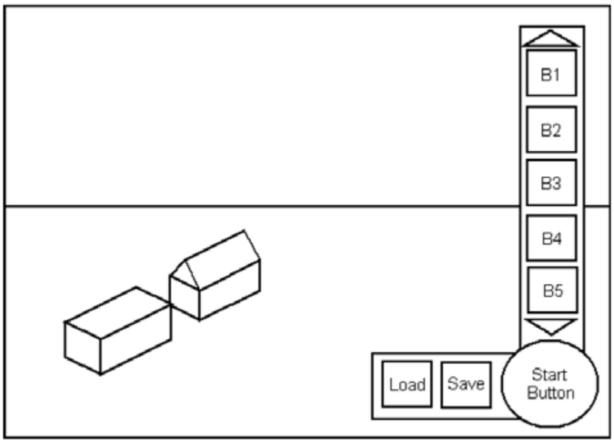

Fig. 6 User Interface Prototype

5.1. Use case scenario

User interaction with the system is an easy procedure. The user selects the building to be constructed in the world. The system checks whether this building's preconditions are satisfied and imposes the question to be answered to the user. The QM module checks for answer validity and constructs the building in the scene. The user places the building in the world on their desired location and the scene is saved and the save file is automatically loaded in the world.

5.2. Problems Identified

A series of problems were faced during the development of the application. First of all, our implementation approach limited us in the events we could handle from within the browser. This implied that, we could handle objects and sensors from the scene but we could not have direct access to the events thrown through the browser, as the browser is responsible for loading system module and not vice versa. Thus, we had to construct a workaround, which included loading the models of the buildings in a hidden scene and appending this scene to the browser. In addition to that, direct access to the scene was not handled properly by our logic module due to limitations imposed by X3D development progress. Thus, this led to saving and loading the current state of the scene after every building construction. Moreover, dynamic route handling (basic operation in dragging a building to a desired location) through scripting was detected and reported to the Web3D consortium, responsible for developing the X3D specifications. This will dealt with in later releases of the platforms. Last but not least, we had to deal with dragging objects into multiple directions. In X3D specification two kinds of object handling procedures (sensors) are defined: a) touch sensors enabling user events to be generated and b) drag sensors enabling modifications in the location (translation) and perspective (rotation) of the object. However, one may not apply two sensors of the same type on a single object. Thus, dragging an object in 3 dimensions can not be accomplished, as it would require three sensors of the same kind on a single object (plane sensors enabling movement on a plane, namely X axis plane, Y axis plane and Z axis plane). However, through extensive Java scripting, we achieved in handling with 2-dimensional movements (X-axis and Z- axis movement). This was achieved by aligning the objects (buildings) with the ground – keeping a stable Y-axis – and attaching handles to each building. Each handle could move on a specific axis (X, Z), and while one of these handles was enabled, the other handle was disabled. Motion in one handle, resulted in moving both the object and the group of the second handle and its accompanying sensor. Thus, we achieved in enabling 2D motion in every building we loaded in our world, and therefore placement on a specific location was accomplished.

Apart from limitations imposed by X3D specifications, we had to deal with functional problems in our user interface, as we troubled ourselves in finding the proper manner of providing the user with a dynamic object menu. The object menu itself is an X3D object residing in our world, moving along with our viewpoint. The objects loaded in the object menu are dynamically handled by the BMI, and reside in an appropriate X3D file. Thus, extending the X3D file with new nodes results in including new buildings in our world – provided that we define the model to be loaded and its unique characteristics.

6. Conclusions

In this paper, we have presented technological issues regarding the engagement of modern technologies in developing a Collaborative Virtual Workspace, with Virtual Reality capabilities, targeting to act as an educational platform in the area of mathematics. The platform named EViE-m is initially targeting on extending user knowledge by motivating the user to continue his involvement in the platform, through an interactive and intriguing virtual world, a world that they construct and develop. Along with the world's development we try to achieve expansion of the knowledge of the learner. We have already identified our next steps that may greatly assist in maintaining and extending user motivation, namely enabling the system performing not only on a standalone, but rather on a network level, where users from all over the world may interact to compare their worlds, creating the spirit of collaboration. Unfortunately, the capabilities of networking in X3D technology are still very primitive, yet we have already tested the ability of transmitting X3D information using peer to peer technology over a network with encouraging results, while the Web3D consortium is already extending the networking component of X3D technology. In addition to that, it is also important to focus in the area of competition between users. To that direction, a global user database may be developed holding the number of points each user acquired through their experience with the game. Thus, a ranking system may be initiated, offering the learner with another goal to achieve.

References

[1] S. Redfern and N. Naughton, Collaborative Virtual Environments to Support Communication and Community in Internet-Based Distance Education, *Journal of Information Technology Education*, Vol.1, No. 3, 2002, pp. 201-211

[2] H. McLellan, *Virtual Realities*, *Handbook of Research for Education, Communications and Technology*, New York, NY, Simon &Shuster MacMillan, 1996, pp. 461-498

[3] G. Gittler and J. Gluck, Differential Transfer of Learning: Effects of Instruction in Descriptive Geometry on Spatial Test Performance, *Journal of Geometry and Graphics*, Vol. 2, No. 1, 1998, pp. 71-84

[4] A. G. Malamos, P. V. Sympa, G. Mamakis and Y. Kaliakatsos, EviE – m (Educational Virtual Environment Mathematics), An Alternative Approach for an Educational Strategy Game For Mathematics, *in Proceedings of the 8th International Conference on Computer-Based Learning in Science 2007,* 2007, pp 374 - 382

[5] R.L. Jackson and E. Fagan, Collaboration and Learning Within Immersive Virtual Reality, *in Proceedings of the 3rd International Conference on Collaborative Virtual Environments*, 2000, pp. 83 – 92

[6] L. M. Lunce, Computer Simulations in Distance Education, *International Journal of Instructional Technology and Distance Learning*, Vol. 1, No. 10, DonEl Learning Inc., 2004

[7] A. Naeve and M. Nilsson, ICT-enhanced Mathematics Education within the Framework of a Knowledge Manifold, *in Proceedings of the 10th International Congress of Mathematics Education*, 2004

[8] A. Yeh and R. Nason, Knowledge Building of 3D Geometry Concepts and Processes within a Virtual Reality Learning Environment, *in Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pp. 2175-2182, 2004

[9] J. Elliot, L. Adams and A. Bruckman, No magic bullet: 3D video games in Education, *in Proceedings of 5th ICLS 2002*

[10] J. Elliot and A. Bruckman, Design of a 3D Interactive Math learning environment, *in Proceedings of DIS 2002*, pp. 64 – 74, 2002

[11] P. S. Wang, N. Kajler, Y. Zhou and X. Zou, Initial Design of a Web-based Mathematics Education Framework, *in Proceedings of IAMC 2002*

[12] H. Kaufmann, D. Schmalstieg and M. Wagner, Construct3D: A Virtual Application for Mathematics and Geometry Education, *Education and Information Technologies,* Kluwer Academic Publishers, Vol. 5, No. 4, pp. 263-276, 2000

[13] H. Kaufmann and D. Schmalstieg, Mathematics and Geometry Education with Collaborative Augmented Reality, *in SIGGRAPH 2002 Educators Program SIGGRAPH 2002 Conference Abstracts and Applications*, pp. 37-41, 2002

[14] M. Conway, S. Audia, T. Burnette et. al.., Alice: Lessons learned from building a 3D System for Novices in CHI 2000, *CHI Letters*, Vol. 2, No. 1, pp. 486-493, 2000