Sketching in Virtual Space with Voxels: A CAD tool for the Early Stages of Architectural Design

André M. Duarte$^{ac}$
Supervisors: Daniel S. Lopes$^a$, Joaquim A. Jorge$^{ac}$, and Rodrigo M. de M. Ventura$^{bc}$

Abstract—Early scale modeling is an essential part of the architectural process. It consists on creating small rough sketches of the building being projected, which are used for spatial perception, testing shapes and volume distribution. The more traditional approaches to scale modeling consist on creating this maquettes using physical materials (e.g., styrofoam, balsa wood, cardboard). This limits the speed at which an architect can work and becomes increasingly complex, expensive and time consuming with the size of the project. On the other hand, CAD software uses mostly 2D interfaces that heavily rely on windows-icons-menus-pointer elements (WIMP) which are not suited for early stage scale modeling due to their complexity and because, as many papers show, they affect the architectural creative process by having a steep learning curve and more tedious tasks. This paper explores the use of Virtual Reality as a medium for early stage scale modeling, at different scales, that will improve on analogue and 2D computer based approaches. A spatial user interaction system was created using a motion capture setup, a videogame controller for user input and a head mounted display (HMD) for creating a virtual reality environment. The application developed, Maquetteer, uses voxels, cubes constrained to an orthogonal grid, as a modeling unit to create early stage maquettes. The proposed system was tested by both laypeople and professional architects and results show that it is easier and faster to use in comparison to a commonly used WIMP modeling application.

Index Terms—Virtual Reality, 3DUI, CAD, Sketching, Scale Modeling, Architecture

I. INTRODUCTION

Sketching is essential to the creative process, it allows the designer to try and explore ideas quickly and to relay them to others. It is a task that is present in almost any project that involves some kind of creative thinking including architecture projects. There are analogue approaches to sketching in architecture such as pen and paper or scale modeling and digital ones that are called CAD software that can be used in computers, tablets and other electronic devices. Sketching, either by digital or analogue methods, are considered a good practice in architecture as it promotes the discovery of relevant concepts [1][2].

Scale modeling consists on creating different size replicas of the final project as a way of knowing how will it look like, studying different topologies and knowing how the structure will interact with the environment. An architect produces many scale models (also known as maquettes) during the lifetime of a project. At the beginning of the project they have very few details attempting only to see how the overall shape of the construction will be, what volume will it occupy and how that volume can be arranged in the structure’s target location. As the project advances and the shape of the construction becomes well defined the scale model starts to have more details until the final prototype which is the one more similar to the end result. This collection of maquettes can be seen as a window to the architect’s creative process and help the architect relay its vision to the client and colleagues. Traditionally scale models are made by using materials such as cardboard, styrofoam or wood which are then cut and pieced together to create the desired shape. It is a laborious work that can take several hours to finish and is limited by the resources available, by the number of people working on it, the size of the room and the physical properties of the materials used.

As computers evolved digital alternatives to analogue sketching began to appear such as AutoCAD, Revit and Rhinoceros. Although they are considered powerful tools in architectural design their interface relies heavily in WIMP elements and due to their complexity learning them is a very time consuming task. Furthermore, this kind of tools can have a negative impact on creative thinking [4] which is essential for the design process specially on early stages. These characteristics make this kind of applications unsuitable to be used in early stage sketching when the user wants to create content and test different approaches quickly without great detail. A software that differentiates from this norm and that tries to be more approachable is SketchUp [5][6] which uses surface push-and-pull methods making it faster to use and easier to master. Finally, another application worth mentioning regarding this topic is Minecraft [7] which, despite being a videogame, serves as proof of concept in using voxels as a modeling unit for buildings and another architectural structures [8][9]. Voxels are cubes that are constrained in position and size to a grid (much like their 2D equivalent, the pixel). Minecraft is one of the most successful games of all time [10], it is seen as a potential learning tool [11] and some really complex constructions were made with it only using a very simple 2D interface and voxels.

There is more evidence that voxels/cubes are very important modeling tools. A great number of toys use cubes as building blocks for improving spatial thinking [12] and creativity. Those kind of toys may have influenced current architectural styles [13] and some of them, such as Minecraft [7] and Lego [14], can be used as modeling tools.

$^a$INESC-ID Lisboa, Portugal
$^b$ISR Lisboa, Portugal
$^c$Instituto Superior Técnico Lisboa, Portugal
for early maquettes. The constraints of this kind of tools, where all the blocks have to be orthogonal to one another, is not as restrictive when applied to early stage maquettes and curiously enough roughly 90% of all modern buildings have orthogonal layouts [15].

Despite Minecraft and SketchUp being easier to master they still rely on WIMP elements and on a 2D interface which is unsuitable for a task as dependent on spatial thinking and visualization as architectural design.

With the recent growth in popularity of Virtual and Augmented Reality the number of VR devices have exponentially risen. Companies such as Facebook, Samsung, Microsoft, Valve and many others have built affordable HMD’s that provide VR/AR experiences that are truly immersive while being small in size and portable. These characteristics make them great tools to produce 3D interfaces for improving the way we interact with digital content. Being architecture very dependent on spatial notions and reasoning this kind of interfaces can improve architectural practice in a way that 2D applications could not by allowing direct manipulation, on different scales, and to more rapidly build a maquette [16][17][18].

The solution proposed is called Maquetteer and presents a modeling paradigm, very similar to Minecraft, based on voxels which are, as mentioned before, cubes that are constrained to a grid in terms of size and position. Because the cubes are very constrained (orthogonal orientation and snapped to a grid) the actions that the user can perform are very limited making this system very easy to learn. Those actions are mainly associated to creating and deleting cubes using a 3D pointer that mimic some of the actions normally associated with physical scale modeling such as cutting or gluing cardboard. On the other hand, because our main focus is on early stage maquettes and most buildings are predominately orthogonal the limitations are not too restrictive because the architect can, in theory, create anything that he desires within that scope or at least not feel that the application is not burdening to much the creative process. This way we have a tool that is both powerful and easy to use. Despite Minecraft and Maquetteer having very similar approaches to modelling the 3D interface of Maquetteer and the integration with VR allows for a greater degree of freedom, added functionality and immersion for the user.

The application was tested in three phases. In the first one by laypeople, where the focus was on the interface, how difficult was to learn to use the application and how fast could content could be created in comparison to other applications. In the second phase the testers were professional architects and the objective was to infer if this tool could be used during the architectural process and what could be improved in that perspective. The final phase of tests was performed by students and professionals of architecture which tried to use Maquetteer in a collaborative setup.

II. RELATED WORK

One of the most successful and widely used CAD application is AutoCAD. It started as a 2D technical drawing software and it expanded with other functionalities such as 3D visualization and modeling [19]. Another commercial hit was the software SolidWorks which focused on 3D solid modeling [20]. There are many other alternatives to these softwares such as Rhinoceros [21] and Revit [22] which follows the BIM (Building Information Modeling) philosophy. BIM is the representation of not just the elements of a building but also spatial relationships, light analysis, geographic information, and quantities and properties of building components in a digital and structured way [23]. As it is discussed in Mateus [24] this CAD tools are not suited for the early stages of the architectural design, where sketching is often preferred, because they are too complicated to use and to learn to easily test different ideas rapidly. One software that tries to be faster to use and to have a smaller learning curve is SketchUp [24] which is a 3D modeling CAD tool. It is a step on allowing digital sketching to be as intuitive as its traditional alternatives but it still relies on a WIMP/2DUI making it slower to master and use while also hampering spatial thinking and creativity.

One of the simplest ways of 3D sketching is painting in space. Typically, a user maneuvers a 3D pointer that it can use to paint the virtual space with strokes mimicking a pencil or a brush. Among the implementations of this kind of paradigm is the project “Sketching in Space” [25] from The Constitute and the concept video “Step into the Page” from Disney [26]. This are simple approaches and are not suited nor even produced to be used in architectural design but worth mentioning as a base line.

In Okeil [27] the authors proposed a hybrid system with immersive and non-immersive tools. The non-immersive tool, Sketchup, would be used for modeling while the immersive environment application would only be used to visualize and navigate the building in 3D. The changes on the model in sketchup would translate in real time to the model in the VRE. The framework used was composed by a computer with a mouse and keyboard connected to a CAVE system. A CAVE consists in between three to six projectors directed to different walls off a room, in this case three, that merge together to form a virtual environment. This solution to CAD for early stages of the architecture process allows the user to visualize in 3D and in an immersive environment what he is sketching, improving his perception of the final product. Despite that, this technique has a lot of shortcomings by being constrained to conventional 2DUI applications for modeling, the constant transition between the non-immersive and the immersive environment can be tiresome, confusing and breaks the immersion and for every 2D CAD tool used there must be a software associated that can translate the CAD format to a format suitable for display in an immersive which can lead to loss of information.

Another CAAD tool developed on a CAVE framework was the one proposed by Hughes [28]. CAVECAD is a fully immersive architectural modeling tool that allows the user to produce low detailed buildings. The modeling process consists on creating basic shapes by selecting them on a 3D widget and arrange them in the desired shapes. The user can also manipulate those geometric shapes, parallelepipeds, cylinders and cones, by moving ore rescaling any of the solids faces and thus create new shapes. There are snapping conditions to improve object positioning and the interaction with the environments was done with a 3D wand. User tests were conducted and although the authors considered their results favorable the users reported difficulty in finishing the test’s task on time, even those with a lot of experience with the tool, and the user sample has poor quality since it was small, only four people, and two of them had experience and included the CAVECAD programmer. Despite their shortcomings this solution allows for a quicker modeling
paradigm than current CAD tools by using a 3D interface and by sacrificing precision.

Sasaki [29] introduced the concept of “facetons” as a modeling paradigm for a spatial interface. Facetons are infinite planes that are represented by a small widget that indicates their position and orientation and is responsible for the interaction with the user. When different planes intersect with one another they create finite surfaces from those intersections. By creating and intersecting many facetons it is possible to rapidly make complex forms. The spatial interface and immersive environment allows for the user to place the facetons with a high degree of precision and to be able to better understand the intersections of facetons required to create the desired shape. The user is also able to inspect his creation as if he was inside of it which is something that is normally not possible on early stages of the design process. Sasaki [29] presents a very good example of a CAAD tool for immersive modeling in the early stages of design allowing the user to rapidly develop models and interact with them as if they were in real scale using only very simple operations of drag and drop. This is true even for users who have little to no experience with CAAD tools or other modeling tools. However, the solution proposed shows one major flaw: the intersection of the facetons can become ambiguous to the user. When the number of facetons increases the number of possible intersections between planes increases and there is more than one solution for the final shape, which is a problem, and caused the major complaint by the users during the tests to be that, a lot of the times, the expected result and the real result did not match.

The following two approaches [30][31] have the same basic modeling paradigm, constrained modeling using voxels using a spatial interface in a VRE. Assuming that the size of the voxel is big enough, which is true for the papers in question, it is possible to create content in a spatial interface with high precision and quickly although having to sacrifice the possibility of highly detailed models. This loss of details is not a shortcoming of the tools because they were meant for 3D sketching and the early stages of architectural design, a stage of design where details are not relevant. Vries [30] introduced the concept of voxel modeling in VR as an architectural tool and also tried to diminish the problems associated with the voxels restrictions while Kuan [31] improved on this concept by introducing grouping of voxels in more complex structures that could then be edited and moved as one. Although both tools were conceptualized to be used on a VRE they both display a 2DUI with WIMP elements that does not take advantage of the VRE. The user tests were made using a keyboard and a mouse. Another shortcoming of this papers, in regard to validating this tools as useful CAAD software, is the nonexistence of professional architects amongst the user testers which could provide valuable feedback.

Leaving the scope of VREs there are two other important works that show the potential of voxel paradigms as modeling tools in the architectural design. The first one is Minecraft [7] which was mentioned before as a game that allowed the user to construct entire cities using only voxels. The second one is Block’hood [32] which was presented in a paper by Sanchez where he describes a game where the objective is to build architectural structures that are sustainable using only voxels that would serve as cells, modules with a specific function within the building. The reason for using voxels was their very simple structure and easiness to model with and to position [30][31][32].

This literature review shows that there is a need for better CAAD tools in the early stages of architectural design and that spatial interfaces allow for the user to explore the space as if they were inside of it [27] [28] [29] [30]. Most of them, excluding Okeil [27], provided means to model buildings using 3D cursors in an immersive space allowing the user to see a 3D representation of the sketch and introduce changes in real time in this representation. Sasaki [29] allowed the user to swiftly model buildings in real scale even if they were bigger than the user. This was one of the better tools presented with a very simple yet powerful interface in a VRE but this simplicity caused the interface to be ambiguous and making difficult to the user to predict correctly the result of their actions. Another conclusion reached was that voxels are a powerful paradigm for low detailed architectural modeling [30][31][7][33] that was already integrated in some spatial interfaces [30][31]. What Kuan [31] and Vries [30] lack is proof that they would work on an architect’s studio and they would improve the set of CAAD tools that currently exist. Both tools were tested using architect students which lack the knowledge and experience to make an informed judgement on whether if this kind of spatial interface can be a suitable CAAD tool for professional studios. That is why the interface proposed on this paper was tested using professional architects. Another improvement of this paper over Kuan [31] and Vries [30] is the fact that it presents a 3DUI that tries to break from the WIMP mindset.

III. HARDWARE SETUP

The spatial interface developed is integrated in a hardware setup that allows for the user to move through the Virtual Environment (VRE) as if it was a real environment.

![Fig. 1 – Image showing the Hardware setup minus the computers.](image)

The user is immersed on the VRE using the HMD GearVR, from Samsung, which has two optical lens for stereoscopy vision and a gyroscope which is responsible for the rotation movements of the head being translated to the VRE. This device comes incorporated with a Samsung s6 with a resolution of 1440 x 2560 which means that for...
each eye, because the screen had to be divided by two for stereoscopic vision, the resulting resolution is ~1440 x 1280. Stereoscopic vision is a fundamental feature because it is what allows to the user to have a 3D and depth perception inside the VRE.

Besides the HMD there is also a Wii remote controller (WiiMote) which is used as an input device and both it and the HMD have special markers on them so they can be tracked by motion capture cameras spread across the laboratory. This information is processed by a computer using a software named Optitrack which determines the GearVR’s and WiiMote’s position. This information and the WiiMote’s input is sent via UDP to the GearVR.

![Image of Maquetteer. The modeled content is divided by white lines that form cubes which represent the chunks in which the voxels are grouped. They do not appear in the actual application.](image.png)

In regard to the renderization of the voxels it is worth mentioning that it cannot be done by rendering them one by one. This would take too much computational time, especially in a smartphone which does not have the same processing power as a computer, and during a normal modeling session, with a reasonable number of voxels, it would have such an impact on the application that would make the user unable to use it. To solve this problem, it was necessary to optimize the number of triangles (triangles are the basic unit of any rendered object) of the modeled content. Due to the fact that all the modeled content is composed by voxels, it is more simple to work with quads instead of triangles which are nothing more than two triangles that form a rectangle.

The solution was to, by knowing where and how many voxels were created, to create a mesh with the same shape as the voxels created but with a lower number of quads. This reduction was done using different criteria: First, all the quads that corresponded to sections of the voxels that were not visible to the user would not be part of the optimized mesh. Furthermore, if two or more voxels are side by side, the faces that are coplanar are merged in to the minimum number of rectangles/quads possible.

B. Modeling Operations

Due to the fact that the voxel is a very basic paradigm and highly constrained this allows for the modeling operations to be equally simple and intuitive. The two main operations are creating voxels and deleting voxels. Creating content can be done in three ways: generating voxels near the tip of the WiiMote in a freeform fashion, creating parallelepipeds of voxels by defining their diagonal using the WiiMote or creating parallelepipeds that instead of being solid are composed by a repetition of a pattern, which is basically a combination of the two previous modes. Deleting voxels works in a similar fashion.

Besides the operations already described there are some auxiliary operations. Content can be selected after being created and then rotated, copied or moved. There are also state operations such as Save modeling session, Load modeling session, Undo operation and Redo operation.
C. Visualizing Content

To visualize content, the user can walk around it as if the VRE was a physical space, due to the motion capture system. The optical lens and the stereoscopical image provide 3D and depth perception.

A method for visualizing content in different scales was also implemented. An auxiliary object with the shape of a person, which can be resized, defines the proportion between the voxel and a person’s size. This tool allows the user to see its sketch in the following scales: 1:50, 1:20, 1:10 and in real scale. When the user chooses the real scale mode the scenery where the user is placed in the application changes for an open space with a day/night cycle.

D. Graphical Elements

The picture above shows the most relevant GUI elements: The WiiMote’s digital representation with text around the buttons that indicate each of their functions, the white panel in the back with an image of the WiiMote and text also indicating the function of each button and the white lines on the floor indicating to the user the area where is safe to walk on. The last element is the environmental context which allows the user model around a physical place instead of modeling in space.

E. Additional Features

Despite the fact that this are important features they were implemented after the usability tests and were not the focus of this work.

The first one is multiple orientations. By introducing changes to the interface the option of modeling in different orientations was added. This was done in a way that respected the constraints of the voxels in a grid so the original intention of having a highly constrained paradigm was not lost.

The second one was collaboration. Through some small modifications to the application different users can be in the same VRE at the same time although they all have to share the same WiiMote. Each user is represented in the VRE by a simplified version of a person (Avatar) which is constituted by a pill shaped ovoid that represents the body and a sphere that represents the head. The head has two small black spheres on it which indicate where the person eyes are with some degree of accuracy conveying to others the orientation of the user’s head.

V. User Tests

The application undergone different stages of user testing. In this section we will present those different stages, talk about what was evaluated, how it was evaluated and what were the results.

A. Tests with Laypeople

In the first phase of testing the users were all people who had almost no knowledge of Architecture and the architectural design process. They were 18 in total with ages from 20 to 29 years old and all of them were Computer Engineering students who had at least one course on Person-Machine interfaces. In addition to this, they have watched 3D movies or IMAX but have not wear HMDs.

The test itself was meant to compare the prototype with a commercially used CAAD software, SketchUp. This software was chosen because it is very widespread and popular and on the other hand is more suitable for 3D sketching than its peers. The test had two different tasks, first the user would model some simple architectural elements to get used to the system. The elements chosen were a pillar, a beam, a slab, a wall with a window and a set of stairs which are elements that can be seen in any building.

After that the task was to model some famous buildings. Those buildings were, in increasing order of modeling difficulty: Empire State Building, Farnsworth house and the Falling Water house. For each example there was a picture (in Sketchup’s case) or a 3D model in the VRE (in proposed system’s case) of the intended final product. In spite of that, it was made clear that there was no need for extreme precision and that an approximate copy would suffice. This tasks were timed and repeated in both systems, Maquetteer and SketchUp, and were preceded by an explanation of their functionalities and an adjustment period of 15 minutes. The order in which the applications were used alternated from user to user so it would not influence the results. After the
test was completed the user answered to a questionnaire about the usability of both systems and in addition to this they were asked to make a heuristic report about the interface.

![Fig. 7 - Basic element that the user had to model. From left to right and from top to bottom: beam, ladder, floor, pillar and wall with window.](image)

![Fig. 8 - Building the user had to model. From left to right: Empire State Building, Falling Water House and Farnsworth House](image)

### B. Tests with Professionals

Five internationally renowned architects were invited to test the system in the second phase. The purpose and methodology were different from the previous evaluation. The objective, in this case, was not to test and improve the interface but to understand if this was a tool that architects would use on the design process, given the chance. Because of this, the professionals were not required to model with SketchUp.

The testing session would begin by an explanation of the application’s hardware setup and the application’s functionalities. After that the user would model freely in space so it could become accustomed to it and also understand better its potential. When the architect was comfortable with the system a 3D scale model, from a real location, would be introduced on the VRE. The objective was for the user to try to design a building in that location using the application and, by doing that, understand how this tool could be used in an architecture studio during the design process.

Once the session was over, the user would answer a questionnaire. After all the user tests were conducted a semi structured interview was conducted with all the professional architects, so a more informal discussion about the application could happen, and even though there were no statistical results withdrawn from this interview, important feedback was received.

### C. Collaborative Tests

The collaborative tests were a way to understand how the application would work in a collaborative context. Two teams of three people participated: one of which was composed by three professional architects and the other one by a teacher and two architecture students. The task was for each team to model in collaboration a building while all participants from that team were in the VRE. In the end of each testing session there was a semi structured interview with the users to understand what were their difficulties on working together and if they had saw an improvement over other forms of design.

### VI. RESULTS AND DISCUSSION

From the testing sessions, described in the earlier sections from this chapter, a lot of relevant feedback was received. Focusing on the statistical analysis of the questionnaires and test results and on the semi structured interviews the more relevant observations and results are discussed here. It is also worth mentioning that some of the feedback received in earlier phases of testing was used to improve the system for latter phases. The statistical analysis of the questionnaires was done, almost exclusively, using the Wilcoxon Signed Ranks Test. This tests allows to compare sets of data which do not have to be normalized.

#### A. Tests with Laypeople

The Laypeople tests addressed two key points: how well it would perform in comparison to similar software which already had commercial success and determining what parts of the interface needed most to be improved. The results of this comparison were favorable, the statistical analysis of the questionnaires indicates that our system was easier to use than SketchUp ($Z = -2.0$, $p = 0.046$) although in terms of learning curve they were very similar. The navigation on the virtual environment was also better ($Z = -2.176$, $p = 0.029$). One aspect where there was no statistical relevant difference was on the perception of the location of the cursor. Table 1 presents the results to the most relevant questions presented to the users about the core aspects of the interfaces which show a slight advantage to Maquetteer. Other positive aspect was the comparison between the examples and the scale models modeled by the users. In almost every case the users agreed that the results were very similar in the two applications being the exceptions task 8 (“Falling Water House”) where the system proposed on this paper was better and that in task 6 (“Farnsworth House”) SketchUp was better but not by a significant margin.

![Fig. 9 - Participants opinions regarding the core aspects of the application. Median (Inter-quartile Range). * indicates statistical significance.](image)
When the task times were analyzed, the conclusion was that Maquetteer add consistently better times such as on the “Stairs” \((Z = -3.174, p < 0.01)\) and “Beam” \((Z = -2.681, p = 0.007)\) tasks. The ones who were significantly quicker than on SketchUp were the two most complex ones from each phase: tasks 1 (“Stairs”) and 8 (“Falling Water House”). The only case where SketchUp outperformed the proposed system in execution times was on task 7 (“Empire State Building”) although the difference was not very significant. This may have happened due to the fact that the constrained measures with which the users had to work did not allow them to keep true to the relation between the proportion of the different elements of the building. For example, the buildings antenna is very thin and was usually the last part modeled by the users and when they reach that phase they would find that it was too big in relation to the rest of the building. Which led to them trying to fix the proportions between the antenna and the rest of the building. It is important to restate that this was only a problem on task 7. That being said, this need for adjustments due to miscalculated measurements may have been the cause for the widespread values of execution times that can be seen in the image bellow.

Another interesting aspect to mention was the fact that, although most of the users moved around the VRE to see the what they were modeling and the examples from different perspectives, some of them were not able to do this remaining static on the VRE. There was even one instance where the user remained on the same place and modeled only a 2D perspective of the task. This resembles a lot the posture associated with WIMP interfaces.

![Fig. 10 – Representation of the average value and standard deviation of the tasks’ execution time (in minutes:seconds) for both systems. Task: 1 – set of stairs; 2 – pillar; 3 – beam; 4 – slab; 5 – wall with window; 6 – Farnsworth House; 7 – Empire State Building; 8 – Falling Water House.](image)

Regarding the interface’s quality analysis, the main source of feedback was the heuristic reports that the users had to write. All users had written a report of sorts at least once before. The fails of the system pointed by the users were compiled to understand where there were the more serious problem and try to solve them. Some problems were not fixed because they had no easy solution. One example was the complaint that was hard to interact with the buttons of the WiiMote even with the digital representation. Another problem was the lack of proprioception, the ability to see one’s body, which is not a trivial matter. Some of the other faults that come up on the heuristics evaluation may have been originated by the lack of proprioception such as the sensation of floating, not being at the correct distance from the ground and difficulty in perceiving distances and proportions in general [43]. Other problems were addressed, such as the complaint about having too much information on the WiiMote’s virtual representation making it difficult to read. Instead of removing or changing the information on the WiiMote the problem was fixed by changing the VRE to something more simple and cleaner. What was happening was that the first VRE had too much unnecessary elements, such as tables and computers which were modeled to simulate the real environment, which added too much visual noise to the background and interfering with the more useful interface elements, like the WiiMote. This decision was also supported by the fact that the realistic setting was not having the intended effect of providing a familiar setting and was increasing the sensation of disproportionate measurements (which was also mentioned on the heuristic reports). Other flaws that were fixed was the cursor having the same color that the generated voxels and the fact that sometimes the cursor was shifted by one voxel from the position where the user was modeling. A positive conclusion withdrawn from the reports was that the system’s interface had no fatal problems and very few serious problems. The changes mentioned were made to this system before the tests with professional and the collaborative tests.

B. Tests with Professionals

In the second experience the subjects were, as mentioned before, professional architects. One advantage that was immediately identified on the interview by them was the fact that they were able to model completely immersed on the virtual space instead of looking at a screen. It helped them visualize things and have a better idea of the relation between the different elements. It also allowed them to model things in 1:1 scale with ease which is something that conventional methods do not allow. All of this was supported by the fact that, during the free modeling phase, the architects modeled around their bodies and many times at real scale. One problem they pointed out however, was the fact that they could not see their body which would have been useful for having a better notion of sizes and proportions by using it as a reference.

In regard to the system itself the general opinion was that it run really smoothly and that the response times were very good. The overall interface was easy to use and was particularly helpful having the virtual representation of the WiiMote with the text indicating the functionality of each button. The high level of satisfaction with the system may have been in part due to the fact that, at this point, the interface was already improved since the first round of tests using the feedback gotten from them.

There were two main complaints about the system. The first one was about the scale of the voxel, when the scale model of the environment was introduced, suddenly the architect was restrained to model at the same scale as it and was not able to change scales. This need was made even more obvious by the fact that, when using the scaling functionality was used, the user was restrained to visualization and navigation. They expected to be able to model at this scales and in fact “requested” that the possibility of modeling at this different scales was added. Despite this, they were very impressed with the visualization at different scales, especially at 1:1 scale, and pointed out that this feature was a very important aspect of the system.
because of the way it simulated a sense of scale and that it could be very useful in architectural education because students usually lack this and it is something which is not easy to convey.

The other complaint was that the restriction of the grid in terms of orientation was too severe. They approved the use of voxel as a modeling approach but noted that being able to change the orientation of the grid and size of it would be beneficial. The objective was not to work with a lot of different orientations or sizes but choose one that would suit better the topology of the site’s scale model. The problem of different orientations was solved and almost by chance another complaint, though a minor one, was also solved which was the urge to have different layers and different versions of the same project and the ability to switch between them.

From the two testing sessions conducted the conclusions were that the system is easy to use in terms of paradigm and interface to the point where it can compete with commercial software to a degree. Despite having almost no experience with this kind of concepts, users rapidly understood how to interact with the virtual reality and spatial interface. Regarding the possibility of this system being used in an architecture’s studio it showed a lot of potential with architects agreeing that they were willing to use it during the design process if the mentioned limitations were overcome.

C. Collaborative Tests

The collaborative tests were not as extensive due to the fact that it is not the main focus of this work. Despite this, the conclusions were positive and noteworthy. The users considered that the avatars were sufficient representation of the other users and that they were able to pass the WiiMote to one another, although with some degree of difficulty, and communicate efficiently. They found the interface intuitive and satisfactory, attributing most problems to the lack of experience and recognizing the potential for this new setup of Maquetteer. Overall the feedback was positive and indicated that there is potential enough to pursue this line of investigation and to continue to improve the application. The users also express this feeling and emphasized the remote collaboration as a valuable feature.

VII. CONCLUSIONS

The goal of this paper was to combine voxel modeling paradigms, popularized in games such as Minecraft, and a spatial interface embed on a VRE to create a CAAD tool for the early stages of architectural design. Due to this, the goal was for it to be easy to learn, to use and to be able to generate content quickly, all qualities associated with traditional sketching. Not only that, but by making use of VRE based spatial interfaces, the goal was to improve over commercial software which generally relies on WIMP interfaces which are not as intuitive and restrain spatial thinking.

The system proposed was a success in many aspects and the voxel paradigm proved to be well suited for this kind of tool making it easy to model with. Because the focus was on sketching and not on detailed models the constraints of voxels were not seen as a problem. The only complaint regarding this was not being able to work with two or three different orientations chosen by the user instead of the default one.

User tests with laypeople showed that the system is easy to learn and works well as a modeling tool and most users felt comfortable in the VRE despite not having little to no experience with this kind of technologies. When looking at the production of simple models of buildings with low details, which is the target of this work, the system can compete in terms of learning curve and modeling speed with commercial software while having all the advantages of a spatial interface. Laypeople’s tests were also valuable due to the feedback received on the system’s interface so it could be improved to a more polished version, which is fundamental for this work, since its aim is usability. This feedback also led to the conclusion that our interface had no fatal flaw.

Regarding real world application, professional architects found the concept interesting and were eager to explore more of its potential. According to them, the system could be used in architect’s studios provided that some alterations were made. Not only this but it could also have additional benefits in other areas such as learning and become a powerful tool in architecture courses.

The VRE and the spatial interface were not just gimmicks but a key point for the success of the system. The users who took more advantage of this fact by walking around the VRE and changing their point of view of the content they were modeling had the best results. Professional architects, when given the chance, would model around their body and try to make real size constructions instead of scaled ones. Being able to be on the same environment as the content that is being modeled not only made some tasks much more intuitive, such as positioning the cursor and changing the perspective on the modeled content, but it helped, as confirmed by professionals, to understand how the envisioned building or element would look like in real scale on the real world.

Despite the positive feedback there is a lot of room for improvement. The professionals do not consider that the system is yet ready for being used in an architect’s studio and on the other hand there are still some optimization issues when the number of voxels increases a lot, although this is only visible on extreme cases.

VIII. FUTURE WORK

As mentioned earlier, there is a lot of room for improvement. Two issues were brought up by professional architects and that were recognized as flaws of the system: being able to change the orientation of the grid and being able to switch back and forward between two or three different orientation and to be able to model at different scales and sizes. Some modifications already were made to the system, and were already described on this paper, to solve the first issue but they were not properly validated by user testing. In relation to the second issue, it would be interesting to understand better what are the benefits gained in modeling all around the body in real scale. Another aspect that requires more working and testing is the collaborative work and also, in a later stage, remote collaborative work.

Another direction that could be pursued is the integration of avatars, proprioception, in the system, and understand how that could improve the overall experience. It has been established that we use our own body to infer distances and dimensions of the environment around us. The collaborative experience could also be improved by this by making it
easier to demonstrate intent and to interpret the actions of others without relying only on verbal interaction.

Leaving the scope of spatial interfaces, there is much to improve in regard of the physical setup. It would be interesting to integrate the system in a more affordable setup that could be sold commercially. Another potential line of work would be to record the movement from the WiiMote and GearVR captured with the mocap cameras and analyze that data.

More important than improving the system however, would be to understand how it could benefit architecture and other similar fields. In the future there will be an attempt to integrate Maquetteer in the classroom on an architecture course and see if can improve the students learning process. But the main goal of this project is to be integrated as an important tool of the design process on architect’s studios replacing CAD software and/or traditional scale modeling techniques on early stages of the design process by improving it.

ACKNOWLEDGEMENTS

I acknowledge that I was financially supported by national funds through the Portuguese Foundation for Science and Technology with reference TECTON-3D PTDC/EEI-SII/3154/2012, IT-MEDEX PTDC/EEII/SII/6038/2014 and UID/CEC/50021/2013.

REFERENCES

[21] https://www.rhino3d.com/
[24] Daniel Mateus, Maurício Sousa, Rui de Klerk, Sandra Gama, Joaquim Jorge, José Pinto Duarte, “From Tectono to Tegono: going back to the classical roots of Architecture using virtual reality”, eCAADe, 2015
[26] https://www.youtube.com/watch?v=GSbkfnmCJXE
[33] Meshing - https://0fps.net/2012/06/30/meshing-in-a-minecraft-game/