Control technology as a means for designing virtual interactive space: what could be learned from blender use in architectural education?

Vassilis Bourdakis, Anna Chronaki[2],

[1] University of Thessaly, Departmentof Architecture
[2]University of Thessaly, Department of Early Childhood Education, Argonafton & Filellinon, 38221, Volos, Greece
{vas, chronaki @ uth.gr}

Abstract. The creative construction of virtual interactive space may benefit from the explicit employment of 'control' technology in the design process. The notion of 'control' technology is neither new for the architectural studio nor for the architectural education. Instead, there is a history of tool-utilization (and production) for developing both realistic spatial relations concerning building sites but also imagining virtuality (and its interrelation to reality) or even combining real (or physical) and virtual (or digital) components into creating hybrid or/and ubiquitous space. The present paper aims to discuss the utilization of blender as part of architect students' education to both 'control' technology and 'virtual interactive space' design. In the following, after a brief background of digital technologies in architectural education, we describe the blender platform and then we describe some representative students' work that exemplify how issues of 'control' and issues of 'virtual interactive' come together.

Keywords: control technology, virtual interactive space, architectural education, design process

1 Introduction

During the last three decades we have experienced, admittedly in various forms, the infiltration of digital technologies in the field of architecture and (often with hysteresis) in architectural education. The process has not been smooth, the conceptual framework is often limited to visual representations of the end product, the tools are typically unsuitable for aesthetic expression, technophobia is high among both students and tutors. Overall, the struggle versus outcome curve is rather steep and, at times even, cannot justify the effort.

The field of architecture, counting for a relatively small share in the software market, used to be offered generic *computer aided*

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

architectural design (CAAD) tools featuring customisations that the software engineers believed to be helpful, important or even crucial for architects as practitioners. Admittedly the typology, abilities and range of tools available to architects has improved over the last decade, *Building Information Management (BIM)* being amongst the crucial milestone. The above maybe true to a great extent (and especially in Greece) however CAAD is not the only digital means that has somehow “infected” the design process.

Time Based Media (TBM) is a field that architects often “borrow” from, in order to investigate and express visually their ideas through scenario/timeline, research scope and expressive abilities. The available tools in this field are plentiful and intuitive supporting to great extends parts of the design process. The integration, aims and importance of TBM in architectural education has been extensively discussed, presented and analysed (Charitos et al, 2001). Linear narration is an area that TBM excel, although, in authors’ experience, non-linear storytelling and the resulting mental shifts are areas that TBM are typically lacking. Moreover they confuse and set back students who attempt to make a move towards using interactive media.

Another step beyond the rigid, inflexible and awkward in use form/design creation tools that have taken varied departments of architecture by a storm over the last decade is *digital fabrication* another research and production direction slowly incorporated in academic curricula worldwide. Form generation and complex spatial assemblies demand suitable (highly specialised) construction workers as well as building sites capable of supporting such designs. However, all this knowledge is often fragmented and generally unrelated to digital design methods.

As far as Greek departments of architecture are concerned, digital media are only partially integrated in the curricula. The department of Architecture at the UoTH was amongst the first to introduce digital media in the curriculum in Greece as part of its original concept back in 1998. However, still the curriculum seems resistant towards integrating more innovative digital media in the design studio such as TBM, digital fabrication and parametric design.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

2 Moving from static to interactive design

At the same time, the evolution of *ubiquitous computing* that we experience in all forms of human activity (Greenfield, 2006) is also addressing the field of architecture (Bullivant, 2006). The responsive building not necessarily in the closest sense of the term (as discussed by Sterk, 2005) nor in the environmentally oriented approach but on a higher level is a concept generally lacking from academic discourse. There is also an increasing interest in the design and implementation of interactive exhibitions, public art installations (Bullivant, 2007), employing such technologies, utilising mobile devices, locative media, sensing and reacting systems, augmented reality, etc. Such attempts support the need for a holistic/inclusive approach strongly featuring interactivity as a core design component, making it the focus of the design process itself. Summarising, digital media have been used over the last few decades to address several thematic areas such as:

- Architectural design
- Physical design
- Narratives using text or TMB (Coyne, 1995)
- Representations using TMB
- Virtual Environments/synthetic space
- Programmatic design

What is indeed missing from the above mentioned tools, methodologies and approaches is a thorough integration of the concept of design change (as transformation, motion, development) together with interactivity in a systematic and coherent manner. Design issues on the environmental performance of buildings, design and implementation of smart homes/environments, information visualisation, implementing dynamic spatial structures representing non-spatial flow/occupancy/interaction datasets are amongst the topics suited to this systematic approach. The authors have been involved in a series of such projects ranging from interactive art installations to smart buildings and to educational virtual reality or multimedia environments and have gained experience on real-time systems, interactions, scenario building, coding/programming, implementing and above all human computer interaction (HCI).

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

Apart from “static” tools, digital media have energised *interactivity* a notion distanced from architectural design often linked to human computer interaction, game design, education, training and various forms of communication. Combined with synthetic space design, as in virtual environments, this topic is taught by the first author at undergraduate level at the Department of Architecture, UoTH since 2001 (Bourdakis et al 2002). Course aim is to understand the rules of digital design distanced from any real space scenario brief, enhance problem solving skills, comprehend and design for interactivity as defined and applied in Virtual Reality settings. Sensors, actuators, programming scenarios (Fox et al, 2009) are the issues dealt within the design process and are core aspects of control technology. In this course, designing interactivity, responsiveness and user or machine generated feedback are core components. This demands that architect students accomplish an attitude towards truly inderdisciplinary knowledge where they have both to master and problematise what they need to learn from varied disciplines such as computer programming, design theory, play/learning theory, semiotics, socio-cognitive science, game theory etc. Although, the nature of such interdisciplinary (or hybrid) work needs special attention, the aim of the paper will focus to analyse merely the potential of a new set of tools (i.e. the blender suite) in architectural curricula that will implement interactivity and integrate it into spatial design leading to a holistic approach promoting intelligence, hybridity and responsiveness of the built environment. In the following, a brief discussion of blender suite as both a tool for designing interaction and gaming in a virtual space context and a tool for introducing student architects to control technology will be outlined. In addition, a number of projects as carried out by architect students will be described.

3 Blender as a tool for programming interaction and gaming

Blender is an open source highly integrated 3d graphics production suite, under the GPL licence v.2 [FSF]. It incorporates 3d design software along with advanced simulation capabilities and an interactive 3d game engine. It was chosen as a companion tool to the array of tools already used and described above as it offered a clear and comparatively easy path for the students to accomplish complex tasks previously unattainable in VRML97. The students used blender 3d in two ways; either implementing their course work solely in blender, or just modelling everything in another 3d package and importing it to blender to add intelligence/logic/behaviour. As such, blender is also a powerful ‘control’ technology tool. Using blender from start to finish proved to be a not-so-difficult (and at times easy) way for the students to build their projects, since the constructed geometry did not suffer from the typical triangulation/tessellation/normal direction problems often occurring during data conversion.

Once the geometry was in place the student would start using the blender game engine (BGE) to build the desired ‘avatar’ behaviour. The BGE is a full game engine with a visual procedural programming interface, and object -oriented bottom up design approach, incorporating accurate physical simulation. The games build in the BGE do not need any compilation, even though it is possible to compile binaries for a specific platform (ie: Windows, Linux, MacOS X etc). The games run within the editor interface with a single key press, making it easier for the student to go through a trial-and-error cycle testing his creation. The interface through which the user builds the interactive part of the game is called "logic bricks" and it contains three main sections: sensors, controllers and actuators (Figure 3). The sensors sense when something is happening and pass on the information to the controllers which parse the information based on logical operations setup by the user: AND,OR,XOR etc acing as the brain of the object. Then the information gets passed along to the actuators which implement the appropriate action setup by the user. An object can have any number of sensors, controllers and actuators allowing for the programming of very complex behaviour. Essentially the user needs to program each object for a specific behaviour, as well as, the interaction between objects themselves and objects and the user creates the interactive experience for the final product – the virtual world.

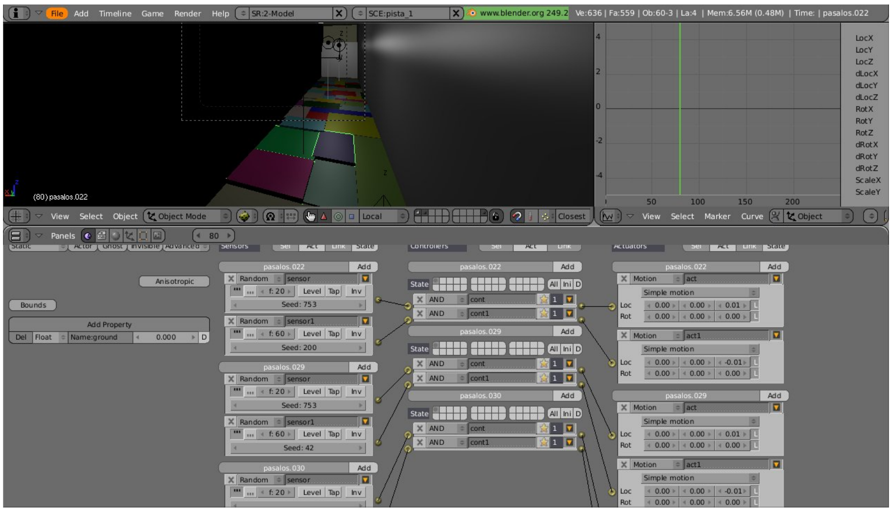

Figure 1. Logic bricks/ programming interactions in Blender 3d

4. Analysing students' work

The 147 submitted student projects over the past eight years –as part of the aforementioned course taught by the first author in the school of architecture- provide a wealth of information for an in depth analysis of genres of VR (virtual reality) as well as modes of interactivity, spatial concepts, etc. In order to accomplish this task, an analytic framework must be defined and the VR genres analysed. Since no such work has been carried out regarding VEs classification, analysis is founded on Wolf (2005), who builds upon Chris Crawford work (1982, in Wolf 2005). He classifies video games based on the iconography versus interactivity ratio and identifies a vast number of genres. As virtual reality worlds relates to video games worlds—both iconography and most importantly interactivity are two essential axis of discussion throughout the course. Following this line of thought, students' project work has been briefly categorised around the iconographic vs interactive spectrum and several genres, that represent the project content, were identified such as: abstract, animation, chase, collecting, demo, educational, escape, explorative, fly, information visualization, maze, platform, puzzle, representative, simulation, sound/scape.

Table 1 presents the frequency of occurrence for the various genres analysed and assigned in the students' work. Note that each project

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

may have more than one assigned genre and there is no primary genre assigned due to the complexity of the projects and difficulty in making an objective judgement.

Table 1. Assigned Genres of VR worlds for the 147 student projects

Abstract	48	Fly	6
Animation	7	information visualization	8
Chase	4	Maze	24
Collecting	15	platform	27
Demo	7	puzzle	12
Educational	15	representative	25
Escape	18	simulation	74
Explorative	110	sound/scape	27

Bearing in mind the focus on interactivity, it is interesting to note that on Table 1, 110 projects are classified as (among other genres) explorative, 74 as simulations and 48 abstract spaces whereas only 12 are puzzles, 15 educational and collecting and 18 escaping—the more interactive genres of all. Despite the effort in helping and pushing the students through the interactivity threshold into designing re/interactive 3D environments, a substantial percentage of work completed falls behind the aims of the course. The following discussion will facilitate establishing the reasons behind this and identify ways forward.

A prior selection was focused on the 12 works presented as part of “myTube”, the department of Architecture, University of Thessaly contribution to “Visions2009”, the 3rd Spot on Schools exhibition, part of the Beyond Media series of events that took place in July 2009 in Florence, Italy. A static image of these selected works (see http://www.arch.uth.gr/visions2009/media/VR/index.html) can be seen in Figure 4.

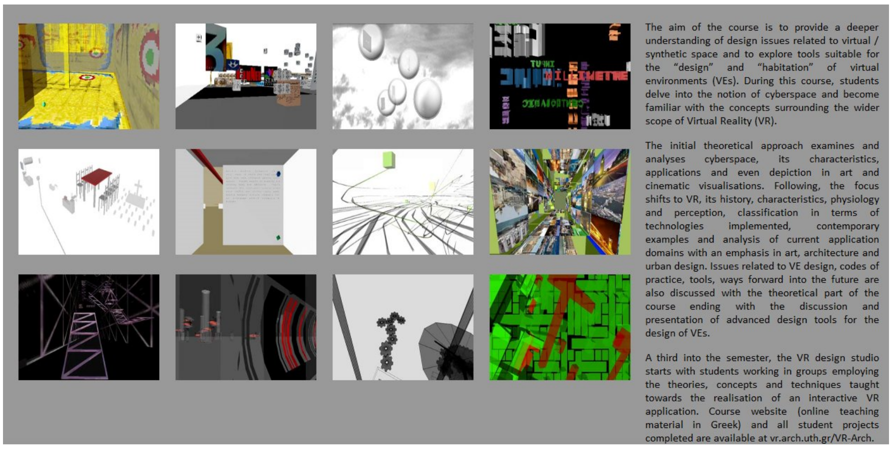

Figure 2. Selected works presented at the Visions2009 Exhibition, July 2009, Florence.

"Disorientation", is the first on top left (see figure 4) and focuses on interactivity in a game-like environment whereas "Linked Constrained Sound Nodes" creates an elaborate soundscape linked to an ever changing spatial synthesis. Regarding the blender 3d works, two out of the five submitted were selected and are also discussed here; the first one is "Laser" presenting a high degree of interactivity in a platform environment and the second "Unsuitable light" exploring space and light interaction carefully synthesising an atmospheric spatial audio-visual experience. In addition, five more respresentative works will be briefly discussed in the following.

A) Laser

The pprject 'laser' is a 3D spatial experience game with the "player" having to avoid the laser beams that will "kill" and/or rotating/moving/shifting planes that one has to walk to complete the "level". Using terminology and metaphors strongly based on video games, the students tried to explore and re-create a game like experience (http://vr.arch.uth.gr/VR-Arch/project.php?id=132)

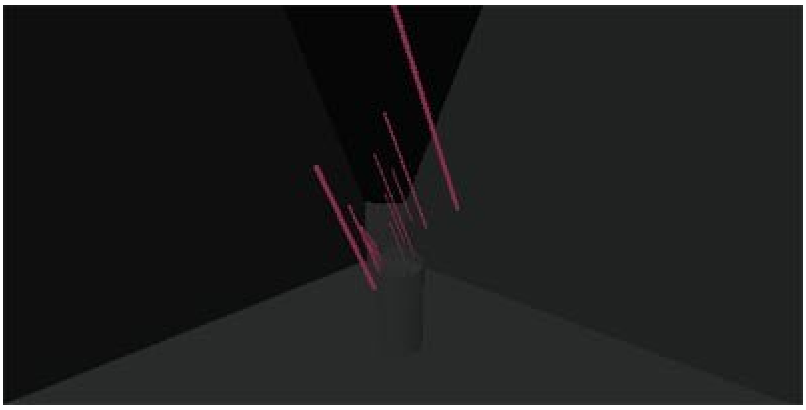

Figure 3. Laser

The initial aim of the students' team was to implement a virtual adventure taking place in a snow-clad mountaintop, with the player evading obstacles using skis or a snowboard. As it was deemed not feasible to design and implement such a scenario within a single semester, the team decided to develop a classic adventure game with inspirations from adventure films like "Indiana Jones". They decided to have the player represented by a character in a third person perspective- one of only three projects following this approach (only readily available in Blender 3d) . Their basic scenario employed a series of rooms or situations filled with moving obstacles that their character would have to overcome to reach the end. The students started calling the main character a "hero" who adopts the stereotypic masculine mission to overcome the obstacles in his path, and at the end of the game to accept the reward of sailing into the sunset with the "girl".

The study contains 4 virtual rooms; the player has to traverse a room linearly trying to overcome the obstacles and in some of the rooms avoid being hit or touched by anything as the room and level is reset at its initial condition. The first room contains a "moving" floor: cubes of various heights move randomly, rising from the floor at times obstructing the player from moving forward or at other times helping if the player happens to be on top of an upward moving cube. The second room has lateral moving planes that obstruct both the player's moves and view: the player must move in an adjustable slalom course to avoid being touched by the planes. The third room has an empty pit in the middle of the room and the player must move on top of horizontal

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

moving planes that act as bridges. The player has to align the moving platforms to pass from one to another and avoid falling down. In this room a subtle use of the physics of the Blender game engine can be observed: the character gathers lateral movement inertia perpendicular to his motion forward, hindering the change from platform to platform as they move in opposite directions. The final room features "laser" rotating perpendicular to the main path. The player has to manoeuvre around the lasers, taking into account the height of the character in trying to find suitable voids.

The main difficulty that students had in realising this scenario was its simplification in a feasible technical model. They started with a lot of "cinematic" inspiration, trying to employ explosions and other impressive visual effects. Although, they started with limited knowledge of blender functionality, they progressed well enough during the semester as they were able to adapt their scenario to blender's technical potential. Concluding this project could be considered amongst the best examples of student work on chase driven platform exploration on an elaborate and rich spatial environment.

B) Unsuitable light

A more realistic simulation of spatial qualities has been attempted in this project. Here, a spatial audiovisual experience is created mapping sensory inputs to unfamiliar methods thus creating an exaggerated sense of being and feeling with light and sound. The team of students in this study wanted to explore the relationship between light-shadow and sounds. They wanted to create an unfriendly environment where the character/player explores these relations. Early on the design stage, their agenda featured a strong architectural concept of an underground space, partitioned by concrete walls in "random" arrangement, while rectilinear openings in the roof act as skylights bringing light inside the underground space. In general they wanted the underground space to be cold, dark and uninviting. They wanted to pair uninviting or disturbing sounds to the light, thereby creating a discomfort to the user: by being in the dark it is natural to seek the light areas of the space but as soon as the player would go towards the light disturbing sounds are triggered.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

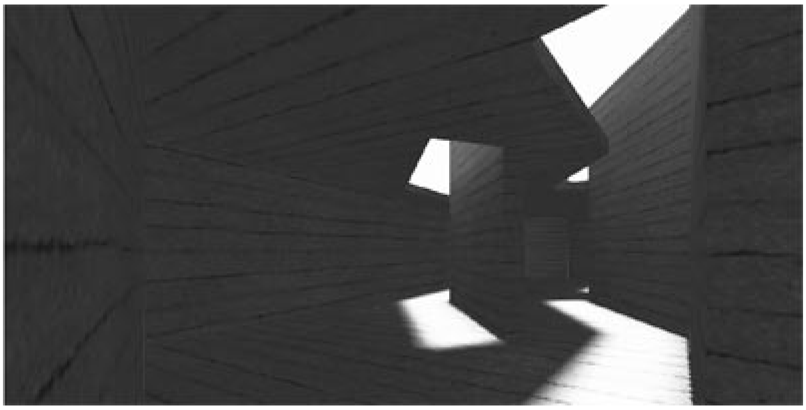

Figure 4.Unsuitable Light

The technical implementation was in general straightforward. The team was proficient in 3d modelling and image processing tools (AutoDesk 3dsMax and Adobe Photoshop) but not blender 3d which they picked up fairly quickly. The major problems during implementation was designing all the proper details in the virtual world to make the space truly uninviting and disturbing. For example the students, used to the automated texturing process of 3dsMax, had to learn the procedure of properly UV-unwrapping a 3d wall and applying the desired textures onto it. As long as sound was concerned the team had no problems implementing their idea, but had to resolve to a trial and error approach in placing the sound actuators/sensors, so as to be triggered and cover all areas planned. This was achieved by testing different configurations trying each time to navigate within the model. Although this virtual world seems simple in its construction and underlying idea, the detailed tectonics employed and the architectural quality of the space combined with the surrounding sound when approaching the light convey to the player a complete surreal experience. This is the only one of the four projects analysed where students made an attempt to create a “realistic” virtual environment with shadows and atmospheric effects aiming towards creating a truly explorative navigational space.

C) Simulation vs exploration and learning vs gaming

Although, there is no space here to discuss all projects, we felt important to refer to a number of projects that attempt to simulate spatial relations that range from realistic to abstract and virtual. These

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

projects at times emphasize aspects of simulation or exploration and at times create a pseudo educational devise aiming to create a context for interactive learning or/and play.

Figure 5. Instruction Manual

For example, the instruction manual (see figure 5) is a simulation/educational manual for kids. The project did not manage to achieve the overall task of explaining electricity to primary school children due to time and complexity issues. The initial brief set by the students was '*….to present in a natural maner (not utilising high degree of visual realism) the phenomenon of electricity*', the implications of electrical safety was eventually oversimplified with a typical room setup where next to a table the user could see a steel object (looking like a sword/being a sword) which the user (a 3D white hand) could pick up with a keystroke once the hand was within a certain range and following the user could start exploring the space "poking" objects. Once on the wall plug socket, an explosion and sound denoted the potential risk. Considering their initial plan to create a project suitable for early primary activity, we felt that this project was in need of further work and elaboration -undoubtly beyond the skope of the course setup but definitely within the scope of a collaborative cross discipline course (see Bers and Portsmore, 2005).

ISBN 978-3-00-032863-3
pp. 606-623

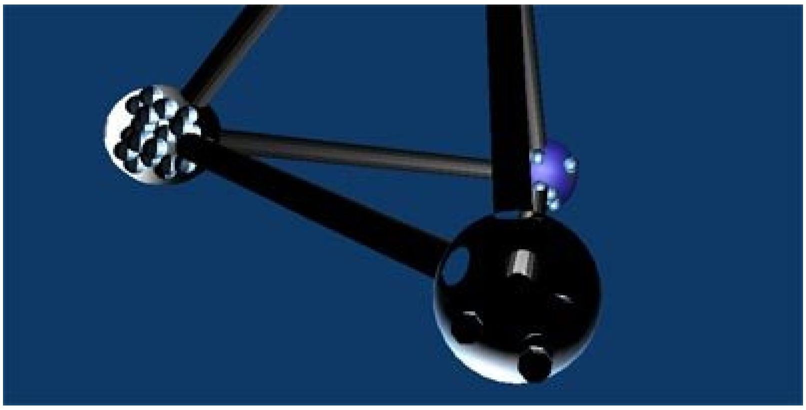

Figure 6.H2O - Water

The project H2O – water (see figure 6) was another attempt at simulating a realistic condition albeit a rather abstract one in the sense of addressing the natural, familiar element of water (H20) and trying to present it in its three states (solid-ice, liquid-water, gass) together with a forth one-plasma- where the previous three co-exist.

Painting Balls (see figure 7) was initially envisaged as a tool for primary school children comprehension of colour mixing and creation. It utilizes simple maths in order to create the mixes using the RGB colour space as known on computer applications.

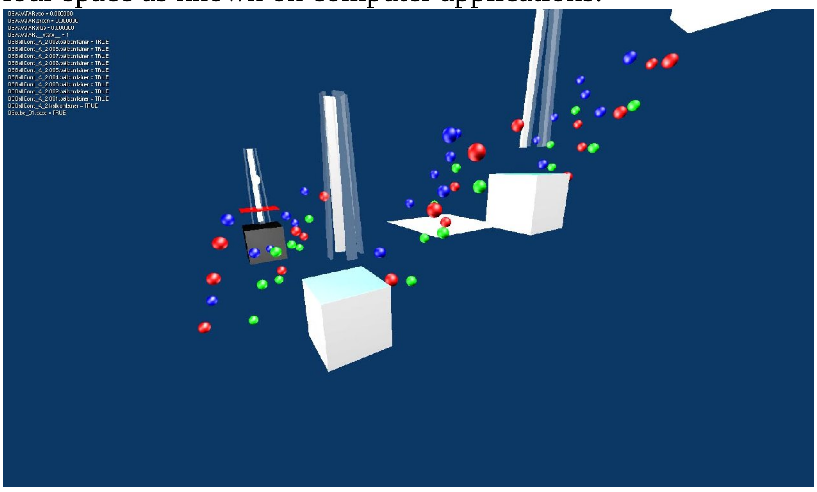

Figure 7. Painting Balls

ISBN 978-3-00-032863-3
pp. 606-623

The overall environment is “littered” with prime coloured balls (red, blue, green) and a series of main box/platform units. Users are confined to certain preconfigured views and are trying to match colours selecting the right number and type of balls. The platforms scale on X,Y,Z axis according the colours added in the list and thus setup the overall paths for exploring the space. The link between colour and size is quite powerful and useful in linking the two phenomena and creating a cognitive map of the space and process. Overall this project features todate the strongest concept and most inclusive implementation of an educational interactive application created in this course.

Music Pan (see figure 8) can also be classified as educational since the work demonstrates in an intuitive manner the process of analyzing and disassembling a music score, and in a playful way recompose/reassemble it. The interface using gravity on a custom shaped “pan” with a ball rolling utilizing real physical properties of friction, etc is challenging and rewarding.

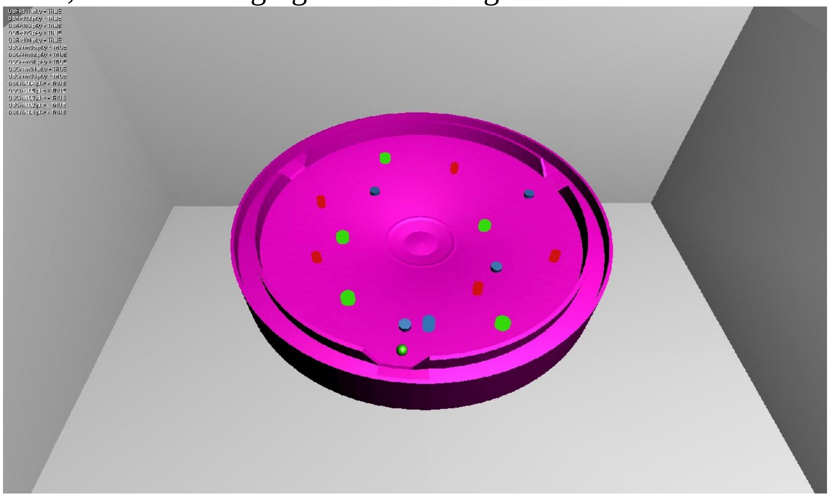

Figure 8. Music Pan

The Spinning Cube (see figure 9) was one of the best attempts to represent a complex spatial navigation in a fairly abstract virtual world that promotes fantasy and playful movements in a space that is constantly rotated in a chaotic way.

ISBN 978-3-00-032863-3
pp. 606-623

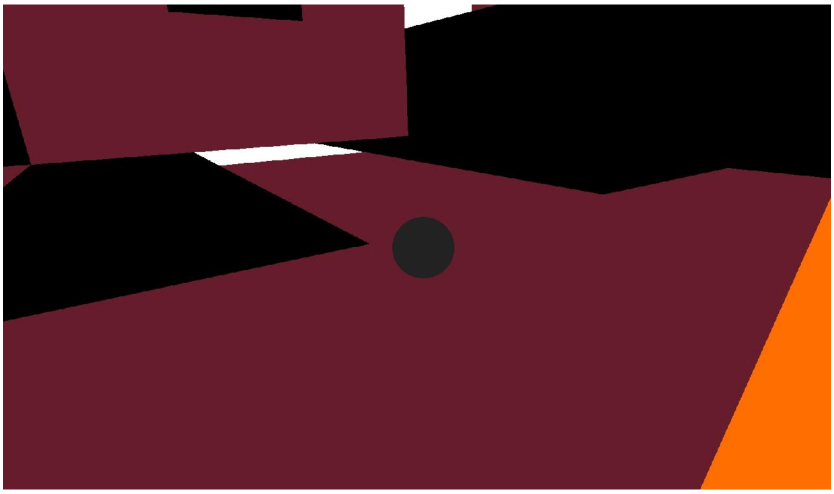

Figure 9. Spanning Cube

7 Conclusions

Blender implements dynamic lighting and shadows as well as classic texturing capabilities. These features enabled some of the students to incorporate lights and shadows as a design feature or even base their whole virtual world on interaction with sound, light and shadow. Even though interactivity with the world in some of these cases was reduced since the students aimed for a better visual result, the virtual worlds they designed were complete. In this the physical simulation of the BGE helped a lot: With a full physical simulation system BGE incorporates the user does not have to programme elaborate sequences, but just programme or adjust the environment to its purpose. For example the blender game engine stars with a default gravity acceleration of 9.85, a value close to real-world gravity. The user can change this number and adjust the behaviour of all objects in the scene instead of having to adjust each one individually. Another example would be the collision engine, where the user can create collision boundaries in the world that are obeyed by every object in the scene. A third example is the creation of recorded animation using the BGE: the user constructs his simulation, for example, cubes rolling on a hill, and records this simulation as an animation inside blender. When execution time comes animation proves economical in CPU processing consumption since it just plays back the animation and does not recompute everything. As the eye of the user though the animation is natural and the student did not have to put in a lot of work to make it appear natural. Again for CPU processing economy the user may elect for some object to be excluded by the simulation if their function as a mere prop or decorative element in the virtual world.

A third feature which helped students build some advanced interactivity was blender's Python scripting capabilities. Python is an object oriented open source interpreted language. Blender incorporates again a python interpreter where the user has access to almost every function in blender. Python scripting was outside the goals of this class, and would be practically impossible to teach since the students have already a full workload with theory understanding and implementation of their virtual world. However, it was decided that students could use readymade scripts and customize them when extra complicated functionality was needed. Usually the scripts used would just supplement the procedural workflow of the "logic bricks" interface, instead of controlling objects on their own. Students were provided with readymade scripts, either from the general blander community or written by the tutors of the course, along with explanations on how to use and customize them.

Usual problems the students had with this approach were the difficulty in developing a bottom-up approach in designing their virtual world. Their scenario was usually created top-down and in blender they had to recreate the scenario by configuring each object and the environment separately. The same difficulty in shifting from top-down to bottom up appears in all game engine or virtual reality tools that employ an object oriented, bottom up approach. The benefit of using blender in overcoming these difficulties was that the design-implement-test cycle is shorter and more direct, since the user goes from implementation to testing with just a key-press, without having to load any third software to experience the virtual world.

This counteracts the disadvantage of the bottom-up approach complexity: by facilitating the design-implement-test cycle, blender allows the student to test his or her design without going through a complex work-flow using multiple tools. Of course one has to mention the fact that blender's unique interface and the procedural bottom up

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623

approach forces the students to learn esoteric terms and theory on how a program or software functions. This does improve their knowledge on exploiting software to design a virtual world but imposes a penalty on the effort required by the student.

One clear benefit from the tutor's standpoint is blender's open source nature. During the course where blender was first implemented, the authors tried unsuccessfully to use blender's game engine with a VR head mounted display with a 6 degrees of freedom tracker. The plan was to test the designs in an immersive environment right in the studio. Blender's engine produced every other video output format than horizontal interlace needed for the i-glasses. With no knowledge on video driver output and interlace setups, the authors posted information on the functionality needed on blender's development and artist forums and four months later the functionality was quietly implemented by the blender developers. This reinforces the notion that open source software tools are more appropriate for teaching a software depended course. Depending on the resources available and the communication with the core developer team, new features can materialise easily and following the much needed testing can be incorporated in the future software releases.

References

1. Bers,M., and Portsmore, M. (2005). Teaching partnerships: Early childhood and engineering students teaching math and science through robotics.Journal of Science Education and Technology, 14(1), 59-73.
2. Bourdakis, V., and Charitos, D., (2002), 'Teaching Virtual Environment Design', in S. Wrona (ed.) *Connecting the Real and the Virtual design e-education, eCAADe2002 Proceedings*, eCAADe Publications, pp. 42-49.
3. Bourdakis, V. *Virtual Reality: Designing Synthetic Space*, Course website [in Greek], URL http://vr.arch.uth.gr/VR-Arch/
4. Bullivant, L., (2007), '4dsocial: Interactive Design Environments', *Architectural Design*, Vol.77 (4).
5. Bullivant, L., (2006), *Responsive Environments: Architecture, Art and Design*, V&A Publications, London.
6. Charitos, D Pehlivanidou Liakata, A Bourdakis, V and Kavouras M., (2001), 'Time Based Media as a Means to Enhance Spatial Representations'. In H.

Penttila (ed.) *Information Management: eCAADe2001 Proceedings*, eCAADe Publications, pp. 233-238.

7. Coyne, R., (1995) *Designing Information Technology in the Postmodern Age*, Cambridge Mass., MIT Press.
8. Fox, M and Kemp, M., (2009), *Interactive Architecture*, Princeton Architectural Press.
9. Greenfield, A., (2006), *Everyware: The dawning age of ubiquitous computing*, New Riders Publishing.
10. Sterk, TD., (2005), 'Building upon Negroponte: A hybridized model of control suitable for responsive architecture', *Automation in Construction* 14(2), pp. 225-232.
11. Wolf, M., (2005) *Genre and the Video Game*. In J.Raessens and J. Goldstein (eds) Handbook of Computer Game Studies, MIT Press, pp.193-204

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 606-623