KTH Datavetenskap
och kommunikation

Towards Embodied Perspective

Exploring first-person, stereoscopic, 4K, wall-sized rendering of embodied sculpting

FERRAN BERTOMEU CASTELLS

Supervisor: Mario Romero
Examiner: Olle Bälter

Stockholm 2014

Visualisation Interaction Collaboration
School of Computer Science and Communication
Kungliga Tekniska Högskolan

Towards Embodied Perspective: exploring first-person, stereoscopic, 4K, wall-sized rendering of embodied sculpting

Abstract

The central goal of this thesis is creating and testing technology to produce embodied interaction experiences. Embodied interaction is the sense that we inhabit a digital space with our minds operating on it as if it were our physical bodies, without conscious thought, but as natural as reaching out with your fingers and touching the object in front of you. Traditional interaction techniques such as keyboard and mouse get in the way of achieving embodiment. In this thesis, we have created an embodied perspective of virtual three-dimensional objects floating in front of a user. Users can see the object from a first-person perspective without a heads-up display and can change the perspective of the object by shifting their point of view. The technology and affordances to make this possible in a unobtrusive, practical and efficient way is the subject of this thesis.

Using a depth sensor, Microsoft's *Kinect*[7], we track the user's position in front of a screen in real-time, thus making it possible to change the perspectives seen by each of the user's eyes to fit their real point of view, in order to achieve a 3D embodied interaction outside the screen.

We combined the first-person perspective into an embodied sculpting project that includes a wireless haptic glove to allow the user to feel when touching the model and a small one-hand remote controller used to rotate the object around as the user desires when pressing its single button.

We have achieved what we call *Embodied Perspective*, which involves an outside-screen stereoscopic visualization, which reacts to body interaction as if the visualization was really where the user perceives it, thanks to the data from the depth sensor. This method does not block the user's view of their own body, but fits and matches their brain's perception.

When applied to virtual sculpting (*embodied sculpting*), it gives the user the ability to feel and understand much better their actions; where they are touching/sculpting and how they should move to reach where they want, since the movements are the same one would perform with their body in a real-world sculpting situation.

A further study of the viability of this method, not only on single person interaction but on group visualization of a single user perspective, is discussed and proposed.

Mot ett förkroppsligat perspektiv: att utforska en förstapersons, stereoskopiska, 4K, rendering av förkroppsligad skulptering i väggstorlek

Sammanfattning

Huvudsyftet med denna rapport är att utforma och testa teknik för att skapa embodied-interaktiva upplevelser. Embodied-interaktion ger användaren känslan av att psyket befinner sig i det digitala rummet och verkar naturligt som en fysisk kropp utan medvetande. Traditionell interaktionsteknik så som mus och tangentbord blir ett mellansteg som förhindrar inlärningsupplevelsen.

I rapporten har vi utformat ett embodied-perspektiv av tredimensionella virtuella objekt som svävar framför användaren. Från ett förstapersonsperspektiv kan användaren, utan en heads-up display, ändra perspektivet på objektet genom att förflytta sig i rummet. Rapporten behandlar tekniken som gör detta möjligt.

Genom att använda djupsensorn i Microsofts *Kinect*[7] kan vi spåra användarens position framför skärmen i realtid. Detta gör det möjligt att ändra perspektivet från användarens ögon för att matcha verkligheten och därmed möjliggöra en tredimensionell embodied-interaktion utanför skärmen.

Vi kombinerade användarens first-person-perspektiv med ett embodied-skulpteringsprojekt. Projektet inkluderar en trådlös haptisk handske, som låter användaren känna beröring av objektet, samt en liten trådlös handkontroll som med vilken objekt kan roteras genom en knapptryckning.

Vi har uppnått det vi kallar *embodied-perspektiv*, vilket innebär en stereoskopisk visualisering utanför skärmen. Detta reagerar på kroppslig interaktion som om visualiseringen är så som användaren uppfattar den, tack vare informationen från djupsensorn. Denna metod blockerar inte användarens vy av sin egen kropp utan matchar hjärnans uppfattning.

Då *embodied-perspektiv* appliceras på virtuell skulptering (*embodied-skulptering*) ger det användaren förmågan att känna och förstå sina handlingar bättre, så som att röra/skulptera och förflytta sig med precision. Detta eftersom rörelserna är de samma som i en verklig skulpteringssituation.

Vi diskuterar och föreslår vidare studier för hur denna metod kan genomföras, inte bara för användaren, utan för gruppvisualisering av användarens perspektiv.

Acknowledgements

Foremost, I would like to thank the professors and researchers in the VIC studio at KTH, Henrik Edlund and Björn Thuresson, for their continuous help, support and feedback during the whole duration of my stay working in the VIC studio, and to my supervisor Mario Romero, whom with his lectures where I have attended during this year in KTH, has been able to enlighten my ideas and changed my way of thinking and seeing this world. I really thank him for all of the overtime hours he has spend with my projects.

I would like to express my sincere gratitude to several people who have had a big positive impact on me during this year: Leonard Graf, for working besides me so many times, Gerardo Daniel Aguirre, Óscar Álvarez Álvarez and Celia Carvajal Almendros for their constant cheers and friendship, and Guillaume Dolange for our guitar sessions to break the routine.

I thank my fellow orchestra mates from PromenadorQuestern for helping me with the translation: Tove Odland, Jonas Sjöberg and Camilla Johansson.

I would like to express my sincere gratitude to Carlos Ramos, professor and philosopher, who kindled the spark of science and technology on me and many others when I was back on high school.

Last but not the least, I would like to thank my family: my parents Josep Carles Bertomeu Sabaté and Rosa Maria Castells Turón for their support throughout all my student life and for believing in me, and my sister Carla Bertomeu Castells.

And my most beloved gratitude to YenTing Lo.

Contents

1 Introduction **1**

2 Related Research **3**
2.1 Embodied and First-person Perspective 3
2.2 Virtual Sculpting Visual Feedback 4
2.3 Virtual Sculpting Haptic Feedback 4

3 Methods **7**

4 Development **8**
4.1 4K stereoscopic-3D rendering 8
4.2 Head-tracking 10
4.3 Interaction and interface 14

5 Results **20**

6 Discussion **23**

7 Conclusions and future work **26**

8 References **28**

List of Figures

1 Display setup 10
2 Screen rendering size comparison 11
3 Kinect FOV and inclination 12
4 Eye position estimation 14
5 Collision detection calculation 15
6 Interaction tools 17
7 Shadow projection 18
8 Body mesh rendering 19
9 Table of affordances 20
10 Embodied sculpting setup 22

1 Introduction

Recently, Virtual Reality devices have become popular and affordable[8]. Therefore, researchers and developers are increasingly engaged on the design of virtual interaction-interfaces. The main concern about VR is the fact that when achieved by today's methods, the user is deprived of the view of their environment. All of the VR techniques involve a heads-up display that blocks completely the user's view of the real world. Alternatively, we can find setups such as the *CAVE*[3], which involves placing the user in a small cubic room which blocks the user's view of the real world anyway.

At the same time, Augmented Reality has appeared as a non vision-blocking alternative to VR that achieves the placement of virtually generated content on the real world from the user perspective, without completely disconnecting the user their surroundings, such as Google Glass[5]. AR aims to not break the user's embodiment, their body feeling and cognition, by letting the user still see their body through transparent heads-up screens, wearable projectors or camera feeds.

With AR though, since the virtual world is *projected* on top of the real world, there is a visual context-blending that the user has to keep performing in their minds in order to not break the reality consistent feeling of the virtual objects, in order to not break the belief of the whole AR experience. Moreover, except for a handful of AR projects (such as Harrison's *Skinput*[17]) there are few examples of AR projects that react or make use to the user's body. Therefore, the interaction with the virtual elements of the AR experience may feel artificial or out-of-body.

We want to be able to keep the embodiment on a virtual interface and visualization, so that it reacts to the user's body interaction without blocking or obstructing their view, while still keeping the consistency of 3-dimensional stereoscopic visualization of a virtual environment. The central research question of this thesis is:

> *What are the affordances needed to design and develop an unobtrusive, practical and effective Embodied Perspective?*

By *unobtrusive* we mean a visualization that does not interfere the user's view of the real world and, to a lesser degree, the user does not have to wear the display. By *practical* we mean we achieve the embodied perspective experience without expensive, single-purpose hardware. By *effective* we mean that we achieve a level of embodied perspective that is close to indistinguishable from reality. Therefore, we asked ourselves how we could achieve such perspective without the use of complex, expensive state-of-the-art devices or gadgets, and without blocking the user's view of their own body, allowing them to interact naturally as they would on a real-world interaction.

We decided to explore this topic in the field of Virtual Sculpting, in the hands of *Embodied Sculpting*[21].

Embodied Sculpting is an interactive experience, which started as *Virtual Sculpting*[12], a real-time volume modification and rendering with in-air full-body interaction and haptic feedback. *Embodied Sculpting* allows the user to

sculpt a virtual object with their own body, without the need of any intermediate interfaces.

For many years, there have been projects such as *Sculpting*[19] which made use of a 3D tool that the user had to move trough a delimited area, which aimed to work in a similar fashion a sculptor would use her hand or tool to carve a soft material such as clay. The user has to grab an object and move it in the defined space, as a 3D brush. The virtual sculptor has always relied on a tool that acted as interface between the real world and the virtual world, which requires skill that has to be learned. On *Embodied Sculpting*, making use of a 3D depth sensor, a haptic glove and a simple one-hand controller, we aimed to remove the learning barrier and adaptation time in order to feel, carve and interact with the virtual object. Technically, the user sculpts by intersecting the depth mesh of her body with the volume of the sculpted model. While we could have created special gestures for controlling the model, for example rotating and translating it, for simplicity and reliability, we opted for a one-handed one-button controller for context switching between sculpting and controlling. Thus, by pushing the button on the controller, the user can rotate the model rather than carve it.

2 Related Research

In this chapter, we cover three main areas of related research. First, we discuss work related to first-person perspectives and embodied perspectives. Most of this work employs some kind of gadgetry that the user must wear. We will briefly cover this topic, but we will focus on unobtrusive first-person perspectives of which there are very few examples. Second we focus our discussion on visualization and displays for embodied sculpting-type interfaces. Third, we cover haptic feedback as it relates to similar sculpting experiences.

In the search of a more faithful experience when sculpting a virtual object, research has been focused on a main topic: haptic feedback and interfaces, since the concept of sculpting inherently implies the presence of a physical material, which involves physical interaction with the material either through a tool or direct body interaction. When it comes to virtual sculpting the need of a physical response feedback is usually taken as the most important factor when it comes to trying to recreate the real world experience of sculpting.

2.1 Embodied and First-person Perspective

First-person stereoscopic perspectives have been researched mostly on Virtual Reality setups. In order to achieve embodiment, these projects make use of rotation and position tracking of the user's head in order to change the view of the virtual world on the user's heads-up displays, which block the user's view of the real world. The lack of vision of the user's body has made researchers use real-time imaging provided by cameras to render back to the user their own view of their body.

A recent example of the use of cameras to put the user's body into the virtual world has been the an approach that makes use of three Kinect cameras in order to materialize real world inside the virtual world of the VR experience with its volume[1]. Making use of both the depth data of the kinect's depth sensor and the color supply from the kinect camera, the authors have been able to provide a 3-dimensional mesh of the user's body and surroundings by calibrating a series of kinect sensors that observe the same region on space, where the user is placed. By coordinating all the depth data from the sensors they are able to provide a really immersive embodied experience while maintaining the visualization to keep completely virtual in the virtual world.

Other approaches to first-person perspectives without the use of heads-up displays such as the *CAVE*[3], utilize a small room where projectors are directed to all of its walls. Making use of 3D glasses and 3D projectors, a full stereoscopic environment can be perceived by the user inside the room. The user's head position is tracked and the projections on the walls are changed accordingly thus allowing the user to freely move inside the room and feel their perspective change, creating the illusion of a virtual world. This setup does not block the user's view of their own body, but rather places the user in a physical space which becomes a virtual environment.

When it comes to screen 3D visualizations there have been projects that have made use of Microsofts's kinect depth sensor before[22] to achieve head-tracking and perspective change on the visualization. The focus of the stereoscopic view of these projects though, has been the visualization of virtual worlds inside the screen. In our approach we want to bring the virtual world outside the screen,

at hand-reach, in order to perform the interaction where the user can move with their body.

2.2 Virtual Sculpting Visual Feedback

Most of the research on virtual sculpting has only explored simple on-screen, flat or stereoscopic, visualizations. This research has focused mostly on haptic feedback, which we treat in the next section.

An enhanced approach to the visualization and interaction on the field of virtual sculpting was developed in *3DIVS*[18], which stands for *3-Dimensional Immersive Virtual Sculpting*. The authors describe an environment designed for 3D virtual sculpting that tries to focus on the ease of use and the increase of productivity for designers and Computer-Aided Design (CAD) engineers. Combining a stereo projection system, 3D computer-generated images, shutter glasses with integrated head tracking, pinch gloves and a stylus device, they created an efficient and precise virtual sculpting and design workbench. The projection system allowed a 2x1.5m projection on a translucent table screen[24]. Using the spatially tracked data gloves and pointers, they developed a two-handed user interface. In their approach, the user is always wearing two gloves and can make use of a stylus in situation-specific contexts. On the set of available controls that they defined, the common factor was the point-position tracking. Using the gloves, they detected when the user was pinching and where in the workbench-relative space the pinch was being performed. Then, through the contextualization of the hands' movements and pinches together with the stylus rotation and position, when applicable, they were able to define and perform the different actions of the modeling system. Although the visualization of the 3DIVS is performed in stereoscopic 3D with head-tracking, it is only available and fully immersive for the user performing the 3D sketch. The context-relative controls to perform the actions in the environment are abstract and fully associated to the mathematical definition of space and shape deformations. Therefore, this interface is meant mostly for expert users.

In our approach, we are looking for a much more intuitive and natural visualization and interface, that can be understood easily by novices. Moreover, we want to make the experience more apt for a spectating audience. Our goal is to allow the audience to enjoy, observe, understand and learn what the sculptor is doing, transforming the event into a group experience.

2.3 Virtual Sculpting Haptic Feedback

A haptic feedback for Embodied Sculpting was designed and evaluated[21], previous to this research, and stated that the haptic feedback was by far the most rewarding an important improvement on the experience when users got to interact with and without it. In our research, sculpting students from Konstfack[6] who tried an early prototype of the embodied sculpting project, remarked that the haptic feedback truly enhanced the experience of sculpting. They animatedly suggested that more resources be invested in the improvement of the tactile feedback. They claim that the tactile feeling of the material teaches the sculptor the most on how to sculpt using whichever tool and remarked that on virtual sculpting is even more important to know and learn.

On virtual sculpting, haptic feedback involves two different but equally important feelings: the perception of the shape of the object being sculpted and the feedback that the user receives when sculpting the virtual object.

This has important implications on how haptic feedback is designed for these kind of experiments. In general, virtually sculpting requires the system to provide the most precise surface tactile texture over the virtual object while, at the same time, the system must be able to update the data model of the object over 1000 times per second.

Blanch R. and Ferley E.[16] studied how just using simple viscosity feedback and surface contact forces one could recreate a good non-realistic haptic feedback for virtual sculpting. They created a voxel-based model which the user could modify by adding or removing material through a *Phantom desktop* device. They defined the force feedback so that the user could feel the outside surface as hard on contact. Whenever the user sculpted with the tool, they would change the force feedback to be relative to the viscosity of the position where the user was sculpting. That means that there was no surface feedback when the user was sculpting, which simplified force calculations. They concluded that their method greatly improved the usability of the virtual sculpting and details had been easier to model.

On the other hand Jagnow R. and Dorsey J. at MIT[20] used a technique to avoid the use of a voxel-based model that involves simplifications of triangle-based models and haptic-displacement maps. Their method consists on generating, from a complex high-polygon model, a simplified low vertex-count model that plummets the amount of triangles and creates a haptic-displacement map for each simplified triangle, which represents a function that maps the height difference between the actual simplified triangle surface-height and the real position of the surface. The technique resembles lightning methods that are used in computer graphics such as *parallax mapping*, but instead is used for smooth haptic surface feedback which allows a really high haptic-update frequency while still being able to compute the force feedback from a complex surface. Haptic displacement maps offer a rapid interaction method for editing complex models, though the authors acknowledge that this method is just able to perform small local changes over the original model.

In *Embodied Sculpting* we have a voxel-based model instead of a polygonal mesh. When using triangle-meshed objects, a practical approach is haptics displacement maps, but as the authors point out, it is not suitable for great deformations since the simplified model of the object should be dynamically updated with its displacement maps, which, therefore, is not viable for real-time feedback.

To maintain large deformations in real-time, it is practical to define a voxel-based model for the entire space of the material and its surroundings, making it relatively simple to modify deeply and freely the form of the sculpted object without prohibitively high computational costs. To provide the highest haptic resolution possible, the system must increase the voxel count as much as possible. In our case, we are increasing the scale of the model's physical space to facilitate whole-body interaction. Therefore, we can theoretically increase the voxel count significantly. Nevertheless, the system is limited by how acutely it perceives the user's body movement and shape. This perception acuity is defined by the resolution of the system's sensors, in our case the depth sensor's resolution and frame rate. Specifically in the case of Microsoft Kinect for Win-

dows, the resolution is roughly $(1.5cm)^3$ at our average operating distance of 1.5 meters.

3 Methods

The methodology of this thesis' project has been iterative design with several cycles of prototyping and formative evaluations. Through this approach, we have detected and improved upon the project's original shortcomings to reach the most satisfactory embodiment for our users. Our target population is composed primarily of sighted on both eyes, hearing and dexterous individuals without experience in 3D modeling and interaction. We designed the embodied sculpting and embodied perspective experience to be immediately immersive without training or special hardware to be worn or operated by the user. Within this target population, we focused our final evaluation on art students with knowledge of physical sculpting to compare their experience with that of embodied sculpting.

Talk about the plan of user study, see conclusions "We planned to.." Originally, we planned to run a user study to compare and evaluate the effect of embodied perspective on the embodiment feeling of Virtual Sculpting, by making users try the project before and after the development of this thesis, with and without embodiment perspective. In the end, due to time reasons and change of plans we did not perform such study.

4 Development

The technical setup of the project consists of a 3D depth sensor, in our case a *Microsoft's Kinect*[7] and a stereoscopic screen. The screen and the sensor face the user and are calibrated to spatially and temporally synchronize the depth data of body movements with the real-world position of the virtual model. In other words, where people see their real hands to be, the screen displays their virtual hands. It coordinates the stereoscopic display of virtual and virtualized objects with the user's perception of physical reality. Virtualized objects are those that result from the scanning of the depth sensor and rendering on the screen.

The system calculates the virtual collision between depth points received from the sensor and the voxels of the volume object. This setup allows any perceivable object placed in front of the sensor to be used as a carving tool because the context is ignored and just the depth values from the sensor are used. The sensor perceives opaque objects. It does not perceive highly reflective or transparent objects.

The previous state of the rendering of the *Embodied Sculpting* project, before we started working on it, was performed on a flat screen, with no real depth perception, which made it difficult for the user to visualize their position on the virtual world, to know and understand where should one move or do with their hands and body in order to reach and interact with the model.

In order to achieve Embodied Perspective, we made use the 4K-resolution 4-meter screen that is available in the Visualization Studio VIC Sthlm from KTH Royal Institute of Technology [13].

The biggest challenge of the project has been to take the existing project and code base and carry it over and upgrade it in order to fit our goals and any upcoming research on the topic. Previous to my overtake as a programmer, developer and designer on the project, there have been at least two other people involved on its development at the VIC studio. Once I got the chance to get involved though, I had to face the understanding and comprehension of the code alone.

The code of the project can be found in Github[4] under the name of *Virtual-Sculpting* and it is open to anyone to see and download.

4.1 4K stereoscopic-3D rendering

My first aim once I got my hands on the project source-code was to port the rendering to the big screen: to the 4K stereoscopic screen that is available at the VIC studio. As mentioned before, the project was initially rendered on a simple screen at FullHD (1920x1080 pixels), so the first step was to allow 4K (4096x2400 pixels) rendering.

The fact that the project was a Microsoft Visual Studio project for Windows machines made the previous programmers go with the use of the native Windows API for windowing and input management. This API is old, obscure and completely device dependant, which made it impossible to render past the FullHD resolution on the machine we had available. For this reason, after some research over the topic, I decided to go with the *Simple DirectMedia Layer (SDL)*[10] 2.0 library, a powerful API for low level access to audio, keyboard, mouse, joysticks, and graphics hardware via OpenGL and Direct3D, distributed

under the zlib[14] license and fully open-source. This library is used and maintained by many high-end real-time graphics companies in order to abstract the complexities of developing on different platforms while still allowing a low level access to the programmer.

Once I ported the whole project to SDL, I could manage to get the rendering at 4K working right away and then started working on the stereoscopic-3D rendering. On the 4K screen of the VIC studio, there are currently two ways to visualize content in 3D. Since I had been working on the VIC studio since September 2013 I have been able to work with both approaches. Behind the 4K screen there are three computers: two identical machines, each connected to a 4K projector and able to render up to that resolution, and one more powerful machine, which has two identical graphic cards each connected to on of the 4K projectors and thus is able to render 2x4K resolution by itself.

The 3D stereoscopic effect is achieved thanks to the *Panavision 3D*[9] technology, a passive 3D viewer technology, the rendering of each eye is needed in each projector and then a color filter is applied which allows the perception of 3D on the viewers wearing the proper glasses. I had previously worked with this setup using the two identical machines, making each machine render each eye and synchronizing the machines through network communication. The approach worked quite well but the network communication was not always consistent so this time, on this project, I decided to take the path of the single machine, multiple graphic cards. Each 4K resolution projector uses 2 displays through a DVI connection. See figure 1.

When rendering on a window through a graphic card, the concept of graphical context or rendering context appears and needs to be understood. The rendering context defines the format of the pixels of the window and how are they going to be communicated to the graphic card and stored in memory. These pixel and context formats and its initialization are platform specific, thankfully SDL gives a nice abstraction which allows us to define and create the context in a simple way. In our case, since we are working on a 2-graphic card machine we need to communicate information between the graphic cards because both are going to be working with the same data. Each graphic card has its own internal memory, where we push the vertices, shader programs and textures to be rasterized and rendered on the screen. Since I needed to render the scene twice, once for each eye, on different graphic cards, I needed to share the OpenGL context between the windows and the graphic cards. SDL again simplified the work and we could get the multiple-window rendering working with not much problem.

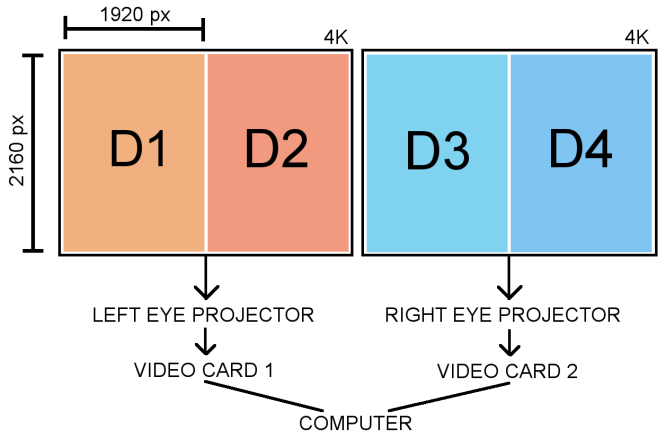

Figure 1: Each display is half 4K resolution, each graphic card holds 2 displays and each eye gets 4K resolution

4.2 Head-tracking

In order to create a fully immersive experience the simple use of stereoscopic 3D is not enough, thus we decided from the start of the project to implement some kind of head-tracking in order to improve the visualization of the sculpting. If we are able to track the position and inclination of the user's head relative to the screen, we can then determine where the eyes of the user are, and thus we are able to perform the proper changes on the image shown to each of the eyes.

In our case one of the most important factors of the 3D visualization is the fact that we wanted to bring the interaction with the virtual object outside the screen, into the real world. This has implications that have to be considered. The fact that we bring the object closer to the user and outside the screen means that the size of the object and the distance that we can place that object are really determined by the size of the screen where this 3D rendering will be viewed. See figure 2.

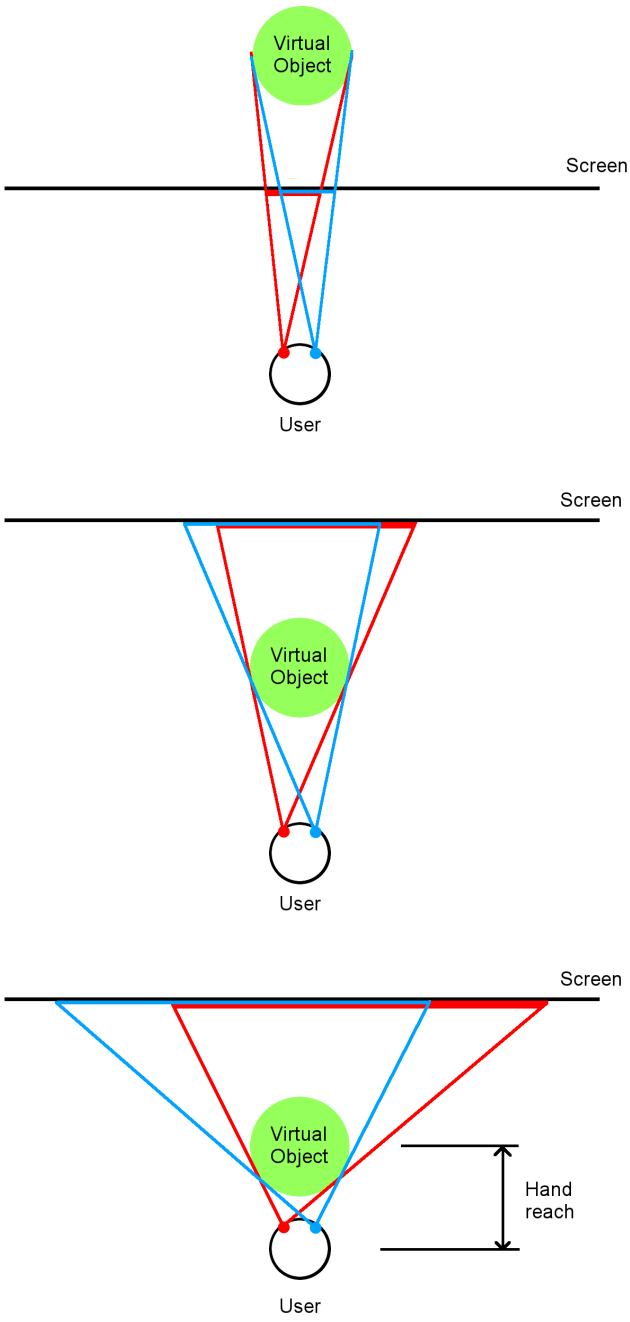

Figure 2: Comparison of the size on screen of the rendered object depending on the position of the virtual object. From top to down: inside the screen, outside and at hand reach.

The closer we get to the virtual object in the real world, the bigger the rendering of the object has to be in screen. This caused the object to fall outside the screen on certain angles due to it's size and position. During the track of the development, the position of the object and it's size has been changed several times, leading to an object occupying a volume of 0.125 m^3 (0.5m sided cube) and at a distance of 1 meter outside the screen.

Since we wanted to keep this project as simple as possible we decided to make use of the official Kinect SDK ability to perform skeleton tracking using Microsoft's own tracking algorithm. We decided not to use any other position tracking method, such as infrared motion-capture camera systems, that delivers a faster and more precise position tracking but it would have been a big money and time investment to setup on the VIC studio.

Initially the project worked along with a Kinect sensor placed right in front of the user, right below the screen. In the new environment with the big 4K screen though, the fact of having a Kinect on a stand, in front of the user, had an huge negative effect on the immersion, and broke the feeling that the object was in the real world. Moreover, after the first tests with Kinect's skeleton tracking we realized that the head-position tracking was constantly being lost due to the user using their arms to sculpt the virtual object; when the user's hand or arm was blocking the direct view from the Kinect placed around 1 meter above the ground, the skeleton tracking could not keep track of the user's head-position, thus completely breaking the experience and immersion.

Because all of this problems related to the position of the depth sensor, we decided to build a stand just on over the screen and face the kinect down in order to reduce the probability of accidentally blocking the Kinect's vision of the user's head while interacting with the virtual object being sculpted. See figure 3.

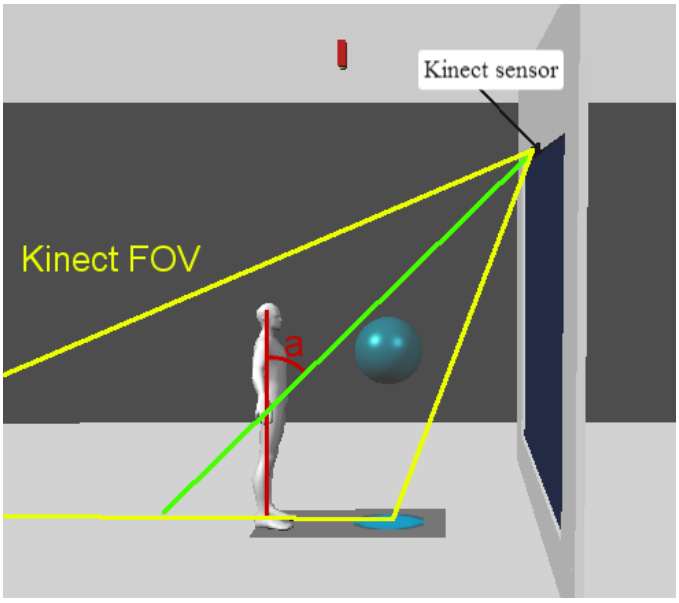

Figure 3: The position, field of view (FOV) and angle (a) of the kinect relative to the user affects the standard skeleton tracking negatively.

This decision entailed unfortunately certain consequences regarding the fidelity of the position tracking. The fact that the user was not totally facing the Kinect, their body was not perpendicular to the sensor direction, made the skeleton-tracking algorithm fail more often on certain circumstances. The algorithm, for certain users, due to their shape and sizes, failed to detect the skeleton positions every now and then. Tall people, small people and people wearing voluminous clothes were a challenge for the algorithm.

We tried to implement some noise-removal to the head position we were getting from the Kinect SDK's skeleton-tracking, but the results were not satisfactory due to the addition of smoothness, latency and lost responsiveness. While it's true that we don't want to dizzy or annoy the user due to head-position jumps caused by the loss of the head-tracking, we want to avoid latency over anything, avoid the feeling that the view that one sees, defined by the head position, does not correspond to the current position of the actual head of the user, which was by far the thing that annoyed the most to the testers who tried an early version of the project. After some tweaking with the Kinect SDK skeleton-tracking's configuration, we achieved a more than acceptable result and feeling.

The head rotation comes to take an important role on the calculation of the projection matrices from each eye as well, Kinect SDK's skeleton tracking just gives us the position of the center of the head but we need to change the view perspectives from each eye position to generate a immersive visualization. Unfortunately the characteristics of the project setup do not allow an easy accurate way to track such head rotation from the Kinect depth data. The Kinect SDK has indeed a build-in face expression recognition, that allows us the get reliable eye positions. In our case however, the Kinect sensor is too far away from the user's face, and the fact that one must wear 3D glasses in order to visualize the project makes the face recognition and tracking fail all the time and this makes it not viable.

Instead I took a much simpler, yet effective, approach to the problem. Since the sculpting object's position is always fixed and we know that the object takes all of the user's attention we can induce that the user will be looking towards the object most of the time. I then decided to, determining a certain eye distance and head depth, induce the eyes' position by adding forward to the tracked head-position in the direction of the object a certain distance to get an estimation the eyes' position and calculate the correspondent stereoscopic perspectives. See figure 4.

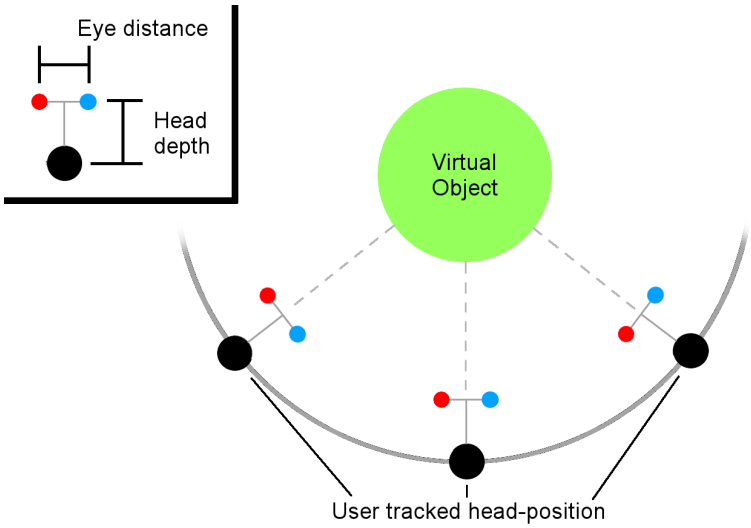

Figure 4: The eye positions, in red and blue, are induced from the head-position, in black, and the virtual object position.

Using this technique we create a very close to reality approximation, this way, we manage to fit the user's perspective, which is crucial for the visualization and interface to not break the feeling of immersion, without the need of any other device.

4.3 Interaction and interface

The initial interaction of the project before we started exploring the embodied perspective took place thanks just to the kinect sensor depth buffer stream. The kinect sensor is a source of colour images and depth images. Using the depth images, which consist on a distance or depth value from the sensor to the closest object for each pixel of the image, we have a knowledge of the shape of the objects in front of the sensor. With these depth values we can generate a polygonal mesh that fits the shape of the world in front of the kinect, and then visualize it and perform collision calculations on top of it.

Since the earlier version of Embodied Sculpting was performed with a kinect standing right in front of the user, the calculations and visualization was made straightforward from the received values and without any knowledge of the context setup or environment. The program directly transformed into virtual-world positions the depth values of the buffer. On our approach, we had to transform all the depth positions into real world coordinates, taking in account the position and orientation of the kinect.

These depth positions were then processed towards the virtual object volume in order to calculate the intersections and collisions between the depth positions and the voxels of the sculpted object. The way this collision detection was performed had to be completely reworked because, when having the kinect so far away and in such inclination as in our project, the the real world distances between pairs of neighbour transformed depth-values from the sensor fall many times really far apart.

This caused the sculpting on the virtual object to not be coherent with the user actions; when the user would use an arm to perform a removal, the object would just be superficially affected as if the user body was not fully solid, the interaction felt inconsistent.

In order to correct this issue I developed a better collision detection approach. Since this collision detection is being done in parallel in several threads, we needed an approach that does not involve concurrency dependencies.

For each each pixel of the depth image, we compute its real-world position and the positions of the pixels right above and to the left of it. With this we know the distances and directions to reach its neighbour's position. Since we know exactly how big the voxels of the sculpting object are, we can discretely check the collision of the surface that is formed through the both directional vectors, up and left, that come from the depth position and the voxel data. We just need to use a slightly smaller discrete step distance than the voxel size. On top this, we need to consider the fact that the depth points have no volume, but the user's tools and body have volume behind the surface that the kinect detects. We cannot know how deep our users are, for this reason we defined a certain depth to check for collisions in each checking point always forward, towards the screen, in space.

In other words, for each transformed depth-point we check a discrete 3-dimensional amount of the space defined by the directional vectors towards the neighbour points and the depth-tool vector forward, in steps that are smaller than the size of a single voxel. See figure 5.

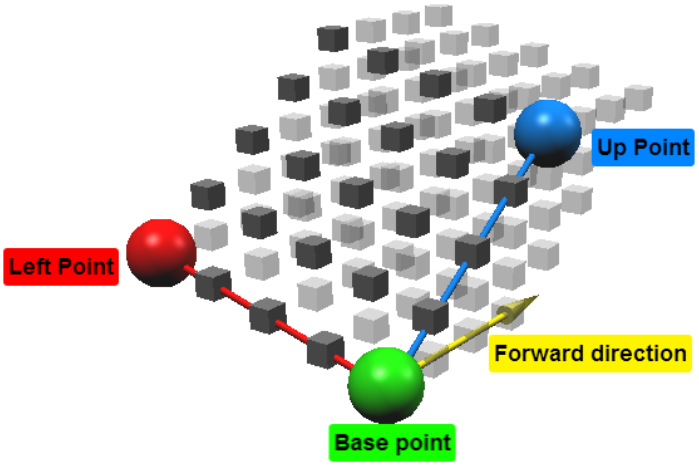

Figure 5: The surface positions, in black, and the depth positions derived from those, in grey. All the cubes and the base point are checked for collision[2]

When detecting the collision with the object this way, we can know how many voxels we carve, how much we are affecting the object. With this information a carving sound is played whenever the user touches voxels and a haptic feedback is send back through a wireless haptic glove that the user wears on one hand. The glove vibrates when touching the object, this way the user feels when they

are touching, which has seen to bee a much important feedback than sound or visual feedback[21].

The previous control interface of the project consisted on an on-screen text menu with options that could be accessed both through keyboard input or voice commands. We kept and evolved the keyboard commands but we disabled the visual text menu and the voice commands because they did not fit the interaction. The fact of not usually being alone when interacting with the experience made it impossible to have a good ambient sound conditions for the voice commands to fully work. Moreover the users are usually talking about what are they experiencing and we didn't want to restrict that. Instead we developed a simplified active interface and defined a set of keyboard actions for the person in charge running the project.

The user was able to interact with the object through a simple hand-controller which was just used for it's only button. We wanted to implement the feeling of grabbing the object to make the user able to control it, to rotate it and spin it around. Initially we intended to detect the user's hand to see if they were *grabbing* or releasing the air with their palm. Myself I had worked with this kind of detection with kinect before and I found it not properly stable for user interface usage.

The first problem with grab detection is the fact that one has to be really close to the depth sensor or camera in order to detect properly. Moreover, having small or big hands makes it impossible to interact this way. The second problem I have observed is that users usually do not keep the track of their hand when using this interfaces. Users would usually grab the air to perform an action but never release or *open* the hand when they are finished with the action, resulting on unwanted, unexpected effects. If one regards to physical interfaces, when grabbing controllers or interfaces the feeling of touch and presence exists, whereas grabbing the air is not consistent and can be confusing and hard to use, because the user has to invest energy and attention on thinking about the virtual grabbing that they are or not performing.

Figure 6: A one-button controller, on the left, and a haptic glove, on the right

For this reason we introduced a dummy physical controller which the user can hold onto and press when they want to perform a rotation action. See figure 6. With the button on this controller we detect whether the user is grabbing the object or not. Since we are tracking the full skeleton of the user, we can know where in space the hand with the controller is (in this case the left hand), transform the coordinates relative to the sculpting object and so be able to rotate the object accordingly. When the user releases the button, if their hand was still on movement, the object is going to keep spinning on the direction the hand was moving in order to accomplish the effect of a potter's wheel, but in our case, we are able to rotate the object in any axis the user wants since we are not delimited to a physical wheel.

After the first tests with inexperienced users, we realized that the user did not understand their position relative to the object at first glance. Many people still would not understand the fact that the virtual object they are seeing is actually located as if it was a real world object. The fact that the object appearance is quite artificial and that the predisposition towards a computer is not supposed to match reality, made people stand back and be afraid of the project interaction at first. Even after moving around in front of the screen and seeing how the perspective changed, some users, unfamiliar with 3D visualizations, had to step

onwards against the object several times in order to be in reach for sculpting. Some users had trouble understanding the virtual world.

In order to ease this initial learning encounter for inexperienced users, we decided to try to give the user another visual cue on where the object in the real world is. For this purpose we made use of a small projector that we placed right on top of where the object should be in the real world, facing downwards, in front of the screen between the sculptor and the screen. On this projector, we rendered a vertical orthographic projection of the virtual-world space into the place where the projection is in real world, in order to achieve a shadow-like effect. On this projection on the floor the user can see the *shadow* of the object inside the real world, outside the screen. This way inexperienced users would understand better the placement of the object by looking at the floor. See figure 7.

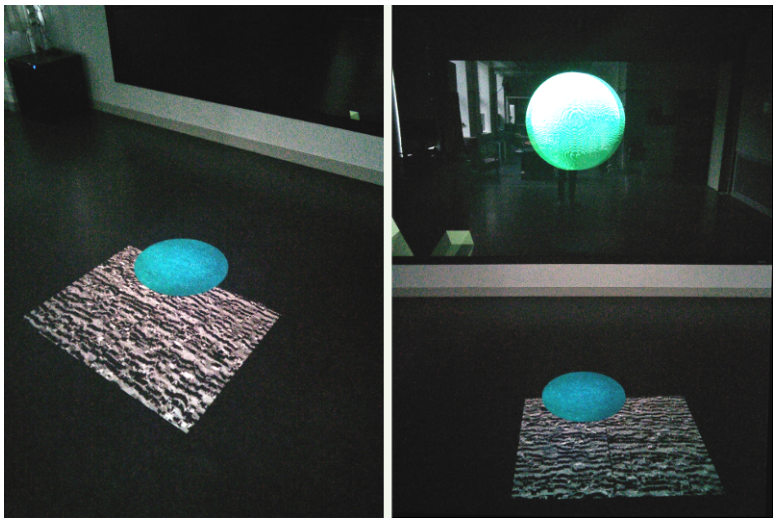

Figure 7: The *shadow* projected on the floor

The last issue we encountered was the difficulty for the user to relate their physical body to interact with a virtual object without any intermediary. The sculpting on the virtual object is performed exactly where the user places their body. Some users though, experienced difficulties to effortlessly understand their actions on the object, probably due to latency, because, even though we managed to keep it really low, there is latency between the user real movement and the depth information that the computer receives from the sensor, around $\tilde{0}.25$ seconds. However, the whole delay when sculpting with the body makes the removal of material on the virtual object to happen on a different place where the user's body is. If the user is to move an arm sideways the removal would be done with the depth sensor latency at a position behind the actual arm position, where the arm was a $\tilde{0}.25$ seconds before, confusing certain users.

For this purpose we decided to include the rendering of the depth-generated mesh of the kinect sensor into the virtual world where the user sees. This way when the users puts their body between the screen and their eyes, a white light semitransparent mesh is rendered where the depth data is perceived and thus the user can relate their movements and body to something visual the can see

in the virtual world. This mesh has a the latency of the depth-sensor so the user can see where the sculpting is being performed. Users felt it much more pleasant and understandable to visualize their virtual selves this way and the learned how to move on this interactive experience much faster. See figure 8.

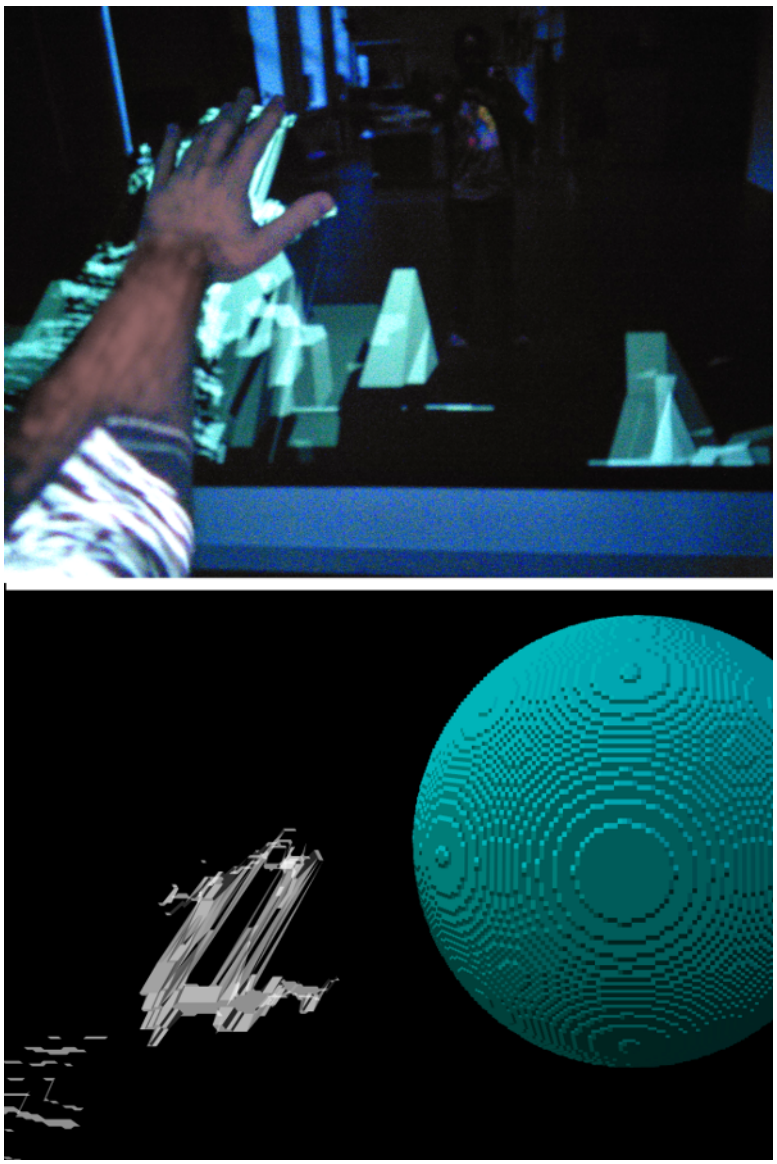

Figure 8: The user's point of view, on top, and the image on screen, on the bottom

5 Results

We recall our research question:

> *What are the affordances needed to design and develop an unobtrusive, practical and effective Embodied Perspective?*

Answering this research question has brought us to develop certain practices and techniques that we have been able to test through formative evaluations. Here we analyze the affordances we created as they relate to our research question.

	Unobtrusive	Practical	Effective
Depth-mesh interaction	✓	✓	✓
Haptic glove	✗	✗	✓
3D on screen	✓	∼	✓
Head-tracking	✓	✓	✓
One-hand controller	∼	∼	✓
Shadow	∼	∼	∼
Self-visualization	∼	✓	∼

Figure 9: Adapted affordances from previous state of the project, on top, and developed affordances, on bottom. Points marked as ∼ will be further discussed on the next section.

Both the depth-mesh interaction and the haptic feedback were present on this thesis' project before the development of our Embodied Perspective. We have adapted them into the new approach because they enabled embodiment and proved to be effective. The previous researchers in this project[21] determined that the haptic glove was a strong enabler of embodiment .through a controlled user study comparing conditions with and without haptics.

With the depth data from the sensor, we have taken a collision detection approach, completely unobtrusive, to interact with the virtual world between the user and the screen. We have moved the virtual environment outside the screen, at user's hand-reach, but we never obstruct their view of their body. This way, the screen becomes a window to the virtual world that allows users to interact with it without forcing them to be placed into the virtual world. It is a practical affordance because we make use of the same hardware that we do for the head-tracking and it becomes effective to locate the user's body in physical space.

Since virtual objects lack physicality, adopting a haptic glove enables users to perceive by touch, not just by sight, the physical boundaries of the virtual objects, given them a true sense of immersion and embodiment. Nevertheless, this approach is not practical nor unobtrusive. The fact that the user has to wear the glove in order to feel the haptic feedback is a drawback on the unobtrusive embodied experience. The trade-off to the partially obtrusive glove is its effectiveness in embodying the user in the experience. Until we develop non-invasive and non-obtrusive mechanisms to stimulate nerve endings directly,

a physical device such as the haptic glove will be a necessary trade-off. Currently there are a few promising research projects exploring such devices. For instance *AirWave*[15] makes use of an air-vortex generator to shoot haptic feedback through the user's body. On the other hand there are other researchers[23] that make use of a phenomenon caused by ultrasound waves to send pressure variations through air at sound-speed.

Stereoscopic 3-dimensional rendering on-screen has allowed us to create an unobtrusive virtual environment because all the visual stimulus that users receive arrives directly to their field-of-view with first-person perspective. This perception is achieved without heads-up display or any type of marker on the head or body of the user. The only device that users wear are standard 3D glasses, the same type of glasses worn on popular 3D movie theaters, which, for the purpose of this thesis, we do not define as strictly obtrusive. Note that we can still achieve first-person perspective even without the 3D glasses. What we lose, of course, is the sense of depth through stereoscopic perception. As it is, we rely on the effectiveness of the naturally evolved human visual system to produce stereoscopic perception.

With the use of a depth sensor, Microsoft's Kinect in our case, we have been able to both track the user's head position and the shape of any solid and opaque object in front of the screen. Thus, we have been able to approximate the user's eyes' position and produce an effective, practical and unobtrusive first-person perspective to the user. The Microsoft's Kinect is accessible to the consumer market because of its price and popularity.

The one-handed controller with one active button that we attached to the system replaces the lack of physical *grabbing* on the virtual world. Through our formative evaluations we have strong evidence that it is an effective interface between the real world and the virtual world for controlling the virtual objects, which have no tangible reality.

To give the user more understanding of the location of the virtual objects in the real world, we used an over-head projector that displays the shadow of the virtual world that falls outside the screen on the floor, right under the interaction. This projection has been placed to make the first experience with the setup more intuitive for newcomers.

At last, to facilitate users to relate the body actions to the effects on the virtual world, the system renders the shape of the depth-data detected by the kinect. A polygonal 3D mesh is created from the depth-sensor values and displayed on screen. The fact that this mesh fits the user's body shape and placement makes users view and understand the effects of their movements easily.

See figure 10 to see the final look of the project.

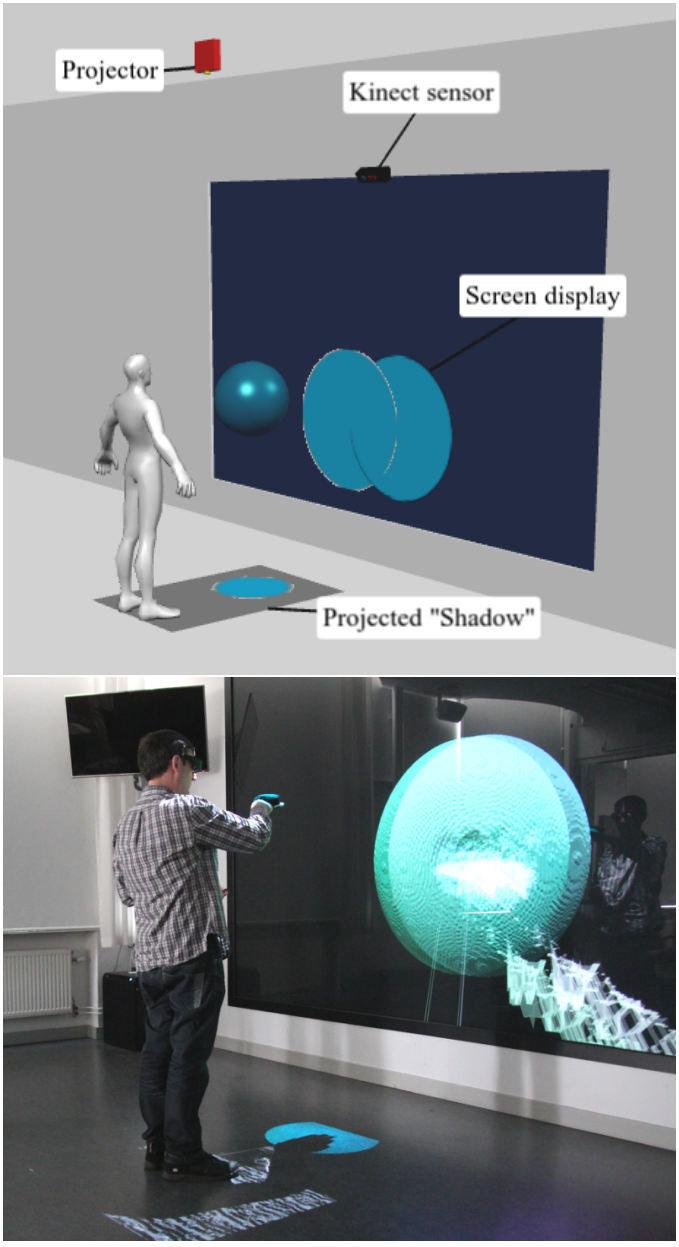

Figure 10: Embodied sculpting setup[2], above, and a user interacting with the project, below

6 Discussion

For the development of an Embodied Perspective the 3-dimensional stereoscopic rendering is essential, but we must consider if this technology is practical. In order to achieve the Embodied Perspective that we pursued we made use of the 4K, 4 meter wide, stereoscopic screen that is available in the VIC studio in order to create a virtual world that takes most of the user's field of view and attention. It is true though that this kind of display is not available for all consumers. All of the approaches that we have made during the development can be taken over other smaller stereoscopic displays in order to create a similar experience. Having a screen of such dimensions as we use makes a straightforward difference on which way and how much can the user move in front of the screen. As explained before, when having smaller screens we are forced to create smaller virtual objects in order not to break the interaction and embodied feeling. Nevertheless, the approach can still be used on other applications.

We cannot deny that the head-tracking is essential for the first-person embodied perspective. Moreover our approach uses kinect's head-tracking system thus makes it a really practical affordance. The fact that the virtual objects are so close to the user's point of view though, makes really small errors on the tracked position of the user's eye's to be really noticeable. Since we place the kinect so far away from the user we decided to make an approximation over where the user's eyes are placed. When the user comes really close to the screen, really close to the virtual object, the approximation does not work so well anymore since the user would probably not be looking at the center of the object. Moreover, from user to user, both the eye distance and head's depth varies. In order to create a truly faithful stereoscopic visualization, the project should somehow adapt to the user sizes.

We learned that the delay of the depth data and the position tracking played a big paper on breaking the embodiment and immersion on the users. The fact that we are using the first version of Kinect for Windows, which processes the depth and position tracking through probabilistic algorithms, makes the delay a noticeable negative factor. If more deterministic motion capture tools like infrared camera installations were to be used, we would surely get rid of a great part of the delay, but that would make the approach to be much less practical and more restricted to be shown outside its designed space. In our approach we always wanted to be able to change screen and emplacement without having to worry much about the installations.

Due to the angle that the user forms with the kinect (as seen in figure 3) the skeleton tracking sometimes failed and caused jumps on the user's view. We tried adding some smoothing to the skeleton prediction but then the latency between the movement and the display augmented even more. We have a trade-off between visual smoothness and responsiveness. In our project the user is supposed to be sculpting a static object, this makes the user not move much when interacting with the object, so applying prediction when the head movement is in fact so low, to increase the responsiveness, makes the prediction fail constantly causing shaking of the user's point of view. Moreover we realized that the frame-rate of the Kinect depth data has an effect on the skeleton position algorithm. The depth stream frame-rate is not constant and varies depending on environment factors such as lighting and size of the space in front of the kinect. Finally, through iterative testing, we could define the smooth-

ness/responsiveness margin that fit the best on our setup.

Having the kinect to be placed on top of the screen and its affect to the skeleton tracking really showed us that it is not the best idea to use the kinect in strange inclinations if what you we are looking for is tracking precision. In our case it was more of a trade-off between blocking the user's view or getting a smooth head-tracking, so we tried to find the inclination and position that fit our needs the most.

The introduction of a one-button one-hand controller can be considered to be an obtrusive tool to embodiment. Whereas it is relatively easy to acquire such hardware, in our case it consists of a slide-show presentation controller, it is true that we are giving the users a gadget that they are not familiar with. As explained before, we introduced the controller in order to be able to add the *grabbing* detection, a context switch, for the users to manipulate the virtual object's rotation at their will whenever they want. We have explored the detection of grabbing through cameras and depth-sensors before and there is always the difficulty to the user to mental-map the grabbing of his hand in the air to the grabbing in the virtual world. Since when grabbing objects on a real-world situation we experience a tactile feedback of the object we are grabbing, when using a grab-detection interface our brain has to constantly be aware of the state of their hand, since such feedback is not received. As discussed on the development section, the accuracy of such grab detection from depth-sensors and cameras is not satisfactory many times, usually resulting on unexpected behaviours of the virtual environment. On our setup, the placement of the kinect relatively far away from the user (around 1.5 meters) and in a non-standard inclination made us decide to not use this approach. On the other hand, having a physical button, allows the user to switch context mentally much naturally, and the learning barrier becomes really affordable due to the habit and experience of persons to use physical tools.

The *shadow* projection on the ground was introduced to visually hint the users where in the real world, outside the screen, the virtual objects were. Many users had troubles understanding the depth, distance and position of the virtual objects in front of the screen. We observed that the stereoscopic depth-perception from the 3D visualization was not enough for some people to get the proportions of the virtual environment at first glance. The introduction of this projector cannot be considered practical unfortunately. The placement of a projector on the roof facing the floor is not a very reusable setup. Moreover the projection falls on top of the users, in a certain manner bothering the user when interacting with the experience. The projection on the floor has been seen to be used by the users just the first time they walked upon the screen, in order to place themselves right in front of the virtual object. Its effectiveness is somewhat relative and ephemeral. From our part, we used the projected image as a markdown of the interaction space, where we could point the users towards. The users, after finding their place, would never look down to the shadow. On top of this, the projection is being displayed on top of the users' body when they are interacting, changing the color of their body. We do not consider this completely unobtrusive. Moreover, due to the default perspective deformation of the projection, the image displayed just faithfully relates real-world positions to virtual-world positions on the floor level.

The self-visualization through the depth-mesh on screen has been a feature that we have been adding and removing as we tested the project. This visual-

ization was the only way the previous state of this thesis' project had available to indicate to users their placement in the virtual world. Once we implemented the first-person embodied perspective we decided to take it away from the user's view because it interferes with the view of the virtual world. Nevertheless, this mesh is the direct connection between the collision-based interaction and the user's movements and body and, as stated before, there is a latency between the movement of the user and the detection of such movement. This reason lead some users who tested the experience with and without the self-mesh rendering to state that they preferred to see the rendering of the mesh. This self-visualization is not obtrusive in the sense that we defined: the screen does not block the user's view of their body. Nevertheless, the latency causes sometimes the mesh to be perceived intersecting the user's body. The fact that the mesh is exactly, in position and shape, placed where the user's body is, causes the 3-dimensional stereoscopic rendering to challenge the user's brain's depth-perception whenever the self-visualization is placed between the users eyes' and their own bodies. This effect challenges this feature's effectiveness on Embodied Perspective although is only visible on extreme cases. If the user moves their body forward to fast for the depth-mesh to keep up, this effect is appreciable.

Lastly, the fact that the virtual object did not look anything like a real-world object, made some users on early tests fail to understand that their body movements were affecting the virtual object if they didn't have any haptic feedback. The importance of the haptic feedback on this situation has been observed to be really important on previous studies[21].

7 Conclusions and future work

The first-person perspective that we have developed that we call *Embodied Perspective* has evolved into what is an unobtrusive, practical and effective method to create a fully embodied immersive experience on the user through 3D-stereoscopic rendering and the use of a depth sensor. We have concluded that the single use of a Kinect depth sensor and a big stereoscopic screen can achieve great immersion and embodiment even to inexperienced users.

This project has been regularly showcased and tested during its development to both students, professors, non computer-related people and even real sculptors. In order to adapt the experience to widest range of users possible we have reworked, refined and tweaked the project through an iterative design during the period of 5 months of development.

During the later iterations of the project we had the chance to let professional sculpture teachers from Konstfack[6] try the experience and give us feedback. Most of them were really impressed and interested on the learning possibilities for sculpture students that our project has. They provided us with feedback about, over anything else, how to improve the feeling of reality on the material, on both the haptic feedback and the sound feedback. We even had the chance to visit and examine their sculpting facilities in Stockholm as they showed us their work. They were really concise on exposing their believe and conviction that the most important value to learn for a sculptor is the tactile feeling of the material, on how we should spend more time into making the material feel more natural, which would make the learning experience more enriching.

We planned to run a user study to evaluate the work that has been made on the project during the time where I have been working on it but we finally ran out of time and people when we were ready to test it.

Nevertheless the project is fully prepared for a single user case study. Even we have thought on group studies to evaluate the effect of this perspective on group learning and showcase. Since the experience is run on a such big screen and room, usually the project is showcased to multiple people at the same time. With the embodied perspective it is true that the user in charge of the head tracking receives the maximum immersion and the best part of the experience, although the audience makes always an influence on the user.

The fact that the audience can still see the 3D-stereoscopic view that the main user has but in a different perspective, from behind or the sides, and the fact that the shape of the main user is somehow visualized on the screen, makes the audience learn and ask questions and petitions to the sculpting user. We have observed how the audience tries to interact enthusiastically with the sculptor in order to understand more how the experience works. The audience never hesitates to ask the main user to switch perspective and to let other people try and see. In a way it becomes a really interactive experience where everybody understands and can enjoy the learning and interaction.

During the development of the project, several professors and researchers at KTH have shown their interest into the future research over the area regarding their respective fields, from haptics research to machine learning. After several meetings with these people, we have prepared the project as much as possible in order to facilitate future research. For this purpose several tools and arrangements have been developed.

A 3D-model exporter with a common standard format[11] for triangular

meshes has been developed in order to be able to export the sculpture at any point of the experience to be able to visualize and import it later on any 3D environment or even 3D-print it.

A proper data export method for the full experience has been designed and developed. In our case, every time a session is run, our program exports the voxel information in the previous mentioned mesh format as well as the time stamp for each voxel, that tells us when has the voxel been removed from the virtual object, and a file containing a discrete collection of data relative to the object's position and rotation and the skeleton tracking positions of the user. Since we have a finite amount of voxels, for each voxel we store at which point in time was the voxel removed. For the storing of the data relative to the position and rotation of the object and the positions of the skeleton tracking, a certain period for data collection is defined in order to save and store the positional data periodically. Using the data generated in each run, the whole session can be recreated by following the recorded coherent time stamps.

We hope that *Embodied Sculpting* can be explored further and deeper in order to provide as much data as possible for future research on methods of interaction, social learning and machine learning, as well as other fields we are now not yet contemplating.

References

[1] 3d Video Capture with Three Kinects
http://doc-ok.org/?p=965.

[2] 3d web visualizations of the project
http://fonserbc.github.io/milo/.

[3] Computer Automatic Virtual Environment (CAVE)
http://en.wikipedia.org/wiki/cave_automatic_virtual_environment.

[4] Github repository of the project Embodied-Sculpting
https://github.com/jonandre/virtual-sculpting.

[5] Google Glass
http://www.google.com/glass/start/.

[6] Konstfack University
http://www.konstfack.se/en/.

[7] Microsoft Kinect for Windows
http://www.microsoft.com/en-us/kinectforwindows/.

[8] Oculus Rift - Virtual Reality Headset
http://www.oculusvr.com/.

[9] Panavision 3D by panavision inc. and omega optical inc
http://www.panavision.com/newsarticles/new-cinema-system-panavision-makes-3d-%e2%80%98simple%e2%80%99.

[10] Simple Directmedia Layer
https://www.libsdl.org/.

[11] Stereo Lithography (STL) format
http://en.wikipedia.org/wiki/stl_(file_format).

[12] Virtual Sculpting
http://www.nada.kth.se/∼benned/sculpt/.

[13] Visualization Interaction Collaboration (VIC) Sthlm
http://vicstudion.se/.

[14] zlib License
http://www.gzip.org/zlib/zlib_license.html.

[15] AirWave: Non-Contact Haptic Feedback Using Air Vortex Rings. 2013.

[16] R. Blanch and E. Ferley. Non-Realistic Haptic Feedback for Virtual Sculpture. 2004.

[17] Harrison C., Tan D., and Morris D. Skinput: Appropriating the Body as an Input Surface. *SIGCHI*, 2010.

[18] Kuester F., Duchaineau M.A., Hamann B., and Joy K.K. 3DIVS: 3-Dimensional Immersive Virtual Sculpting. *NPIVM*, October 2001.

[19] T. A. Galyean and J. F. Hughes. Sculpting: An interactive volumetric modeling technique. *ACM SIGGRAPH Computer Graphics*, July 1991.

[20] R. Jagnow and J. Dorsey. Virtual Sculpting with Haptic Displacement Maps. *MIT*, 2002.

[21] Romero M., Andreé J., Thuresson B., and Peters C. Designing and Evaluating Embodied Sculpting: a Touching Experience. *CHI*, 2014.

[22] Greuter S. and J Roberts D. Controlling Viewpoint from Markerless Head Tracking in an Immersive Ball Game Using a Commodity Depth Based Camera.

[23] Hoshi T. Development of Aerial-Input and Aerial-Tactile-Feedback System. *IEEE World Haptics Conference*, 2011.

[24] Krueger W. and Froehlich B. Visualization Blackboard: The Responsive Workbench. *IEEE Computer Graphics and Applications*, May 1994.