

Being and Thing become space[s].

Reflecting // Interacting // Mirroring

June Kim

This exegesis is submitted to the Auckland University of Technology

Master of Creative Technologies, July 14th, 2015

TABLE OF CONTENTS

LIST OF FIGURES	4
ATTESTATION OF AUTHORSHIP	6
ACKNOWLEDGEMENTS	7
ABSTRACT	8
INTRODUCTION	9
1	
DRAWING //	11
FRAMING	
SPACE, BALLOON AND BUBBLE	12
INTERACTIVE ART AND EMBODIMENT	14
DISEMBODIMENT AND EMBODIMENT	17
SCREEN AS FRAME	20
LIVENESS	22
VISUAL NARRATIVE.....	23
2	
EXPERIMENTING//	25
BUILDING //	
REFLECTING	
EXPLORATION OF CORE TECHNOLOGIES	26
openFrameworks	27
Arduino	29
Microphone	31
Bluetooth Communication	33
Kinect 1 and 2	34
PROCESS OF CODING AND PHYSICS BEHIND	38
Creating the Project's Scaffold	38
Code Structure	40
Role of Configuration the config.xml File.....	41

Algorithm	42
Simulation of Balloon – Soft Body Physics.....	43
EXPERIMENT AND DEVELOPMENT OF	46
VISUALISATION AND INTERACTIONS	46
Visualisation.....	46
Interaction methods.....	49
Set up for blowing machine – prototype stage.....	52
Interaction with Breath	56
Interaction with Hand gestures.....	59
Interaction with Sound.....	66
 3	
CLOSING	67
I SEE ME INDEED	68
 BIBLIOGRAPHY	72
APPENDIX A	76
APPENDIX B	80

LIST OF FIGURES

Figure 1: Space.	12
Figure 2: Balloon and Bubble.	13
Figure 3: Diagram on embodiment relations.	19
Figure 4: Golan Levin and Kyle McDonald: Rectified Flowers.	27
Figure 5: Arduino board, microphone sensor and Bluetooth shield used in the project.	29
Figure 6: Arduino IDE with open sketch SimpleAnalogFirmata that can be used to communicate with openFrameworks-based applications.	30
Figure 7: Kairalla: Blowing Sensor	31
Figure 8: Kairalla: Breathless	31
Figure 9: Microphone sensor.	32
Figure 10: Bluetooth Shield used in the project (see http://tron.cc/btsh).	33
Figure 11: Microsoft Kinect version 2, for Windows.	35
Figure 12A. Gesture recognition by the system.	36
Figure 12B. Gesture recognition by the system.	36
Figure 13: openFrameworks project generator application.....	38
Figure 14: openFrameworks project generator application: list of available add-ons...	39
Figure 15: Xcode with open project junikBubble.	40
Figure 16: Diagram of the interactive application.....	41
Figure 17: Sample 3D object (sphere) with normal vectors (white).....	43
Figure 18: Textured 3D object with normal vectors used as texture coordinates. Duck.	45
Figure 19: Textured 3D object with vectors used as normal texture coordinates with a Stanford Bunny model.	45
Figure 20: Initial stage of the sack with a bounding box, meshes and normal vectors displayed.	46
Figure 21: Initial stage of the sack from a different camera view.	47
Figure 22: Mirroring (with vectors normal visible).....	47
Figure 23: Mirroring (installation view).....	48
Figure 24: Interaction with mouse: grab & stretch using pressing, dragging and releasing functions	50
Figure 25: Conversion of interacting functionalities between mouse and kinect.....	51
Figure 26: Wireless setup includes: 9V battery, Arduino board, and Bluetooth Shield, microphone and power switch.....	53
Figure 27: Wireless setup.	54
Figure 28: Soldering the power switch.....	54
Figure 29: Prototype Megaphone setup 1.	55
Figure 30: Prototype Megaphone setup 2.	55
Figure 31: Default state of a sack.	56
Figure 32: Blowing breath into the sack 1.	57
Figure 33: Blowing breath into the sack 2.	57
Figure 34: Blowing breath into the sack and interacting with a hand gesture.....	58
Figure 35: Testing interactions with Kinect.	59
Figure 36: Gestures calibrations (through Kinect SDK 2).	60

Figure 37: Interaction with the sack (right hand).....	61
Figure 38: Interaction with the sack (left hand).	61
Figure 39: Interaction with the sack 1.	62
Figure 40: Interaction with the sack 2.	62
Figure 41: Kinect detecting two people 1.	63
Figure 42: Kinect detecting two people 2.	64
Figure 43: Two people influencing the sack 1.	64
Figure 44: Two people influencing the sack 2.	65
Figure 45: Two people influencing the sack 3.	65
Figure 46: Interaction mode.....	66
Figure 47. Pamphlet for the Interactive Installation, experiment.....	81
Figure 48. Interactions with the bunny using key strokes, at the QUT	82
Figure 49. Interactions with the bunny using touch screen, at the QUT.....	82
Figure 50. Interactions with the bunny, using mouse, at the QUT.....	83
Figure 51. Interaction with the bunny.....	83

ATTESTATION OF AUTHORSHIP

I hereby declare that the submission is my own work and that to the best of my knowledge and belief, it contains no material previously published or written by any other person nor material which to a substantial extend has been accepted for the award of any other masters, degree or diploma of a university or other institution of higher learning, except where due acknowledgment is made in referencing.

A handwritten signature in black ink, appearing to be 'J. Smith' or similar, written in a cursive style.

ACKNOWLEDGEMENTS

I would like to thank my supervisors James Charlton and Kim Newall for providing guidelines and encouragement at the Colab. I also wish to acknowledge the significant contributions and teaching in technical concepts, inspiration and encouragement of Dr Tomasz Bednarz, Associate Professor at the Queensland University of Technology. Finally I would like to thank my mother, Grace Cha for her support and love and Dr Marcus Henning for ongoing academic support and encouragement.

ABSTRACT

This research explores notions of embodiment and aesthetics of interaction within the context of the construction of meaning for interactive art. While investigating notions of the construction of meaning, this research draws attention to audience engagement with an interactive art installation during the interactive process.

An interactive artefact 'Indeed' is a process that embraces my reflections of theoretical frameworks based on personal views and studies in engaging and experimenting selected technologies in relation to creating interactivity.

INTRODUCTION

In the midst of crowds and conversations...

I often invisibly switch my eyes and ears off from my surroundings and turn these senses inward. I listen and feel myself. I begin sensing myself as an organic form of sack. I experience myself as an evolving modular form of the sacks that I interpret as space[s]. Beings and things become spaces. I visualise and sense many single and modular, separated and overlapping organic transformative spaces. Here, I play with my breath and motions to make space[s] bigger and smaller and to reshape them by dragging them in various directions.

This research was initiated by my desire to enact this sensing experience as a *meaningful* interactive artefact. The resulting artefact – *Indeed*, constructs a space that immerses spectators in their own narrative through interaction.

This research embraces the opportunity to challenge theoretical frameworks and to experiment with and implement interactive technologies coherently. This exegesis includes three sections:

- Drawing // Framing
- Experimenting // Building // Reflecting
- Closing

Drawing // Framing focuses on the process of sketching a visual narrative and of searching for meaning and context through associated philosophical frameworks

and artefacts created by other artists. The **Drawing // Framing** section includes reflective anecdotes which are indicated by italics and #. In terms of constructing and fabricating a meaning for *Indeed*, I delved into my own sensing experience. This interwoven reflective anecdote describing my memories on selected theoretical frameworks represent the essential process of research.

Experimenting // Building // Reflecting discusses the process of developing *Indeed* using selected technologies. The **Closing** section reviews and reflects on *Indeed* based on reflective anecdotes that describe my experience of embodiment.

1

DRAWING //

FRAMING



Figure 1: Space.

The above figure is an image of *Indeed* in my mind. The undefined form epitomises the notion of 'I' and 'Wander' while the black background depicts the enclosed surroundings. I conjure up the characteristics of this form - floating, malleable, lively, temporal, inflatable, deflatable and embrace them with 'breath'. Furthermore, this drawing also conceptualises my ideas about viewing interactive [arts].

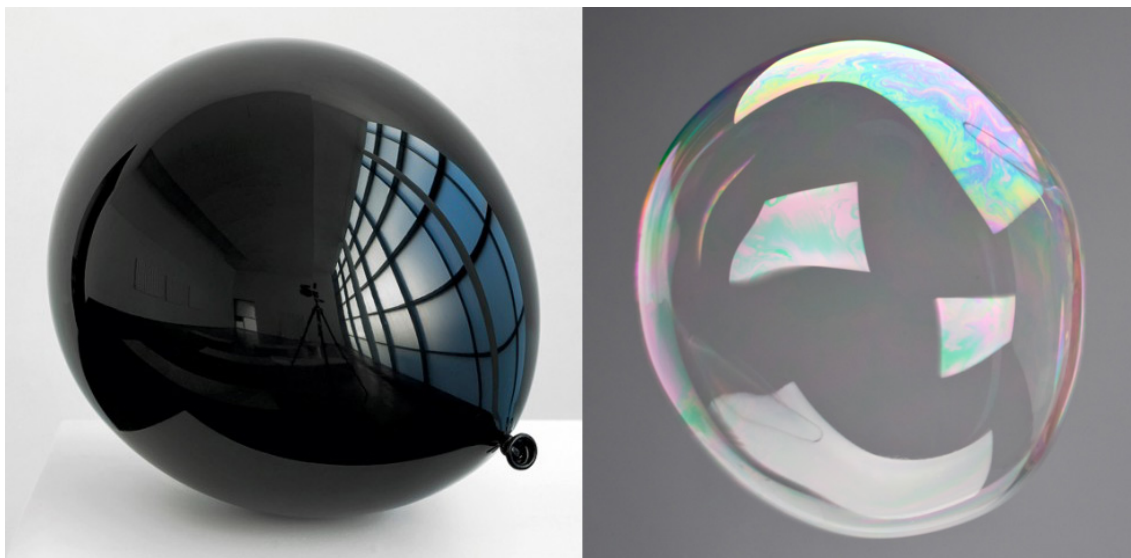


Figure 2: Balloon and Bubble.

When I was little...

I would occasionally go out and walk on streets with my mom. She would loosely strap the string of a balloon to my wrist. It's time to go back home. I feel something is missing from my wrist. My balloon! It flies into the sky higher and higher very fast. I gaze at it for a long time; I cannot take my eyes off my balloon. My mom asks me to wave my hand and say 'bye' to my balloon. She says that it's time for my balloon to go home. I am sad.

Today, I am back home with my balloon! I have a very good evening showing my balloon around my house just in case it needs to go to the bathroom at night. I sleep with my balloon. Where are you? As soon as I open my eyes, I search for my balloon. My balloon seems very exhausted and shrunken. It does not float like it did the day before. I am sad.

I say 'bye' to all my balloons. I no longer want to feel sad.

Mom, look! I delivered a bubble! I am extremely fascinated by the fact that I can create forms that look like balloons both big and small, short and long. I calm myself down and take a long breath to make it big. I also make a series of small bubbles. With my breath, I whisper, giggle and laugh. I make a whisper breath, giggle breath and a laugh breath. I make thousands of bubbles all day but none of them are with me after all. I am no longer sad.

INTERACTIVE ART AND EMBODIMENT

I engaged with balloons as if they were my closest kin. Additionally - and unintentionally - I was sad at every farewell to each balloon. Donath's notion (Fiefield, 2008) of 'subjective interactivity', in other words, the "illusion of autonomous interactivity" (p. 10), can be applied here. However my balloons were inert; my illusory endearments rendered them autonomous and subjective, so that interaction might occur. Humans have a long history of regarding belief as the existence of autonomy in non-human matter. An example of this can be found in totemic belief systems.

On the other hand, it is seemingly difficult to relate the notion of subjective interactivity to the bubbles. I do not view the bubbles as distinct bodies, nor, as autonomous entities. There were no intimate interactions created between my movement and the balloons. On the other hand, I could initiate changes in bubbles by subtle changes in my movements. I felt embodied the *process* of making bubbles, even *situating* and feeling bubbles as being my altered self.

Thus, I perceive the 'process of making bubbles' as one inseparable event. In practice, my first actions are to play with my breath, a modality that I intend to explicate as a semeiotic concept. Then by being immersed in practicing these actions, I become embodied by the situation. Upon reflection of my personal experience, I formulated an interpretation of the balloon as a metaphor for conventional art and the process of making bubbles as a metaphor for interactive art. Furthermore, I investigated the essence of interactive art as embodiment and compare it with Nathaniel Stern's (2013) account of interactive art as providing *situations* where audiences may develop their own meaning and understanding. I identify this as the process of creating a personal narrative through interactions. When we move - think - feel, we are a body. Rather than viewing this body as a thing, Stern emphasises the constant relationships that it is 'on' and 'through' and a dynamic form of changing which ensues embodiment as incipient activity. Stern described interactive art as a work of process of moving-thinking-feeling, which is embodiment. (p. 2).

However, my interpretation of the process of making bubbles as an interactive art is founded by the concept of process. I view neither play, games nor performance as an interactive art. For this thesis, I will define interactive art as a system of sensors, processors and output modules which respond to audiences and open a space for those audiences to develop their own meanings. Even though interactive art contains conventional elements such as sculptural or installational, the technologies play a central role in manifesting the interaction in the interactive art (Kwastek, 2013, p. 143). Thus, with the technological development of interactivity, more interactive artworks and projects are appearing in both site-specific and

gallery spaces. However, in order to define a new form of art, people tend to focus on new methods of embedded technologies and interactive experience only. They may overlook the meanings of interactive art and the potential for contribution of interactive art to contemporary art from both the creators' and viewers' perspectives.

I view technologies as tools to create *situations* and focus on “what interactive art does – what we do – when it frames our moving – thinking – feeling” (Stern, 2013, p. 6). Furthermore, I comprehend the process of art as not just making, but questioning. Artists do not just utilise new technologies, they consider them as being more important than seeking new structures, meanings and metaphors by creating artworks and questioning new technologies. In *Video Art*, Michael Rush considers Bruce Nauman's practice draws attention to the artistic process, establishing it as being as important as the artefact itself (2008, p. 125).

DISEMBODIMENT AND EMBODIMENT

A place with many mirrors

My mom's friend runs a boutique, I call it "a place with many mirrors"

Mom started chatting with her friend, so, for me, it is time to play with mirrors. I mirror my entire body on to the first piece and slowly move to the side and see a sharply sliced half of me. I love viewing a part of my body and that mirrored part of me reflecting onto other mirrors. I occasionally feel haunted seeing sharply cut body part and multiple copies of them. I feel that I am cut into pieces. On the other hand, I feel that those reflected images of parts are not mine.

I am confused but I am amused.

I look different from different mirrors, it means...

In the boutique, I often heard that people ask staff to bring other mirrors, as they did not appeal nicely in the given ones. Maybe because I was too little, or maybe I was overly amused to my play, I did not pay attention to what they meant.

Now, I understand it!

I look chubbier with this mirror! I look shorter with this mirror! Should I get a new mirror?

Hang on a second...

That mirror may be right. But the other mirror, which reflects me prettier may be lying. I now see that no one mirror reflects the same me. I look different with each mirror. Which one is right? Have I ever been able to view a true image of myself through a mirror?

Hang on a second...

What about through the eyes of other people? It is unpleasant to think that I would look different to different people, and that I cannot view how they see in me. 'How do I look to my boyfriend?' I wonder.

I stand in front of a huge and somewhat shiny apple sculpture. I started jumping around the apple! A friend of mine asks me what I am doing? I told him I see me. He says that I would not be, as it is neither a mirror, nor a shiny object like a mirror. But I see me. I see a natural shape of darker spots moving and dancing. That's me!

Without wearing glasses, to me I am always somewhat blurred, merged in an undefined organic form to my view.

I have been wearing glasses since I was seven. Since then I have always had two different worlds. I have only memories of being able to see clear outlines without having my glasses on. Thus, to me there is significant difference between these two worlds and I often intentionally draw two different versions of scenes.

In the essay, *Technology and Embodiment in Ihde and Merleau-Ponty*, Philip Brey (2000) introduces Don Ihde's notion of embodiment relations between humans and artefacts. In contrast to most other objects, some artefacts such as 'telescopes', 'glasses' and 'hearing aids' work as 'means' rather than 'objects'. Hence, humans perceive 'visual', 'aural' and 'tactile' information through these artefacts, which then illuminates their status as objects.

Embodied artefacts and technologically mediated effects change the way we relate to the world. However, since many embodied artefacts are technologically mediated, we are also required to comprehend and be familiar with technologically mediated artefacts.

Philip Brey (2000) argues that Ihde's account of embodied relations as a perceptual entity is limited and it appears to be more like a 'behaviour'. Instead he proposes an extension of Ihde's theory through Merleau Ponty's phenomenological perspective in which embodiment defines the 'nature of the human body and of perception,' rather than only distinguishing it through technology.

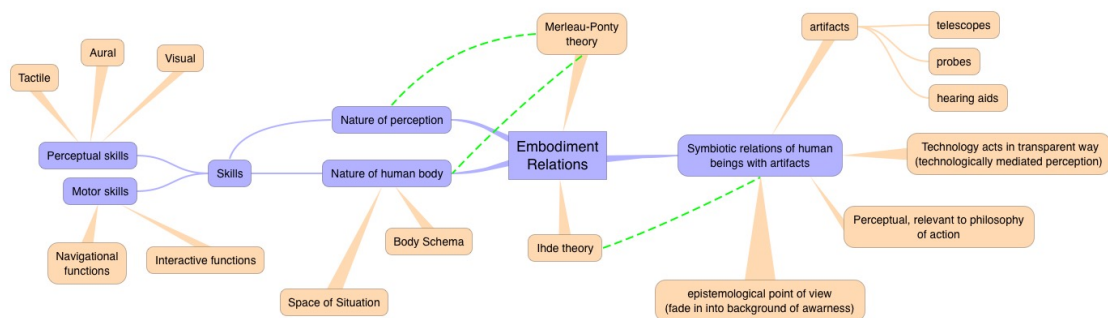


Figure 3: Diagram on embodiment relations.

Embodiment relations shown in Figure 3, could therefore be understood and explained in three ways: human-technology relationships, the nature of the human

body and perception. The nature of the human body is mediated through a skill set that could be either perceptual (visual, aural, tactile) or motor skills (navigational, interactive). Perceptual skills are found in artefacts designed to mediate perception. Motor skills (navigational functions, interactive functions) explain relationships with the body as 'space of situation' and are characterised by body schema and body image.

Everyday, I see myself through many different things and beings. I feel that all things and beings contains shiny and reflective hidden persona even in an abject with the roughest surface. And I suppose that we do not have to use only vision to capture the reflection. I see myself while looking at the back of a homeless person, I see myself in from the my mother's voice and I see myself in my old handwriting.

SCREEN AS FRAME

Through mirrors, I see me. Through screens I see others. Through mirrors, I see now. Through screens I see past. Mirrors do not replay but screens do replay. I want to see me and now through a screen. I want to remove the barrier between the mirror and screen.

In the *Aesthetics of Interaction in Digital Art*, Katja Kwastek (2013) defines Interactive Art as a hybrid form of art that combines components of the visual arts, the time-based arts, and the performing arts. Earlier, Belting (Rush, 2001) noted that boundaries, which existed between the visual arts and the performing arts, have become blurred. Borrowing from Belting's account, I see interactive art as being located in the overlapping boundaries of visual arts and the performing arts. With this standpoint, I interpret the notion of liveness in the interactive art area.

Dixon introduces the phenomenological perspective on *Liveness*. He reflects that the sense of technology has affected changes in our perceptions of 'liveness', 'presence' and 'real'. Furthermore, he introduces Jean Baudrillard's notion that "The Virtual replaces the real" (p. 127). Before introducing the phenomenological perspective on *Liveness*, Dixon clarifies that regardless of a sense of convergence or confusion, there is a distinction between the live and the recorded, the virtual and the actual, in both perception and phenomenology.

Dixon suggests that a phenomenological perspective on *Liveness* addresses itself as time and "now-ness," rather than the notions of corporeality or virtuality and regardless of the performance forms (recorded or live). Dixon proposes that a phenomenological examination opens the ontologies of media forms by simplifying the problem. But he also emphasises that the phenomenological perspective oversimplifies the problem, overlooking the indisputable contrast in our perception of live and recorded media, of corporeal and virtual.

I am aware that it is difficult to judge whether a performance is live once a performance or multimedia project incorporates mediated footage. Although Dixon does not welcome Phelan's notion towards *Liveness* enthusiastically, even in relation to its ontological position. It is certain that we would not call recorded live performance a live performance but a documentary film of the live performance. However, if I watch motion captured footage of a live performance, I can call it a live performance depending on the degree of dominance that the mediated footage has. Furthermore, frequently mediated aspects convey the dominant of aesthetic throughout the performance.

VISUAL NARRATIVE

When a spectator appears in the installation space, a virtual mirroring sack will appear. Up to six viewers can simultaneously interact with the sack via a megaphone and hand gestures. The sack responds to the audience by changing its shape. The image it reflects is also deforms organically. Both 'breath' and 'hand gesture' are used as metaphors for the influence of self on the surroundings. I recognise that a human metaphorically changes their shape (both physiologically and mentally) because of their own thoughts, but also due to external influences. Other people can have tangible influences on us—the interactions with the sack symbolises this. The sack can be moved by another person, but the first person can

still see their own distorted reflection in it. We are always influenced by other people, but whether we want to remain uniquely ourselves, or follow their example depends on our will (for example, we might ask a person to leave the installation space – how will they respond?). And, if the person leaves the installation space, the influence he or she created remains in the scene for some time, similarly to our lives: many people are not with us today, but they hugely influenced our way of perceiving the world.

2

EXPERIMENTING//

BUILDING //

REFLECTING

“The essence of technology is nothing technological”

- *Martin Heidegger*

In selecting the technologies used to build *Indeed*, I had two conditions in mind: building a methodological (relatively universal) system that I could use and build upon, in future projects, and learning new techniques by doing and creating my art installation. Based on these, I selected the following technologies:

- openFrameworks
- Arduino
- Microphone
- Bluetooth
- Kinect

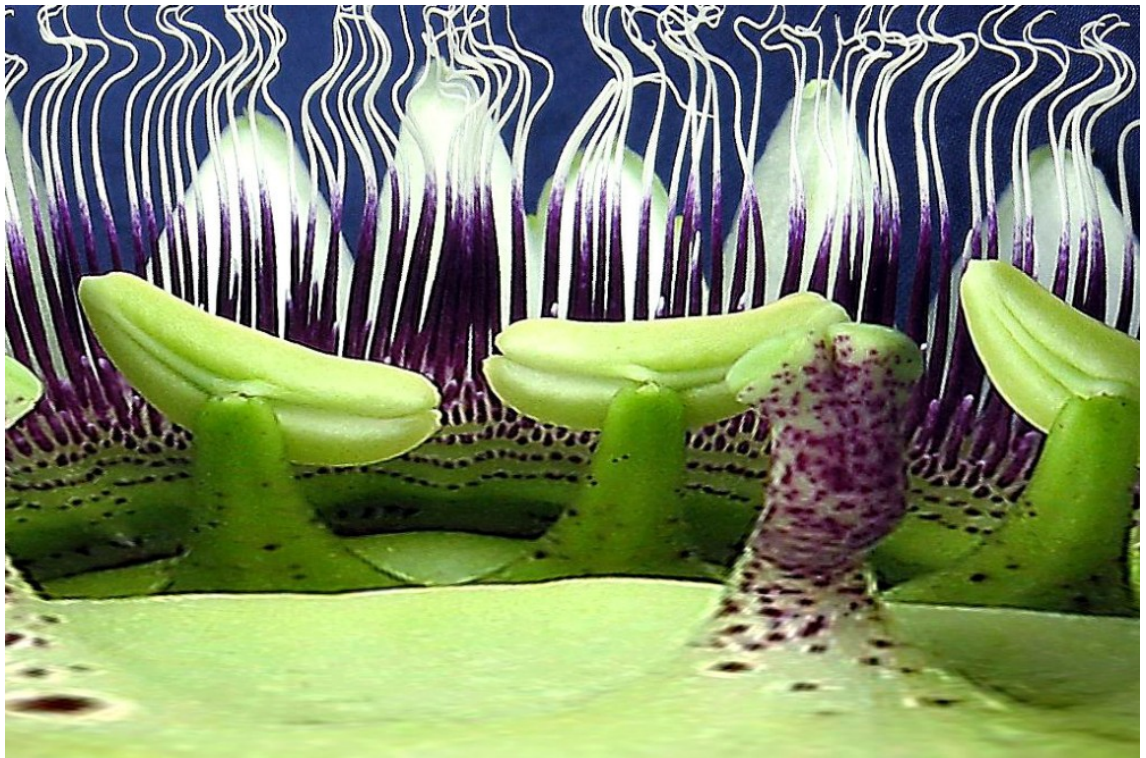


Figure 4: Golan Levin and Kyle McDonald: Rectified Flowers.

In a Colab workshop I attended last year, Golan Levin (2014) talked about sharing open source technologies and helping each other solve logistic and artistic problems. In fact, because I held the intention to participate with a greater level of artistic experience, it was inconvenient for me to think of the arts and technology as two parallel fields.

OpenFrameworks (OF) is an open source C/C++ toolkit for creative coding (openFrameworks, n.d.) designed to assist in developing creative real-time interactive projects.

I chose to use openFrameworks for developing *Indeed* due to its rich built-in functionality and support for low-level data processing libraries from a community

of makers. These benefits enabled me to understand the process of coding as well as my specific technical requirements.

Arduino

Arduino (n. d.) is an open-source microcontroller platform for building digital devices and interactive objects that can sense and control the physical world (Blum, 2013). [Figure 5](#), shows an Arduino board (with six analogue pins and 14 digital pins), Freetronic Microphone, Audio and Sound Pressure sensor, as well as a 5V Freetronics Bluetooth Shield, used for the experiments conducted for *Indeed*.

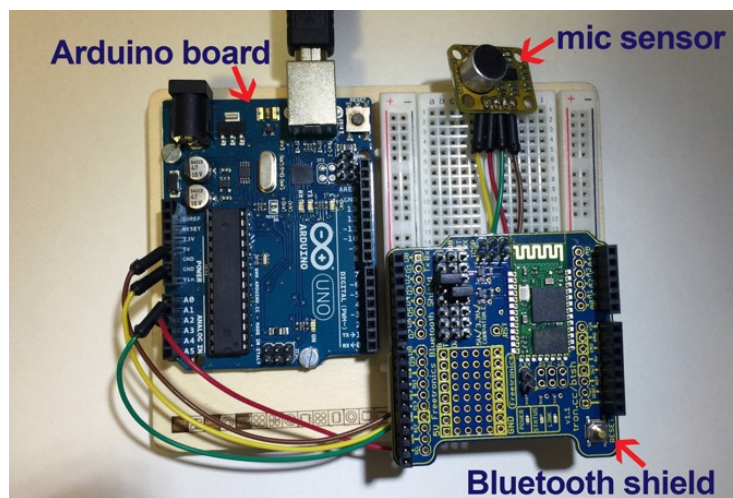


Figure 5: Arduino board, microphone sensor and Bluetooth shield used in the project.

The initial testing of Arduino for *Indeed* was done in Unity using a plug in asset, Uniduino. This was subsequently transferred to OpenFrameworks, which can control an Arduino (when connected via USB) using the Firmata sketch uploaded onto the board ([Figure 6](#)).

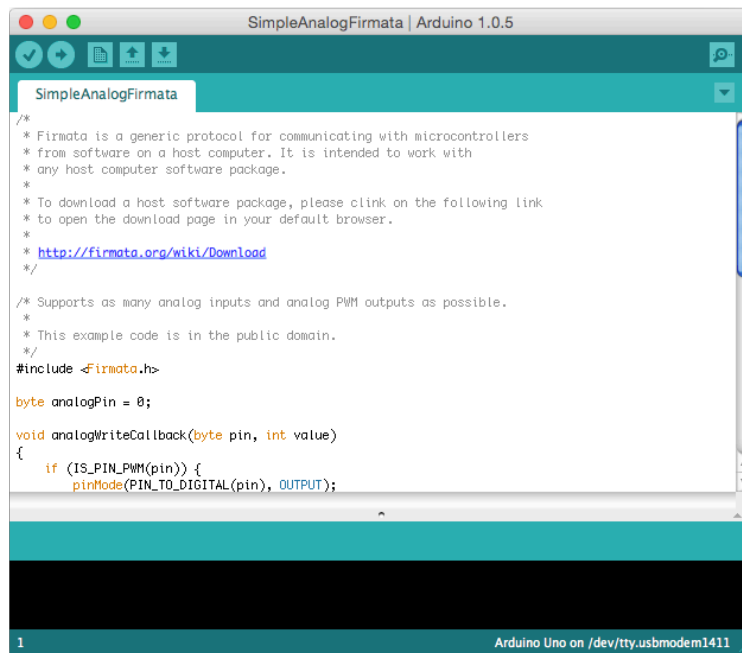


Figure 6: Arduino IDE with open sketch SimpleAnalogFirmata that can be used to communicate with openFrameworks-based applications.

Microphone

In terms of detecting breath, I utilised the type of 'Breath Sensor' that Chris Kairalla (n.d) used. His breathless project is shown in Figure 7 and Figure 8.

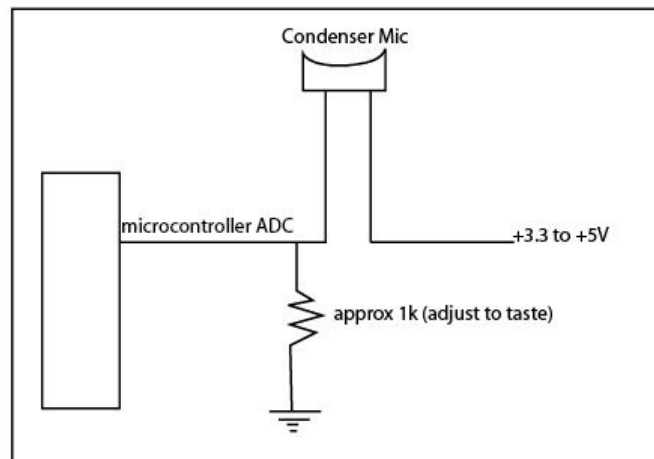


Figure 7: Kairalla: Blowing Sensor

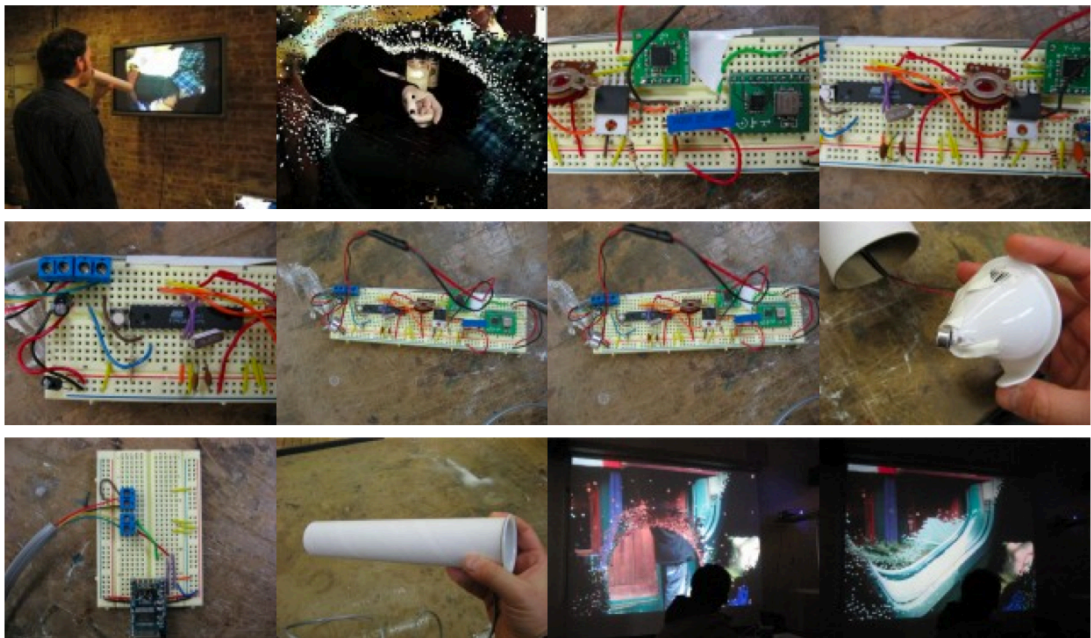


Figure 8: Kairalla: Breathless

Based on this reference, I built my breath sensor using a module that includes an omnidirectional microphone, as seen in [Figure 9](#).

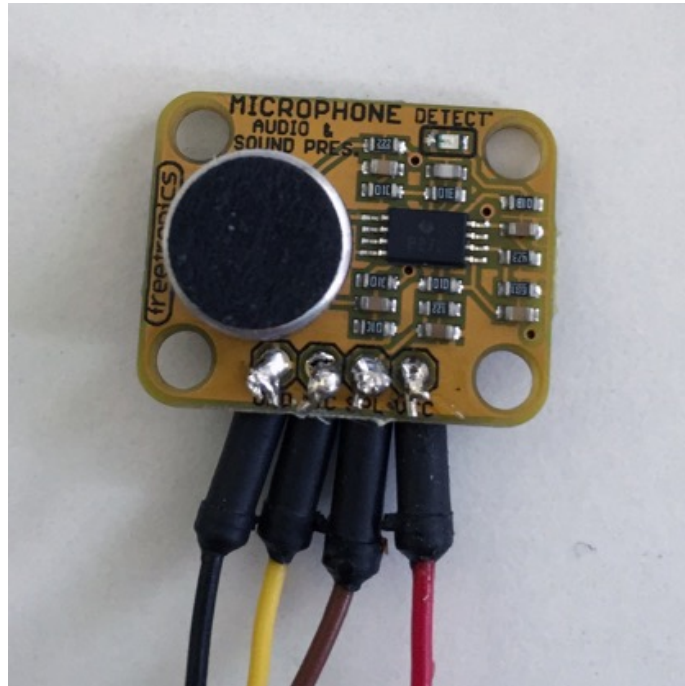


Figure 9: Microphone sensor.

The microphone sensor consists of an integrated dual signal amplifier that converts the sound to separate channels for pulse/frequency measurement, as well as volume, or Sound Pressure Level (SPL). There are two outputs provided: MIC output for measuring the raw signal waveform, and SPL. The sensor also has a green LED, which provides an immediate visual feedback whenever the module detects a sound. After experimenting with blowing, SPL was found to be the most reliable measure for blow-strength for the art installation.

Bluetooth Communication

A Freetronics Bluetooth Shield shown in [Figure 10](#), allows Arduino to create a wireless connection to a computer and send data from the attached sensors – in this case, analogue readings from the microphone connected to port A0.

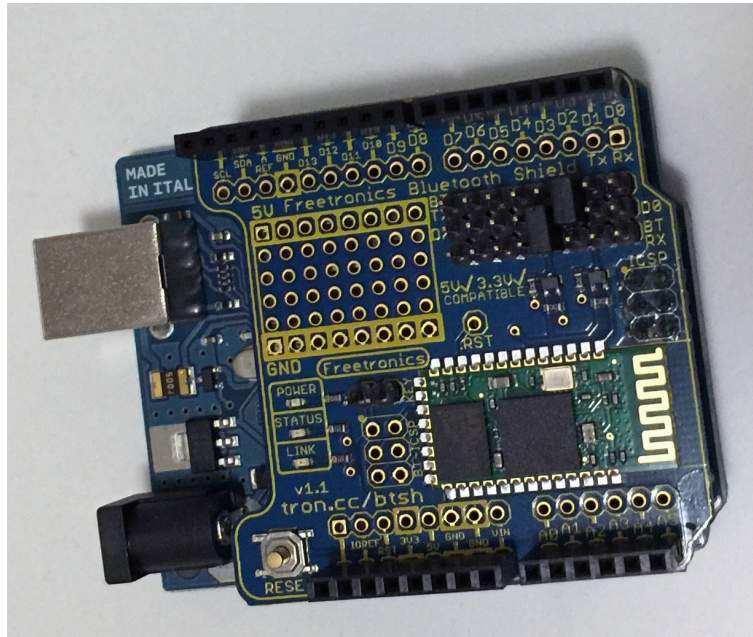


Figure 10: Bluetooth Shield used in the project (see <http://tron.cc/btsh>).

The setup was tested and executed according to the instructions available at the Freetronics (n.d.) website.

Kinect 1 and 2

Microsoft Kinect was first launched in 2010 and its name was chosen as a portmanteau of the words kinetic (producing movement) and connect (Microsoft, n.d.). It is a motion sensing input device that provides a natural user interface, allowing users to create innovative and natural human computing solutions across a variety of interactive applications. The Kinect uses a depth sensor – an infrared laser projector that uses speckles to create a dot-pattern combined with a monochrome CMOS sensor, which can ‘see’ in 3D and create a skeleton image of a user. It detects movements by analysing spatial changes in this pattern. The sensing range of the depth sensor is adjustable and Kinect software is capable of automatically calibrating the sensor, based on the user’s physical environment.

Additionally, the Kinect system is able to identify individual people through face recognition and can understand spoken commands using speech recognition software. During the development of *Indeed*, the second version of the Kinect was used, with increased depth fidelity, full 1080p colour video, improved body tracking and new active infrared detection (Microsoft, n.d.).

Kinect version 1 is a great device for body tracking; however, Kinect version 2 (see [Figure 11](#)) offers greater accuracy, and is able to find the user’s joint positions and accurately track hands and thumbs.



Figure 11: Microsoft Kinect version 2, for Windows.

In order to create more coherent interactions between hand gestures and the sack, I used the Kinect 2. Microsoft Kinect official Software Development Kit (SDK) 2.0 was retrieved from Microsoft (n. d.) website and addOn ofxKinectForWindows2 for integration with openFrameworks was used. The project was migrated to Windows 8.1 due to requirements of the Kinect SDK 2.0.

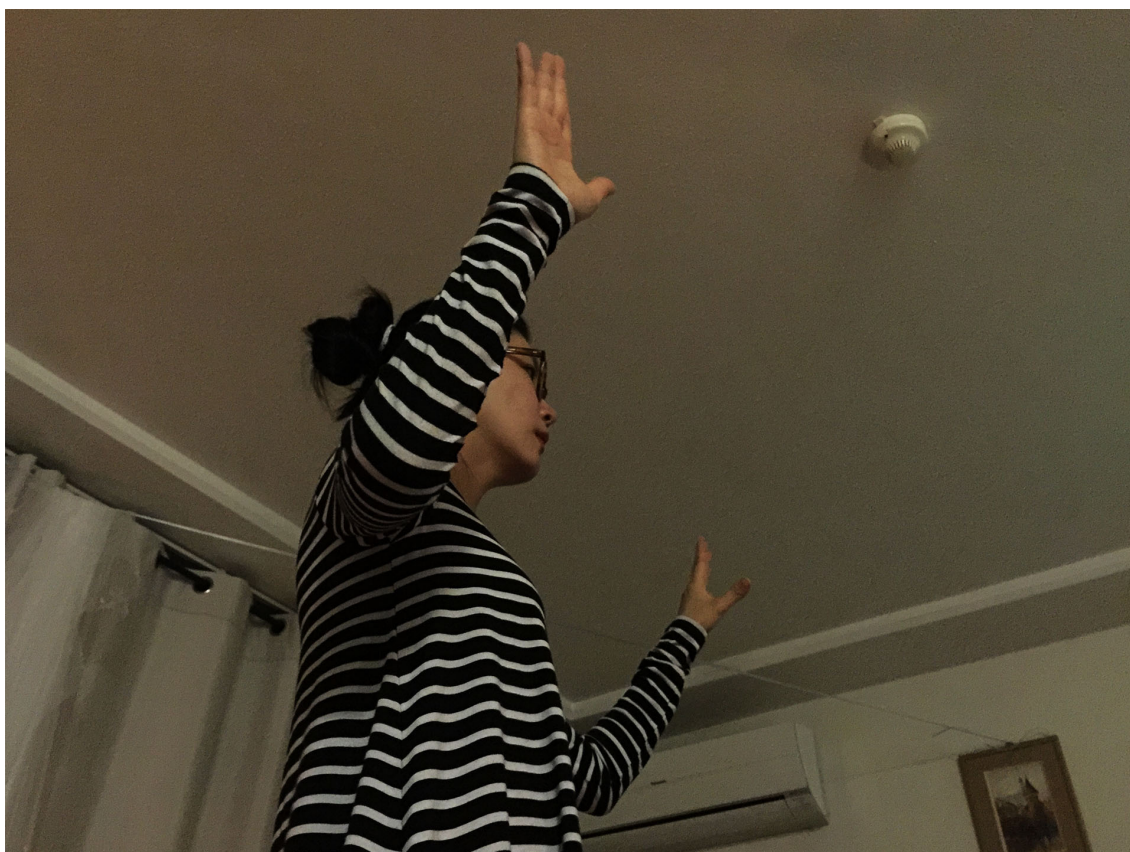


Figure 12A. Gesture recognition by the system.

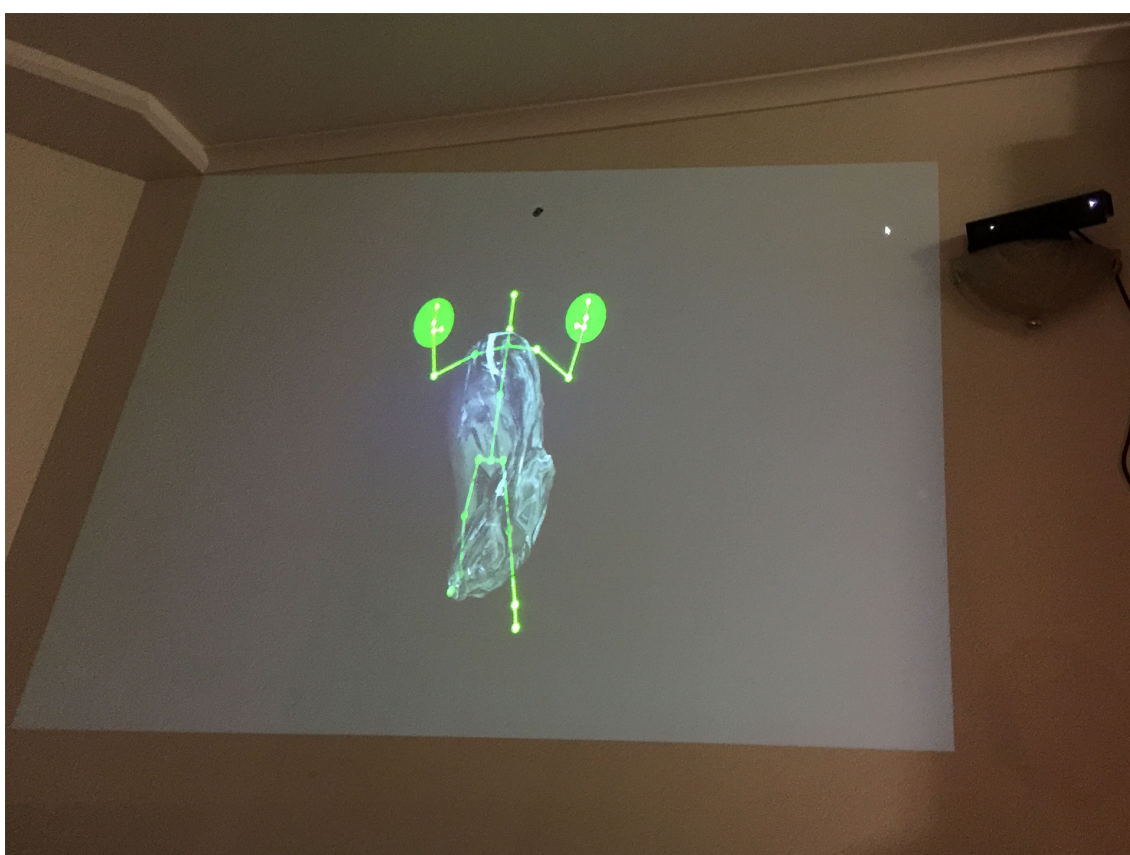


Figure 12B. Gesture recognition by the system.

Creating the Project's Scaffold

I initially used openFrameworks version 0.8.4, downloaded directly from the openFremworks (n.d.) website and installed for Xcode on Mac OS X. However, the code was later recompiled under Windows 8.1 using Visual Studio 2013 (due to Kinect SDK 2.0 requirements). Although, the setup procedures are exactly the same, transitioning between Mac and Windows was still time consuming.



Figure 13: openFrameworks project generator application.



Figure 14: openFrameworks project generator application: list of available add-ons.

Next, all scaffold (files) were automatically created and I could continue the project's setup inside the Xcode:

- The file **junikBubble.xcodeproj** was opened in Xcode (see [Figure 15](#))
- ofApps.cpp was changed to bubbleapp.cpp, and ofApps.h to bubbleapp.h
- The files softbody.cpp and softbody.h were added with the soft body physics code. The original code was developed by Dr Tomasz Bednarz (Queensland University of Technology, Brisbane, Australia) for his CAVE project collaborating with NIST in 2014.

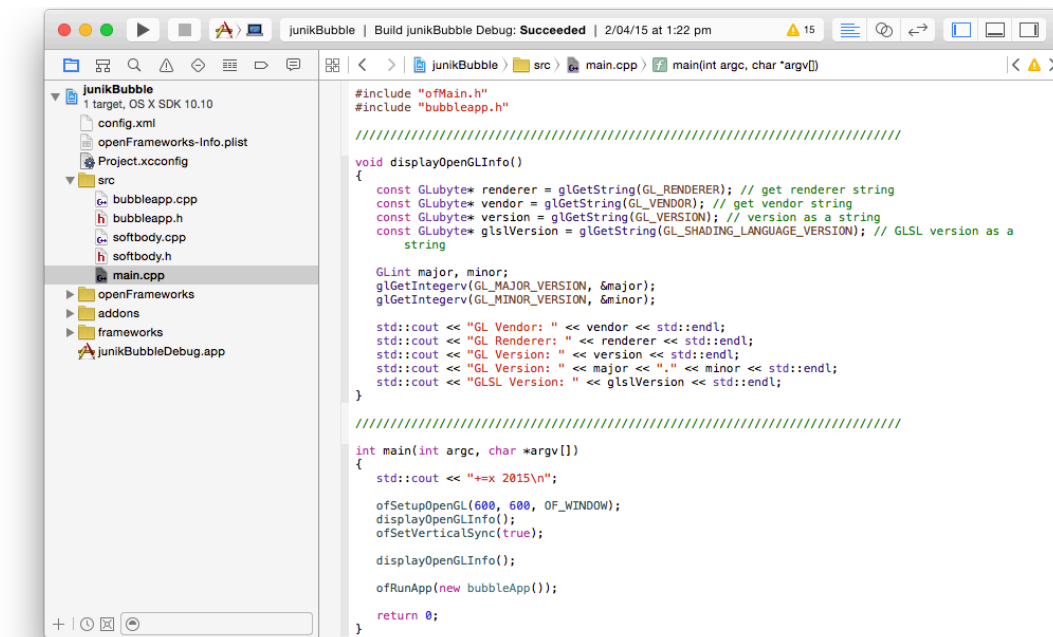


Figure 15: Xcode with open project junikBubble.

The source code of the project is encapsulated in the following files:

- **main.cpp**: main entry point to run the application, basic window size and OpenGL setup
- **softbody.cpp, softbody.h**: soft body physics C++ code
- **bubbleapp.cpp, bubbleapp.h**: main interactive code (parsing of configuration file, initialising and communication with Arduino, drawing, mouse and keyboard interactions, Kinect interactions, etc.)
- **bin/config.xml**: project configuration file (see Appendix A).

Code Structure

A diagram of the interactive application that was developed using openFrameworks (n.d.) is depicted in [Figure 16](#). As shown, the code structure is divided into a number of distinct blocks:

- Load CONFIG.XML

- Simulation – soft body physics
- Interactions through attached sensors
- Visualisation

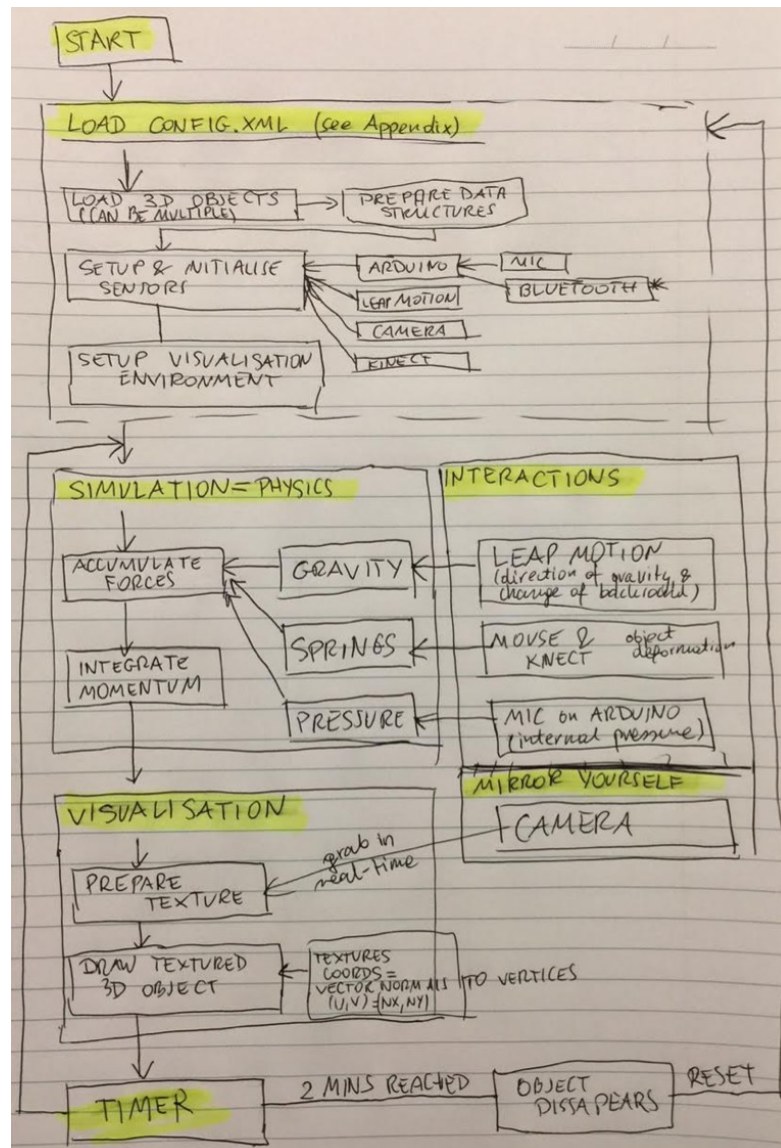


Figure 16: Diagram of the interactive application.

Configuration of the config.xml File

Configuration represents the initialisation part of the application. The central configuration settings are pre-defined in the file **bin/config.xml**, which defines the control parameters for the application, basic camera settings, music settings, as

well as a list of soft bodies and their physical parameters for numerical simulation and interactions. Descriptions of the functionality of all XML tags supported by the application are placed as comments in the **config.xml** file, which is listed in Appendix A.

The application loads 3D objects and prepares the required data structures for the physics simulations, allocates the required memory buffers and sets up the default parameters. Additionally, sensors such as those of the Kinect, Leap Motion, camera and Arduino connect to a computer and are initialised using pre-defined configuration settings and prepared for active utilisation. The visualisation environment is then initialised: window resolution or full screen is enabled and the bounding box is defined and used as 3D spatial boundaries for the physics simulation.

Algorithm

Each object imported by the main application was built from a set of vertices and faces in 3D space. Additionally, an object file can also consist of information about texture coordinates at each vertex and normal vectors (vectors perpendicular to the surface) to all objects' faces. The code initially prepares proper structures for computations by making copies of points in 3D and changing faces to a springs list. This connects vertices and calculates vectors perpendicular to the surface of the object (see [Figure 17](#); normal vectors represented in white colour).

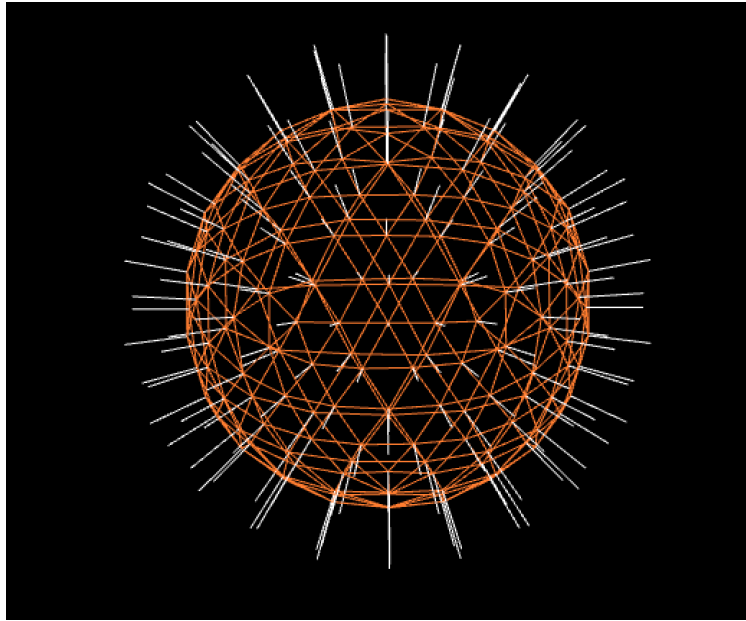


Figure 17: Sample 3D object (sphere) with normal vectors (white).

Once the data is prepared, the algorithm for one computational step is shaped as follows:

1. Calculate vectors perpendicular to the surface.
2. Accumulate forces:
 - a. Acceleration force
 - b. Pressure force
 - c. Spring force
3. Integrate momentum equation to get new positions
4. Update position based on the forces
5. Move to the next time-step

Simulation of a Balloon – Soft Body Physics

In order to simulate balloon-type behaviours on 3D objects, a soft body physics model code was implemented (by Dr Tomasz Bednarz), based on techniques

described in Matyka (2003). The vertices of imported meshes are represented as mass points governed by Newton's second law of motion, and the edges connecting these points as elastic links. The relative positions of the mass points and object deformations changed due to external forces, such as gravity, spring force and pressure. Parameters of the springs (stiffness and damping) were chosen according to experiments and could easily be modified in the configuration file `config.xml`.

After each step in the physics simulation, the 3D object is rendered using the texture image yielded by the camera attached to the PC. The texture coordinates used for the rendering of triangles (faces) are calculated in every frame. Normal vector to vertex (the sum of all normal surfaces surrounding the vertex divided by the number of surfaces surrounding the vertex) is used as $(U, V) = (NX, NY)$. See sample result in [Figure 18](#) and [Figure 19](#)). During the experiment stage, different shapes were tested such as a bunny and duck, which consist of smooth curves in terms of testing the texture rendering.



Figure 18: Textured 3D object with normal vectors used as texture coordinates. Duck.



Figure 19: Textured 3D object with vectors used as normal texture coordinates with a Stanford Bunny model.

Visualisation

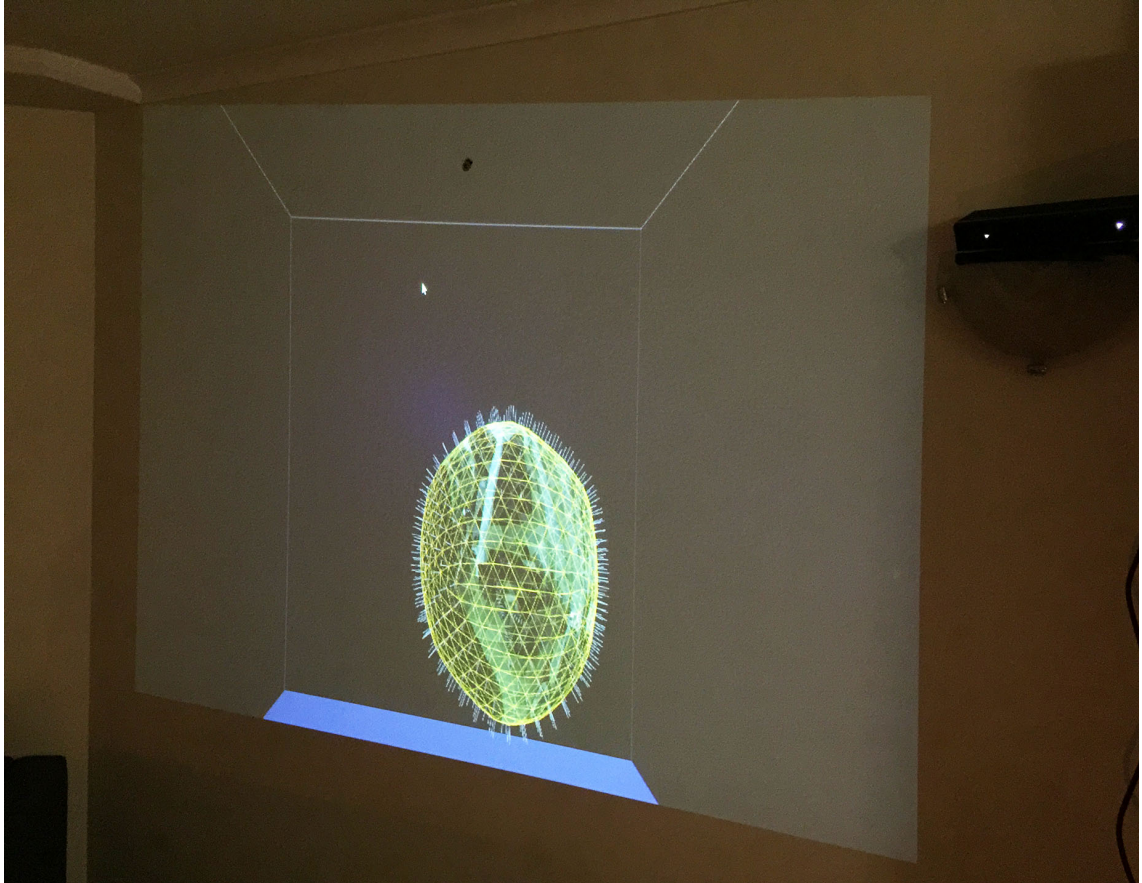


Figure 20: Initial stage of the sack with a bounding box, meshes and normal vectors displayed.

Figure 20 illustrates the production stage of *Indeed*. A bounding box within the screen prevents the sack from moving and growing beyond the edges of the bounding box. It also indicates the change in camera view. (See Figure 21)

The box is invisible in the final installation. (See Figure 22 and Figure 23)

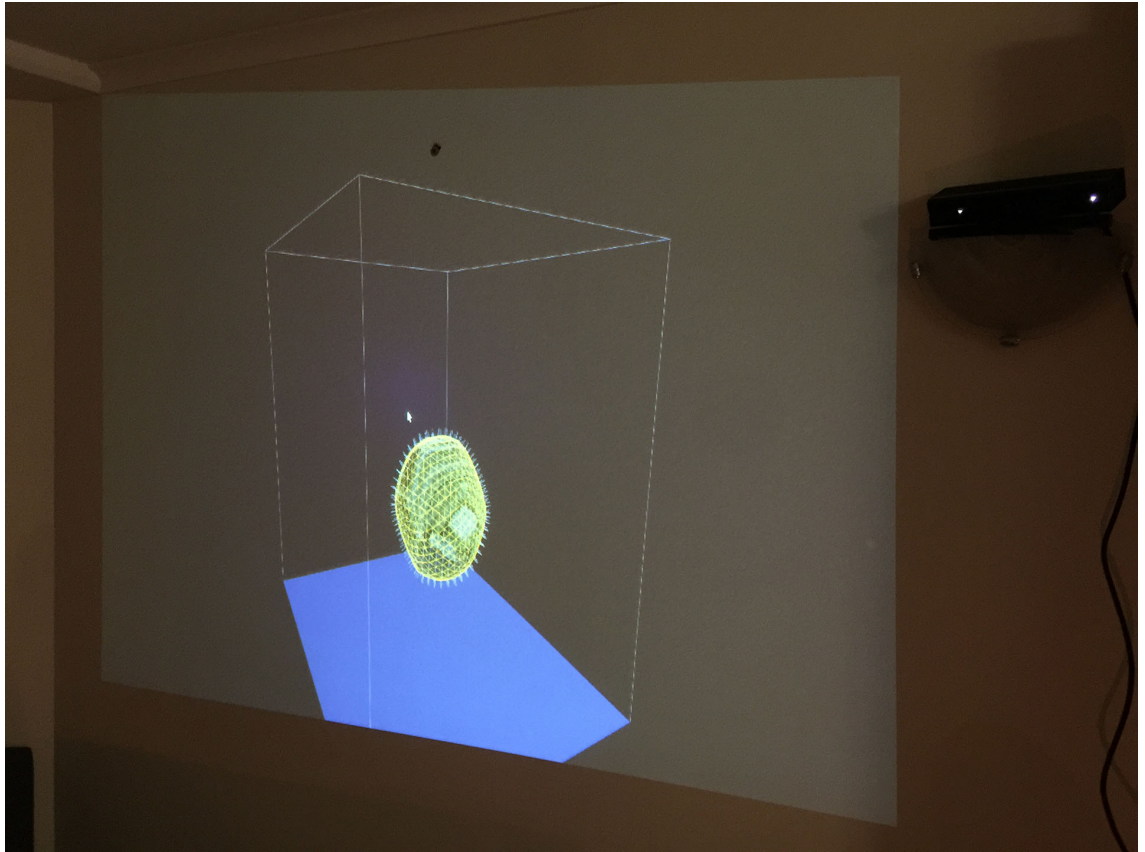


Figure 21: Initial stage of the sack from a different camera view.

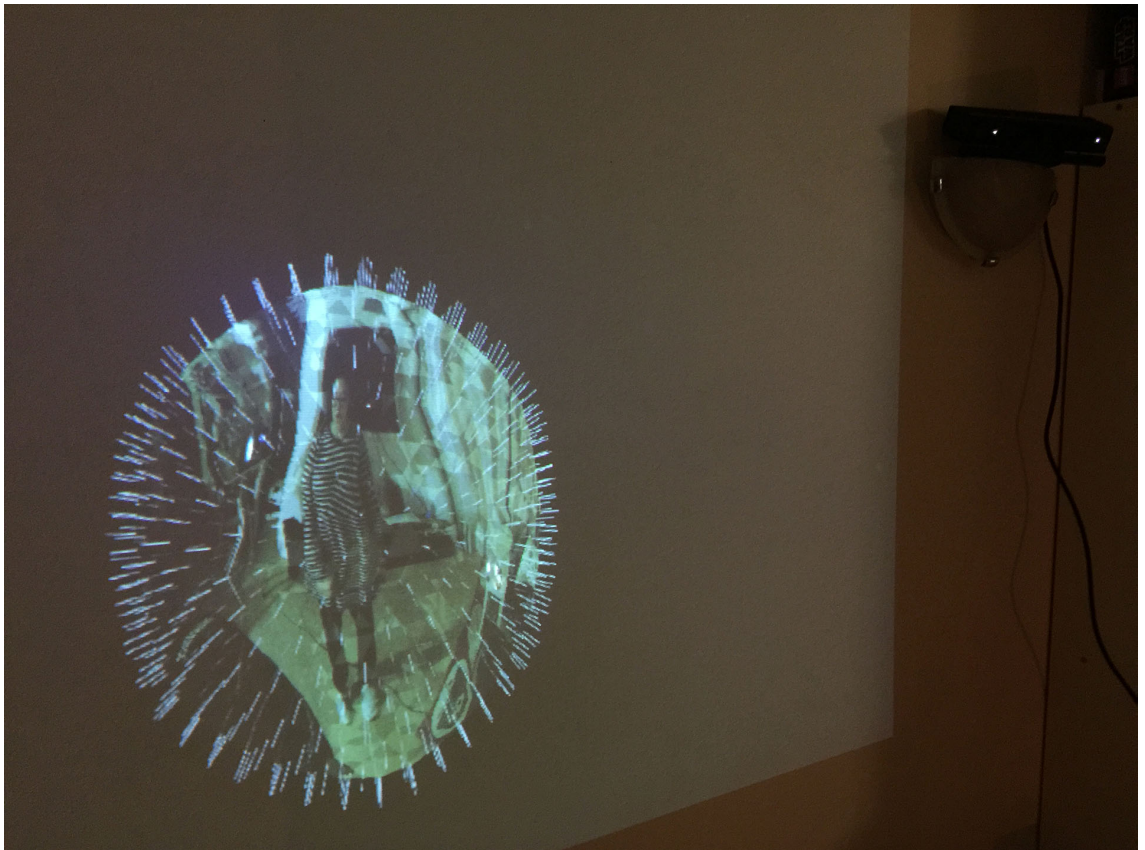


Figure 22: Mirroring (with vectors normal visible).



Figure 23: Mirroring (installation view).

During the conceptualisation, I modelled a sack similar to my initial drawing (Figure 1). This sack basically mirrors the view in front of it, which is a metaphor for a reflected altered self. I intend to focus on the notion that I am neither visible as just a body nor located in my body, but an undefined shape, which includes closed surroundings. In terms of clarifying the relationship between self and others, I keep the background black.

This sack is responsive to inputs of breath and hand gestures. The reflected image and a shape of sack shape will be constantly interacting accordingly.

Interaction methods

These interaction methods available to interact with *Indeed* are listed below:

MOUSE & KEYBOARD:

The method of development was to execute all desired fundamental interactions with a mouse and keyboard, then convert the interactions to my blower and Kinect. These mouse and keyboard interactions were implemented using a built-in open Framework events system.

Mouse Interaction	Keyboard Interaction
Moving (mouseMoved)	B/b: Blow in
Dragging (mouseDragged)	F/f: Toggle between full screen
Pressing (mousePressed)	P/p: Pause the simulation
Releasing (mouseReleased)	x/X: Gravity on, in -X/X direction
	y/Y: Gravity on, in -Y/Y direction
	z/Z: Gravity on, in -Z/Z direction
	C: Reset gravity to 0
	Space bar: Restart the object

When users press a mouse button, a **mousePressed** event is automatically initiated. The routine then searches for the closest point to the 3D object from the mouse position and adds a spring between these two locations, causing the physics engine to attract the object toward the mouse's location. The user can move the mouse (event **mouseDragged**) around and the physics engine follows its path. When the user releases the mouse button (event **mouseReleased** fired), the

created spring is released and removed from the iteration queue. Sample 3D object interactions are depicted with a Stanford bunny model in Figure 24.

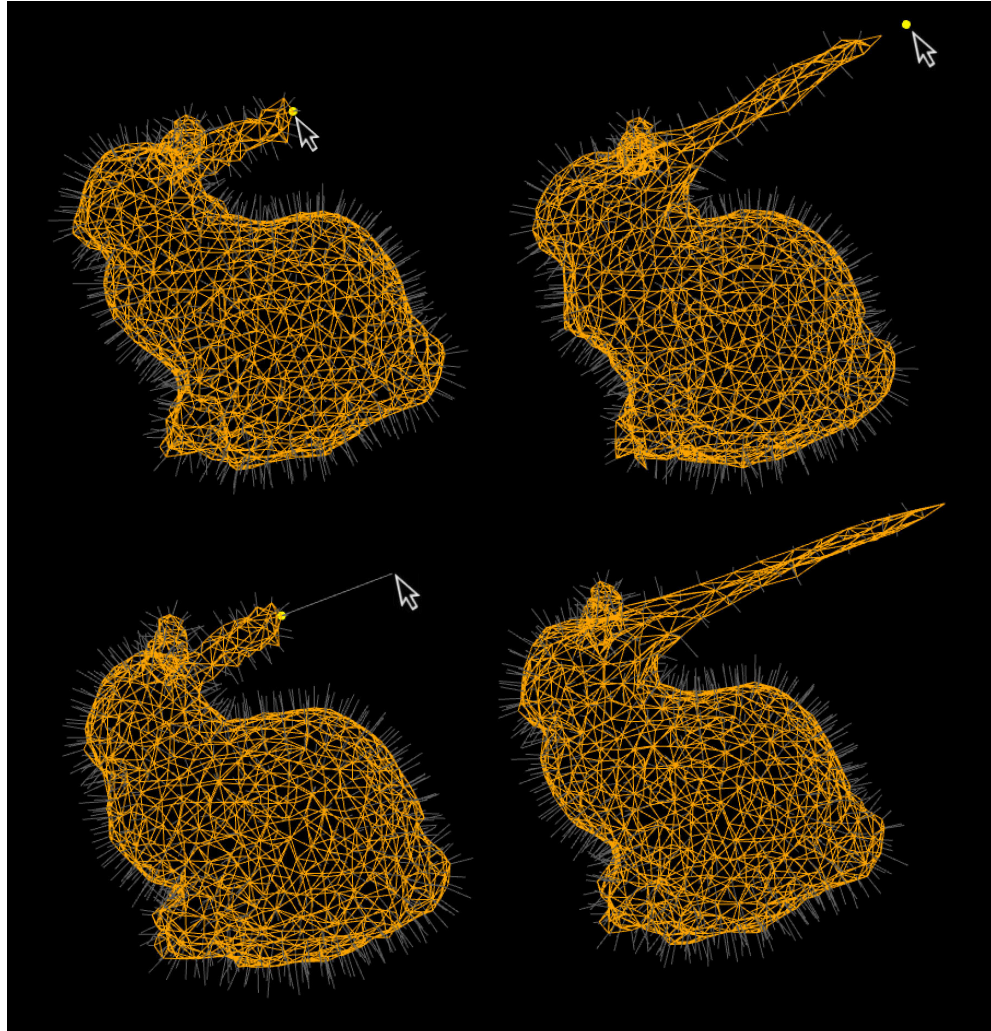


Figure 24: Interaction with mouse: grab & stretch using pressing, dragging and releasing functions

KINECT:

The mouse interactions needed to be converted to Kinect interactions. As a result of 'grab and stretch', deformation and movement occurred, as shown in Figure 24.

Interaction with mouse	Interaction with Kinect
Moving (mouseMoved)	Hand / arm move with an open hand
Pressing (mousePressed)	Closing hand
Dragging (mouseDragged)	Hand / arm move with a closing hand
Releasing (mouseReleased)	After dragging, open hand

Figure 25: Conversion of interacting functionalities between mouse and kinect.

BLOWER:

A microphone connected via the Arduino board controls the internal pressure force of the sack. When a viewer blows air on the sensor, the peak signal is detected and causes a linear increase of the pressure variable, which simulates more air being pumped into the virtual object.

Set up for blowing machine – prototype stage

[Figure 26](#), and [Figure 27](#), display the initial wireless setup that included a 9V battery, Arduino board, Bluetooth Shield, microphone sensor and a power switch. All the elements fit neatly into the small container, with only the microphone sensor placed outside of the box.

The setup can be switched ON and OFF. When paired with the computer once the application starts, the setup is automatically connected using openFrameworks serial port communication functions. The system works well under both Mac OS X and Microsoft Windows 8.1 environments. Once the setup was tested, I placed it in to a Megaphone shape, as shown in [Figure 29](#), and [Figure 30](#).

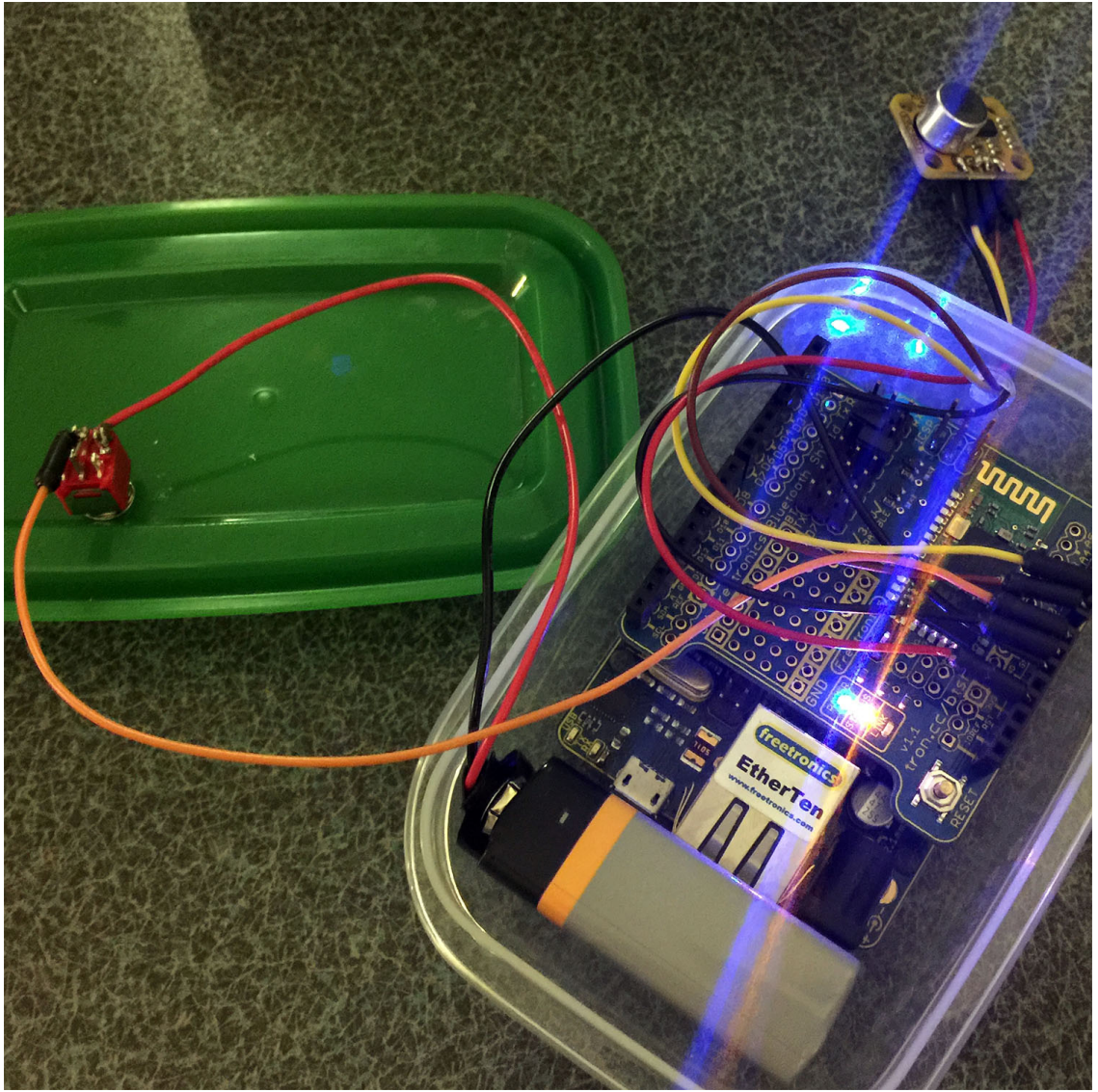


Figure 26: Wireless setup includes: 9V battery, Arduino board, and Bluetooth Shield, microphone and power switch.

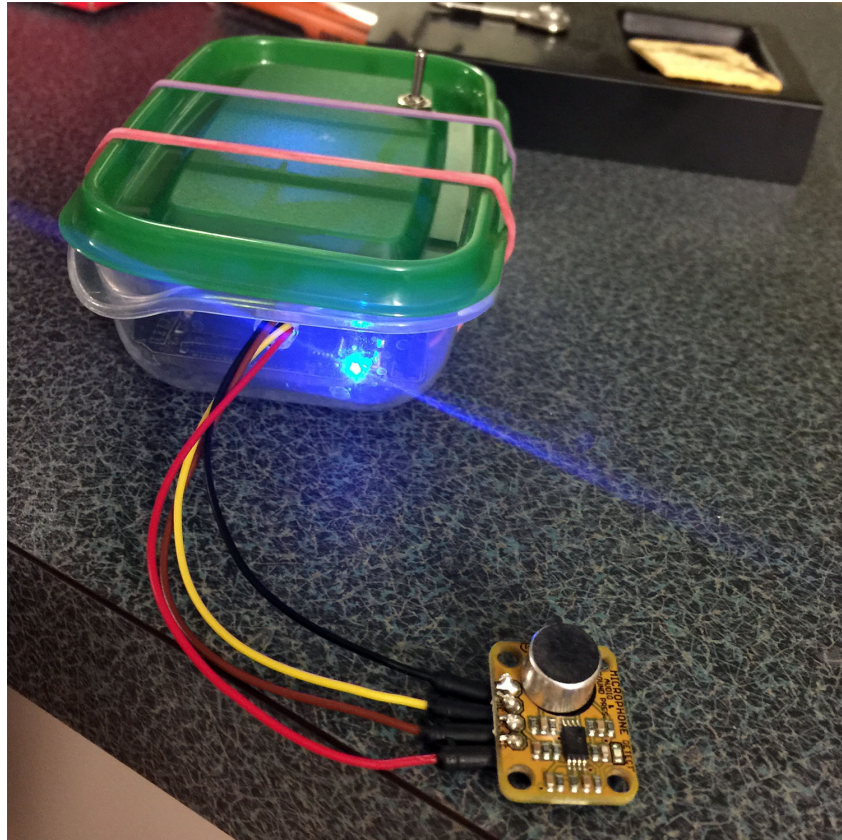


Figure 27: Wireless setup.



Figure 28: Soldering the power switch.



Figure 29: Prototype Megaphone setup 1.

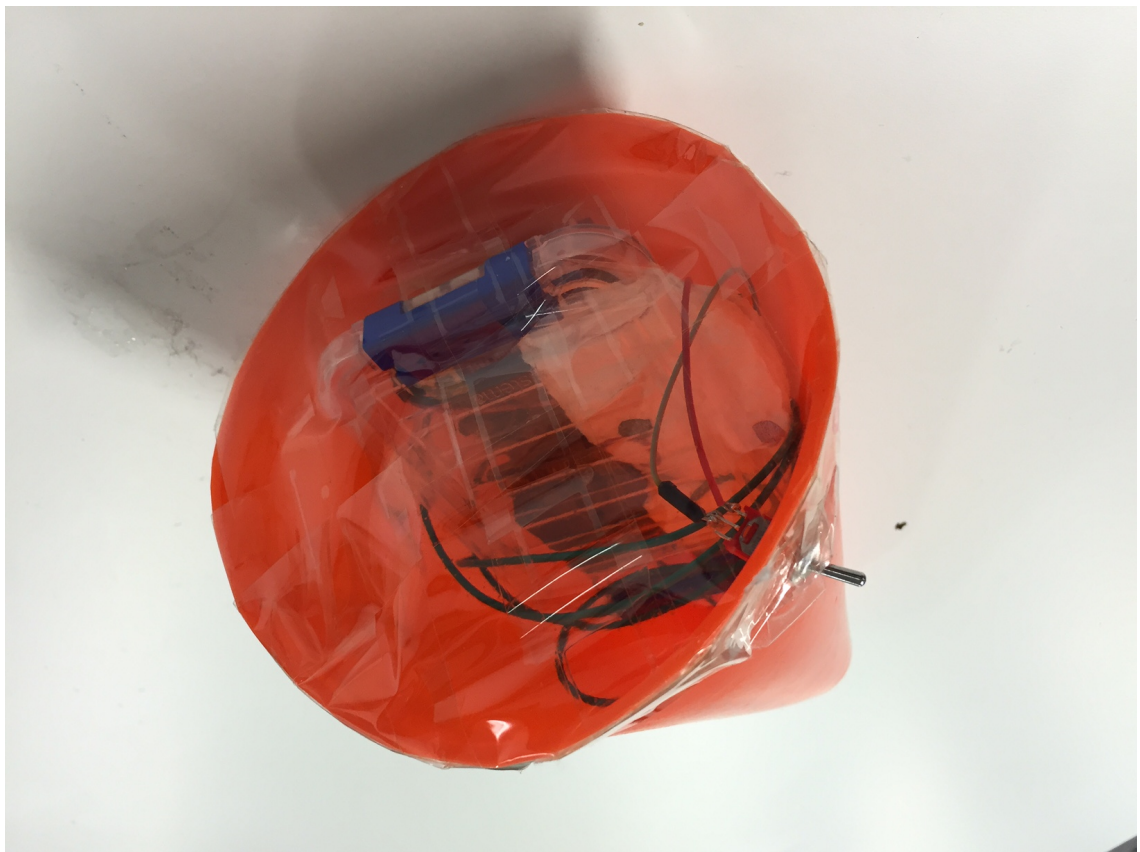


Figure 30: Prototype Megaphone setup 2.

Interaction with Breath



Figure 31: Default state of a sack.

[Figure 31](#), and Figure 32 illustrates an experiment to interact with a sack by blowing breath into the blower. This was also tested with the prototype megaphone. Blowing instantly pumped air to the sack and the reflected image was also distorted. However, as time passed, the air leaked out and shrank.

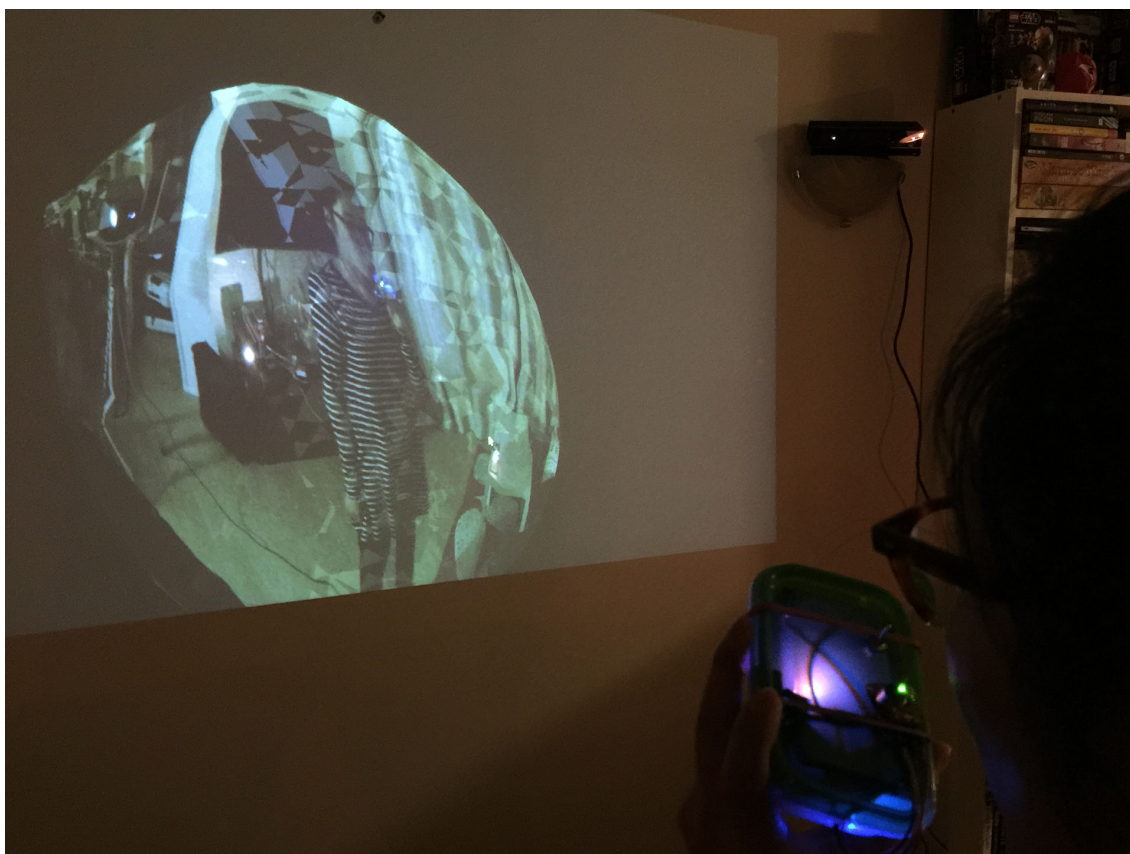


Figure 32: Blowing breath into the sack 1.



Figure 33: Blowing breath into the sack 2.

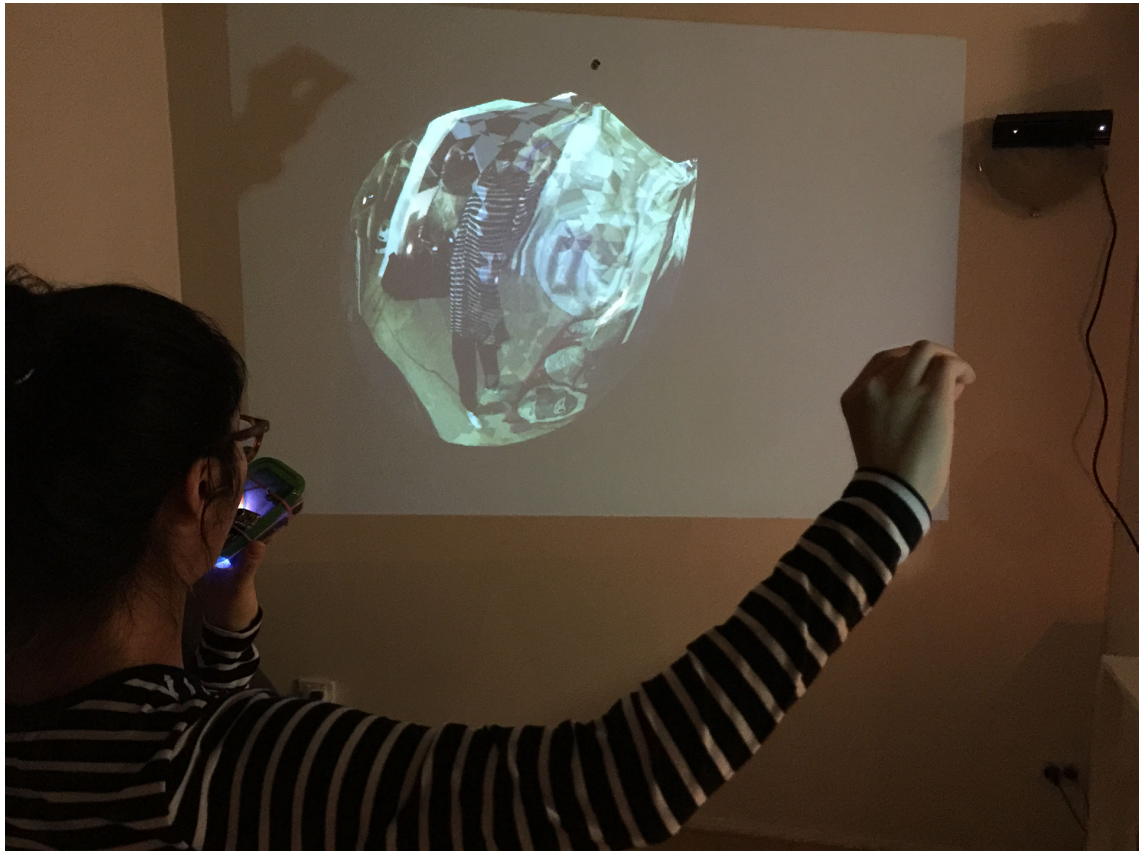


Figure 34: Blowing breath into the sack and interacting with a hand gesture.

Figure 34 shows the possibility of concurrent interactions with breath input and hand gestures. In fact, up to 6 people can influence this project at once since Kinect 2 detects up to 6 users.

Interaction with Hand Gestures

Both right and left hands can be used in any direction to interact. The SDK methods allow for estimating the states of both hands, whether open, closed, lasso or unknown. Figure 25 describes the types of hand gestures. When the hand is open, nothing happens. When the hand is closed and moves, the interaction is equivalent to dragging the object around using a mouse. [Figure 35](#), and [Figure 36](#), illustrate my tests and calibrations of the system.



Figure 35: Testing interactions with Kinect.

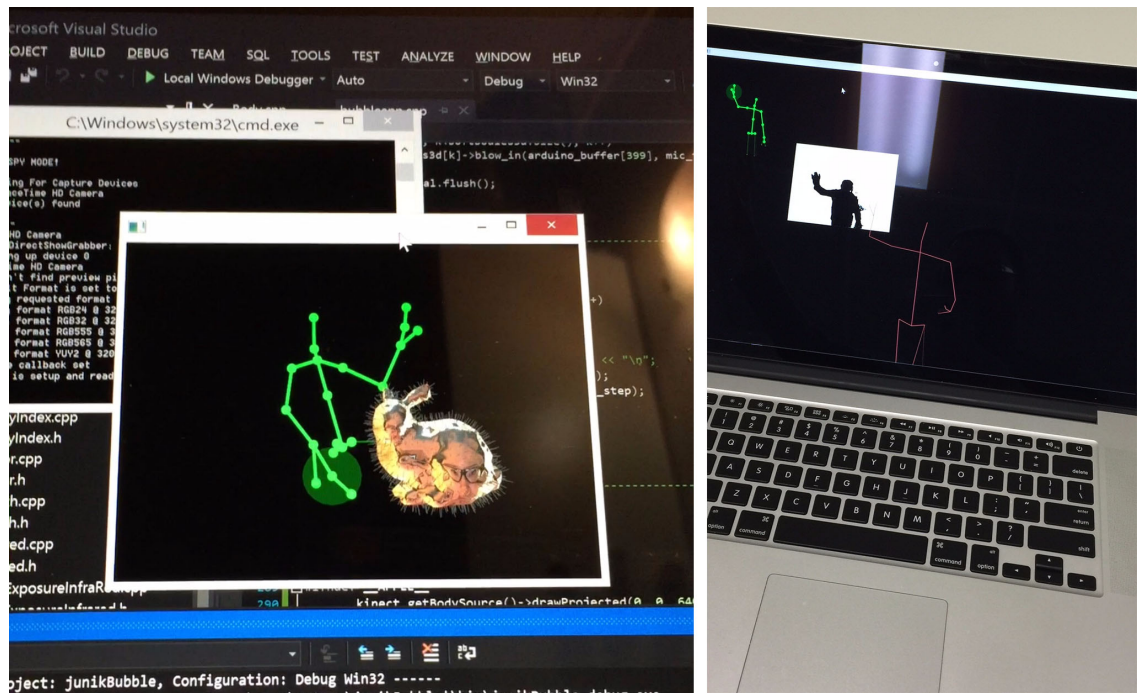


Figure 36: Gestures calibrations (through Kinect SDK 2).

In terms of experimenting and investigating the hand gestures accurately, the following tests have done with the skeleton view turned on. Figure 37 and Figure 38 indicates the option to interact with both the right and left hands. Figure 39 and Figure 40 display the flow of deformation of the sack and hand gestures in varied directions are shown on the next two pages.

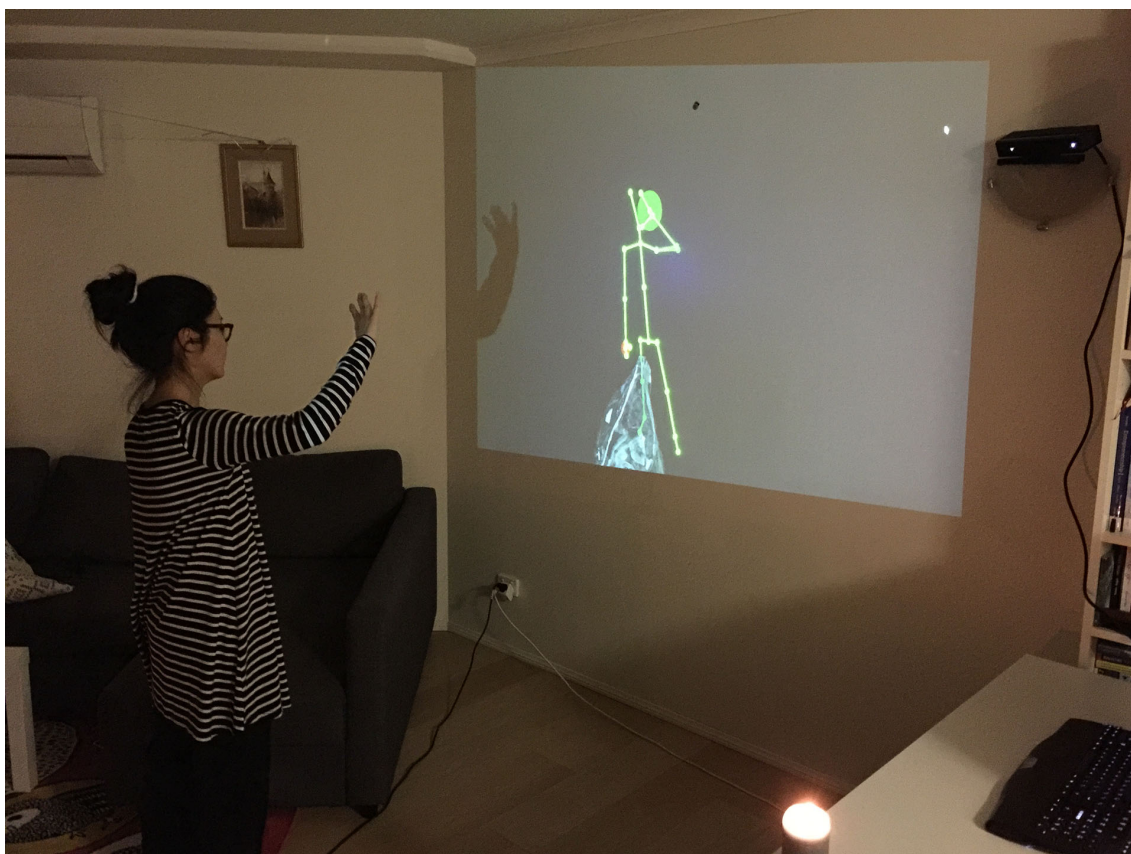


Figure 37: Interaction with the sack (right hand).



Figure 38: Interaction with the sack (left hand).

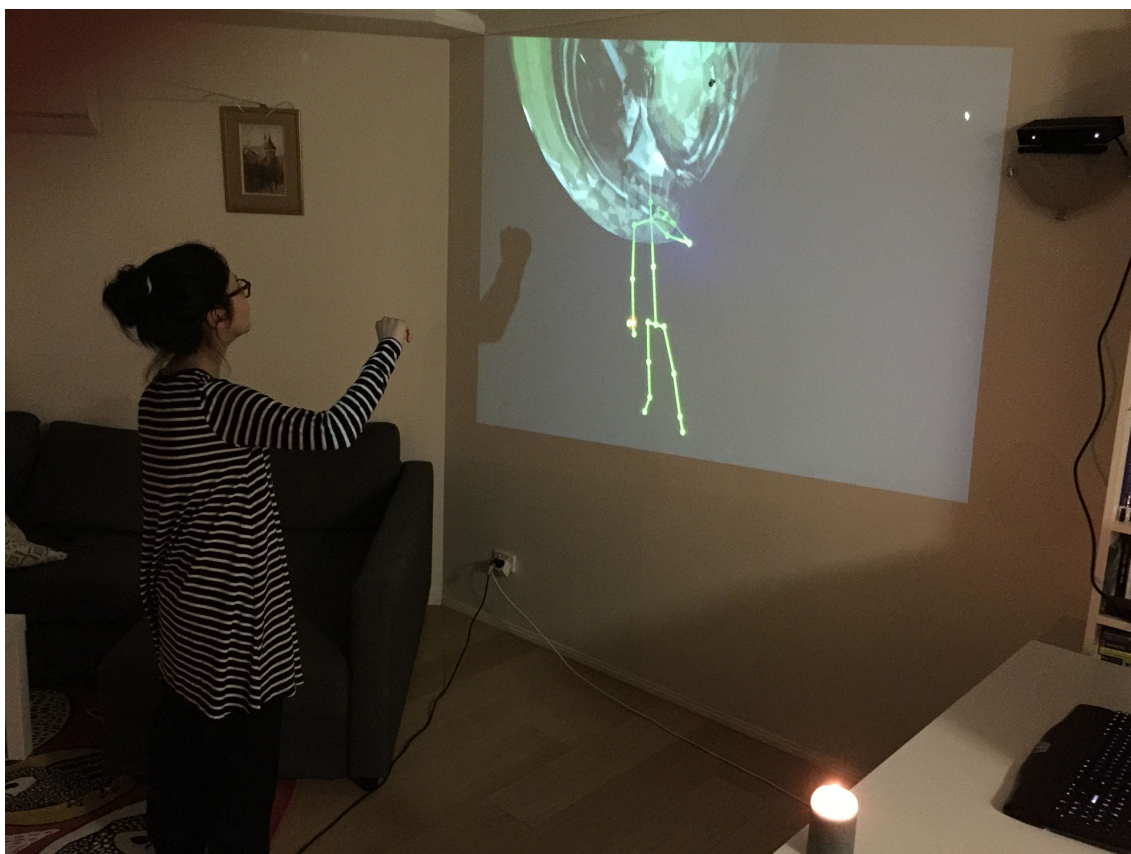


Figure 39: Interaction with the sack 1.



Figure 40: Interaction with the sack 2.

In *Indeed*, multiple interactions with multiple allow people to create something together. Figure 41 to Figure 45 demonstrate that *Indeed* is able to detect multiple people (up to six people) and interact with hand gestures and breath input from multiple people.

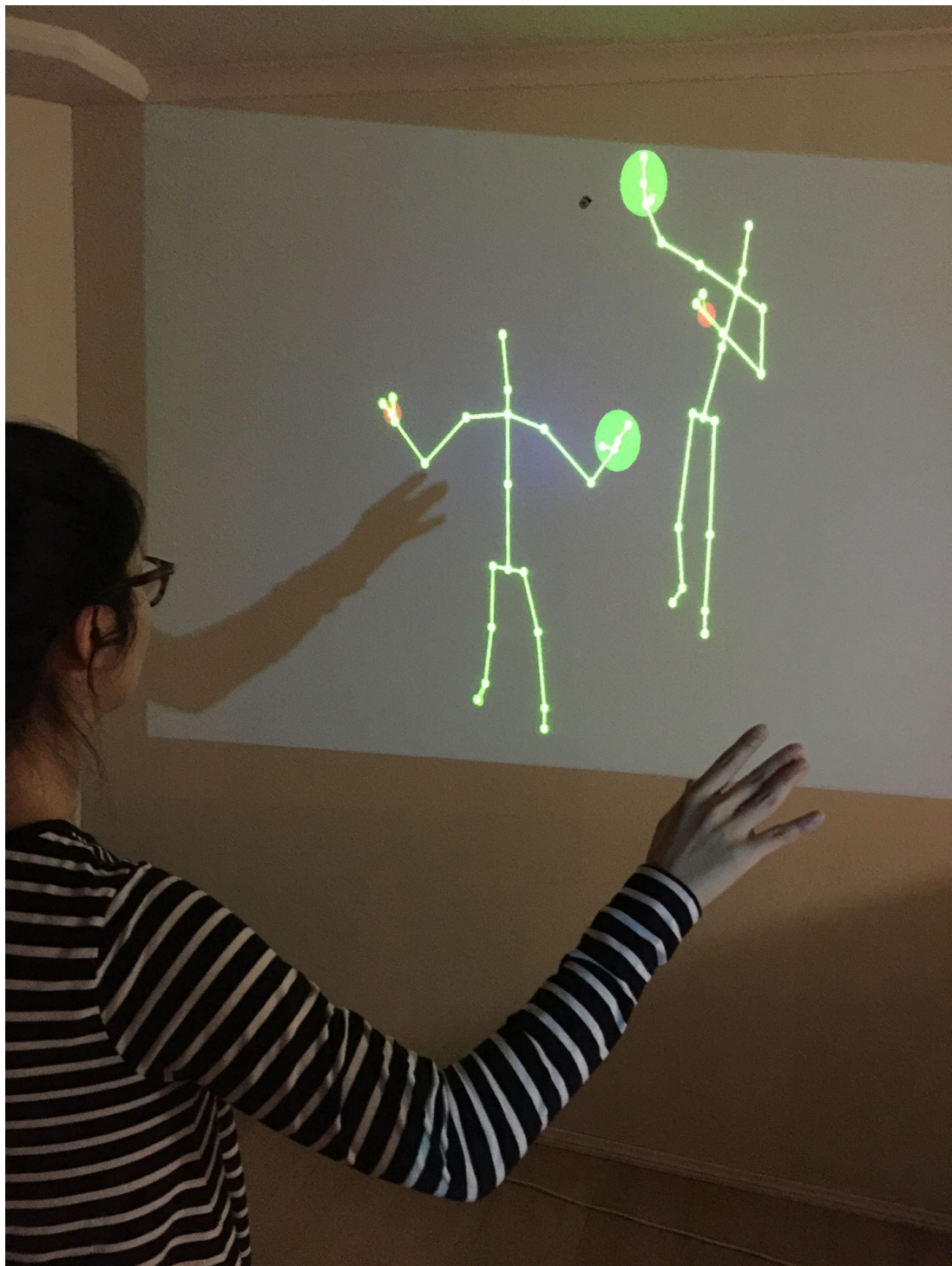


Figure 41: Kinect detecting two people 1.

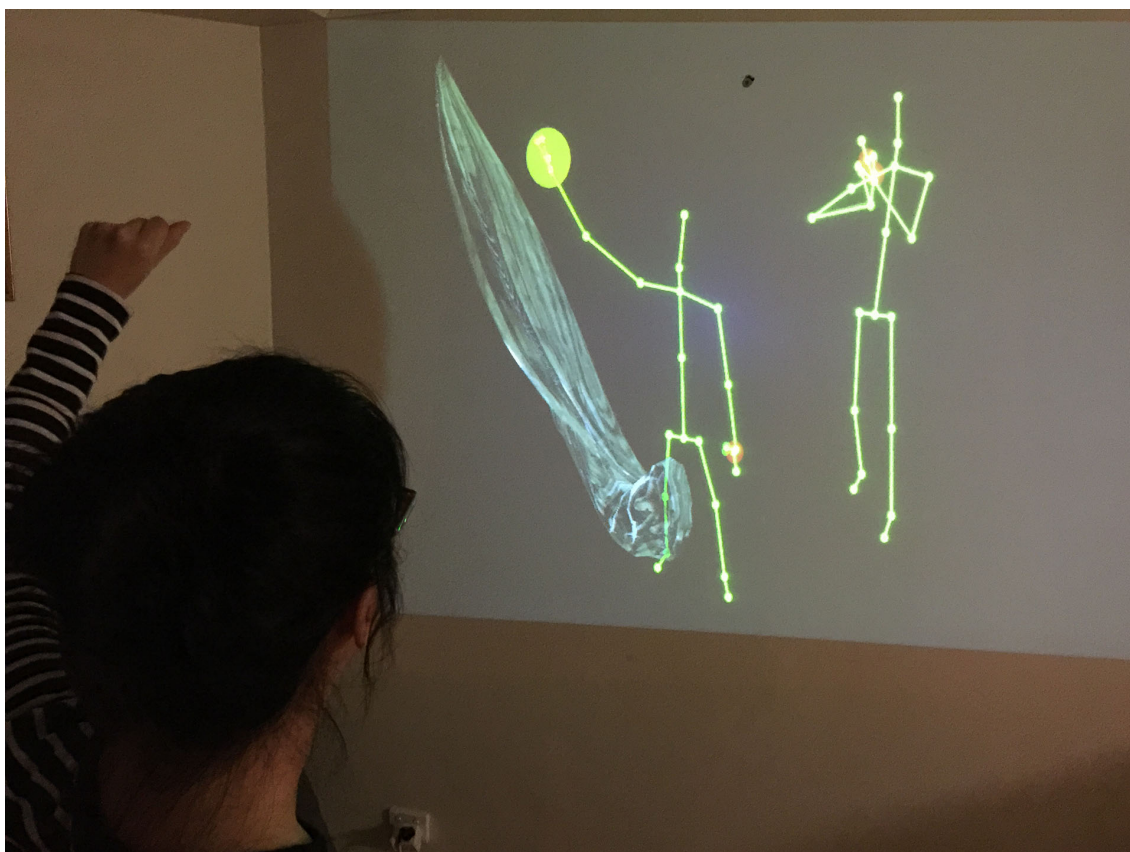


Figure 42: Kinect detecting two people 2.

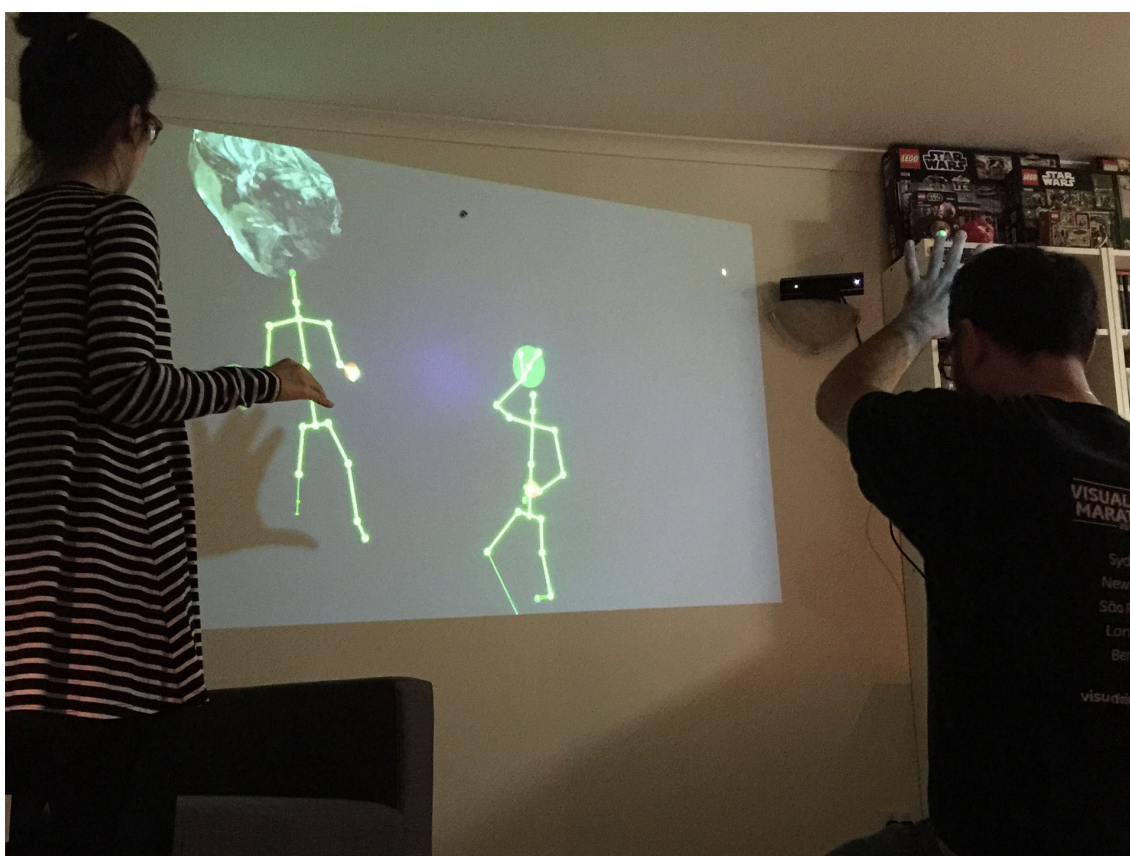


Figure 43: Two people influencing the sack 1.

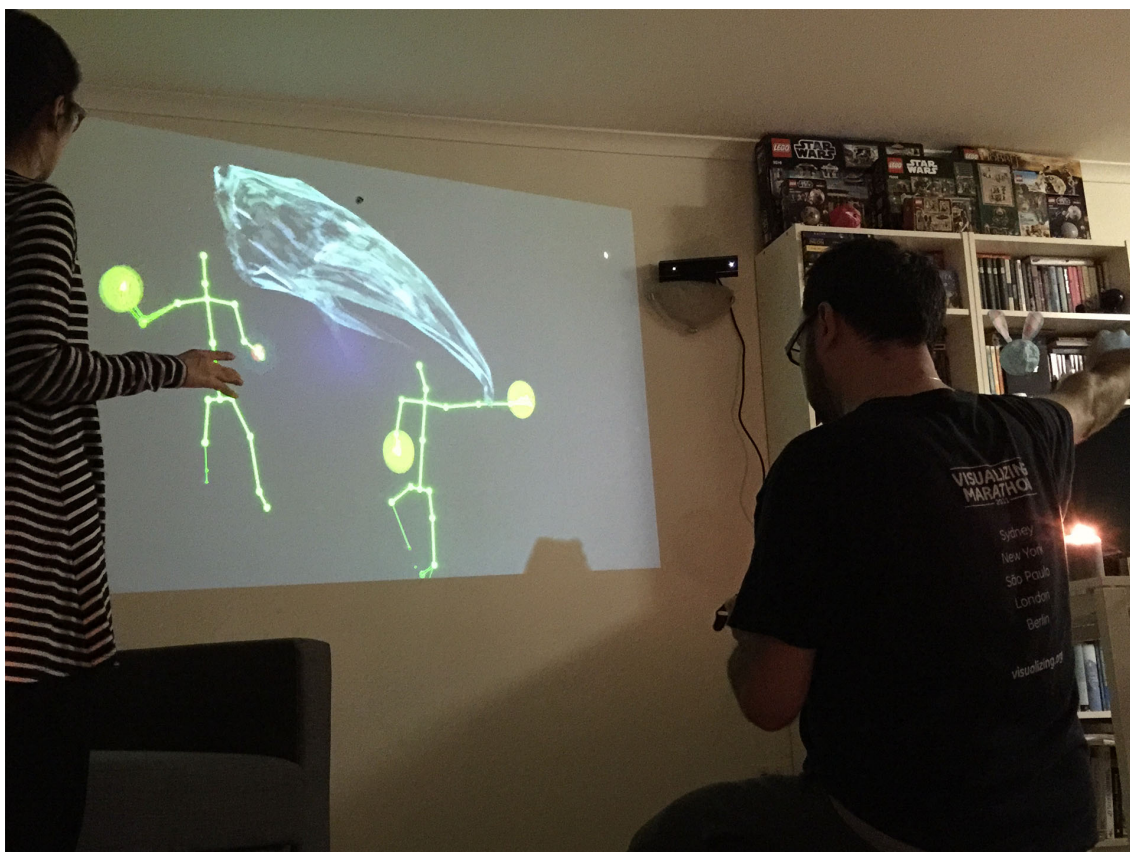


Figure 44: Two people influencing the sack 2.

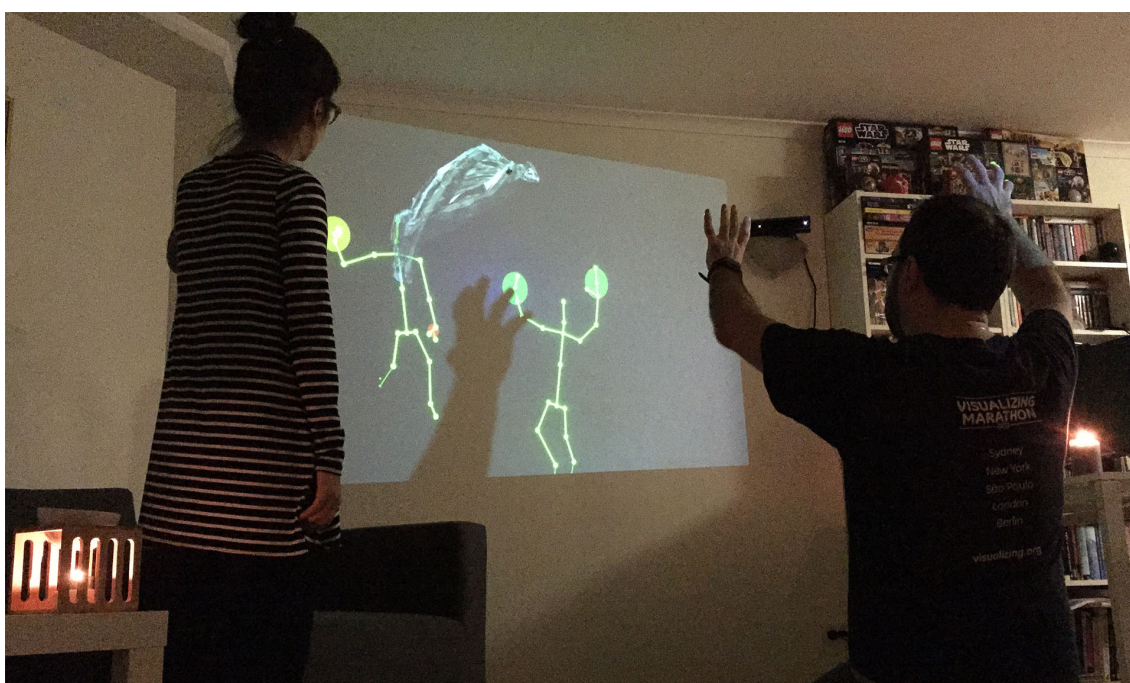


Figure 45: Two people influencing the sack 3.

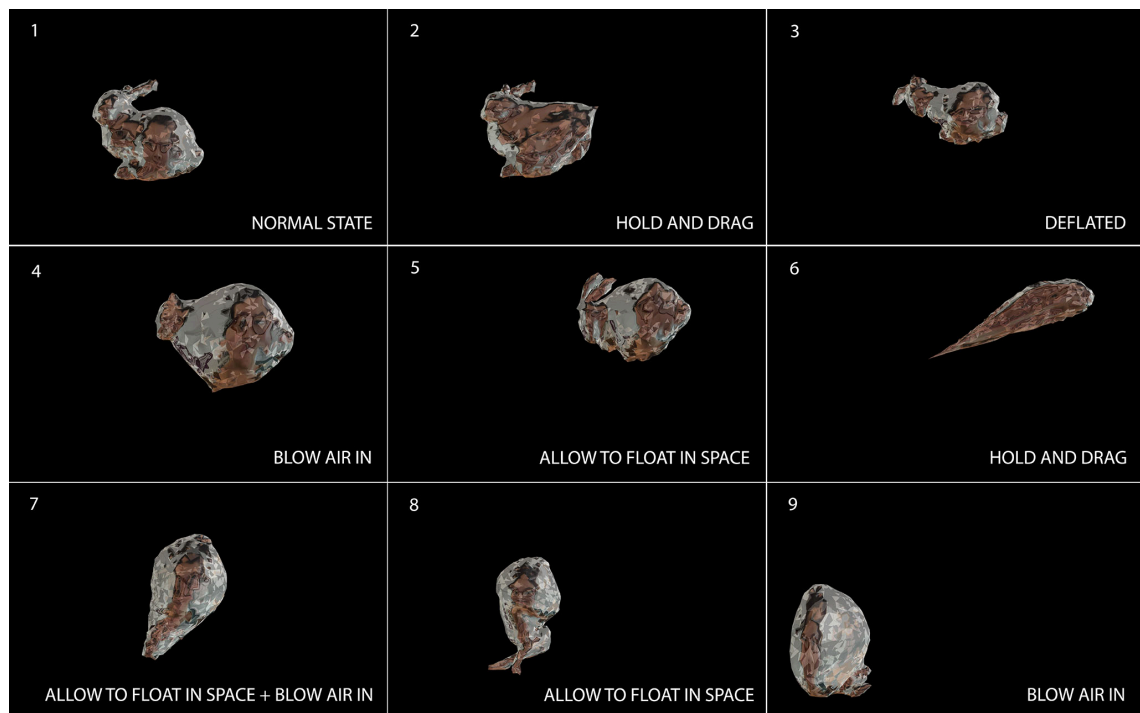


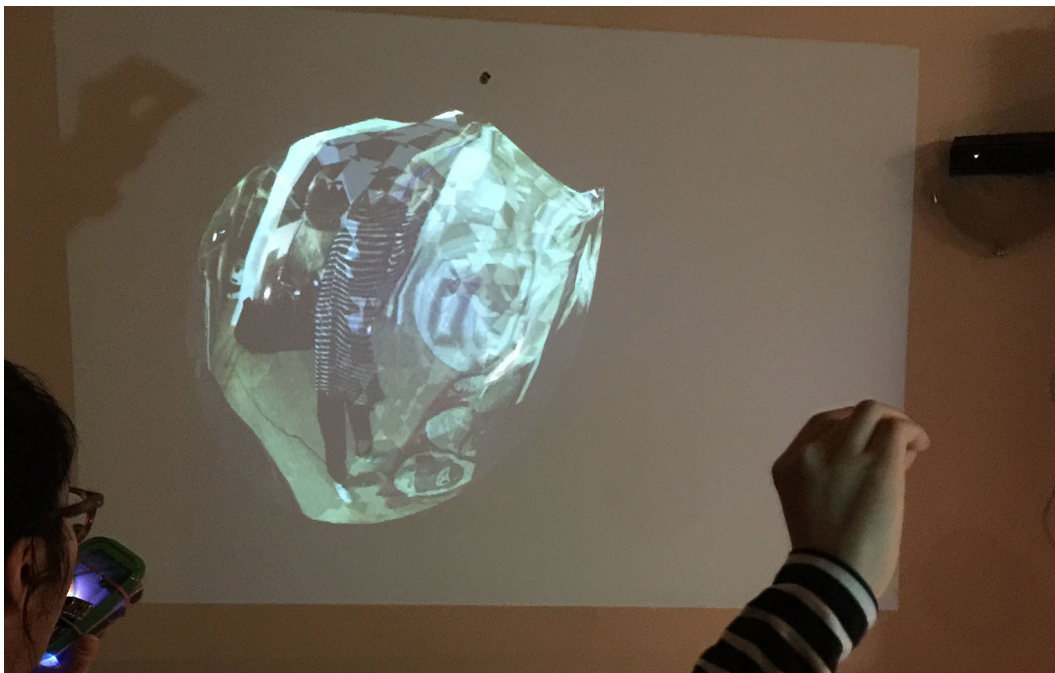
Figure 46: Interaction mode.

Interaction with Sound

Once the visualisation portion was developed to its current state, I experimented with some openFrameworks SDK examples, recording and playing sounds and music. In order to deepen users' sense of immersion, I used the `ofSoundStream` class to capture the sound coming from the microphone and played it back through the speakers. This created weird spatial sound feedback, and there is an option in `config.xml` to switch this on and off. Additionally, I built in the capability to play MP3 files in the background in order to embody my art installation even more so. I also added an option to play short sounds when a user grabs the sack, and releases as well as during the gesture of grabbing and dragging.

3

CLOSING



I enter a space. As soon as I enter, one floating sack appears on the wall. This sack mirrors me and my surroundings. I see that it floats and shrinks gently. According to its movement, I am changing. I am constantly being reshaped

and deformed, mingling with the surroundings and facets of this sack. I observe all these subtle changes that this sack mirrors to me.

When I close my eyes and ears, I *listen and feel myself*. When I open my eyes and ears, I play *hide and seek* with myself by mirroring through many different beings and things. I *search pieces and facets of myself*.

This floating and mirroring sack implies the maneuvers of reflecting myself inwards and outwards. Thus, this becomes a reflected self. Rather than focusing on spotting pieces of my body, I acquire this sack, which is constant changing, and deforming images of me, an altered self. It is blurry but unmediated.

I now stand in front of a blower. I breathe out slowly, like whispering to myself. The sack becomes bigger. It reminds me of how those mirrors made me look chubbier. I stop breathing in. The sack is slowly slimming down and floating. It reminds me that some mirrors lied to me made me look prettier.

I reach my arm out, open my hand and grab in air. I slowly move my grabbed hand. A point is grabbed and dragged. I open and release my hand slowly. Now the body of the sack is moving toward my hand point moved to. I try this movement with my other hand. I try this movement in different directions as well. I see a man walks into this room. He does what I do. I see that the body of the sack is moving toward the point he points.

This bombastic sized blower looks like a cross between a megaphone and a wind instrument. It is a medium to embrace my influence – to create an altered self.

Through designing the blower, I focused on formulating the meaning of breath in this installation as 'desire'.

Different mirrors show me differently. My sack exaggerates this phenomenon by moving and shaping constantly. However, I perceive that my desires and any actions regardless of desires are significant factors that change me. Besides, influences from others are also signals that change me. This is me, indeed, in this true moment. I developed this project *Indeed* with a desire to be able to see me.

In order to articulate the experience of changing myself every moment, I sought an interactive art form with concepts of process and embodiment within the form. Through developing and experimenting with the interactive process, I encountered myself in situations.

The aesthetic of interactive art that I realised in creating this project is the ability of artists and audiences to share an intimate process of developing personal narratives. I deliberately described the interactive process in present tense here.

For instance, a single painting may be composed of many layers of lines and colours. An artist has reasons for choosing certain colours and brush strokes etc. However, these individual choices are often layered over by later brush touches and are buried in the final painting. Hence, this process becomes an artist's very personal memory. On the other hand, in interactive art, it is understood that interactivities are an artist's reasons and meanings. Although the technologies utilised are hidden, the entire process of meaning making is tied into different forms of interaction.

In interactive art, neither past tense nor hidden processes exist.

BIBLIOGRAPHY

- Arduino. (n.d.). *Arduino Wikipedia*. Retrieved March 30, 2014 from <http://en.wikipedia.org/wiki/Arduino>
- McCarthy, A. (2000). *Ambient Television*. Durham: NC. Duke University Press.
- Applied Philosophy: Philosophy of Technology 1*. (n.d.). Retrieved March 20, 2015 from http://www.slideshare.net/DMLab/aheidegger?next_slideshow=1
- Auslander, P. (1999). *Liveness: Performance in a Mediated Culture*. New York, NY: Routledge.
- BeAnotherLab. (n.d.). *Gender Swap - Experiment with the Machine to Be Another*. Retrieved June 14, 2015, from <https://player.vimeo.com/video/84150219>
- Blum, J. (2013). *Exploring Arduino: Tools and Techniques for Engineering Wizardry*. New York, NY: John Wiley & Sons, Inc.
- Brey, P. (2000). *Technology and embodiment in Ihde and Merleau-Ponty*. Metaphysics, Epistemology and Technology. Research in Philosophy and Technology, 2000(19). Ed. C. Mitcham. London: Elsevier/JAI Press.
- Chyaka, K. (n.d.). *How the open source disrupting visual art?* Retrieved March 30, 2014, from <http://thecreatorsproject.vice.com/blog/how-open-source-is-disrupting-visual-art>
- Dixon, S. (2007). *Digital performance: A History of New Media in Theater, Dance, Performance Art, and Installation*. Cambridge, MA: The MIT Press.
- Fallman, D. (2008). *The interaction design research: triangle of design of Practice, design studies, and design exploration*. Design issues, 24(3), 4-18. doi: 10.1162/desi.2008.24.3.4
- Feenberg, A. (2003). *Active and Passive Bodies*. Techne: Research in Philosophy and Technology, 7(2), 125-130. doi:10.5840/techne2003725
- Fifield, G., Ketner II, J.D., & Donath, J.S. (2008). *Act/React: Interactive Installation Art*. Milwaukee, WI: Milwaukee Art Museum.
- Findeli, A. (2010). *Searching for design research questions: Some Conceptual Clarifications*. Chow, R., Jonas, W., & Joost, G. (Eds.). Questions, Hypotheses & Conjectures: Discussions on Projects by Early Stage and Senior Design Researchers. New York - Bloomington, NY: iUniverse, Inc. Retrieved February 15, 2015, from http://projekt.unimes.fr/files/2014/04/Findeli.2010.DRN_.Design_research_questions.pdf
- Fretronics. (n.d.). Fretronics: Arduino-Compatible Electronics Kits & Parts. Retrieved January 15, 2015 from <http://www.fretronics.com.au/>

- Gold, R. (2007). *The Plenitude: Creativity, Innovation, and Making Stuff (Simplicity: Design, Technology, Business, Life)*. Cambridge, MA: The MIT Press.
- Gough, P., De Berginy Wall, C., & Bednarz, T. (2014). *Affective and Effective Visualisation: Communicating Science to Non-expert Users*. 2014 IEEE Pacific Visualization Symposium (Pacific Vis), 335-339.
doi:10.1109/PacificVis.2014.39
- Grau, O. (2003). *Virtual Art: From Illusion to Immersion*. Cambridge, MA: The MIT Press.
- Kairalla, C. (n.d.). Chris3000, Portfolio site for Chris Kairalla. Retrieved January 26, 2015 from <http://www.chris3000.com/>
- Karvinen, K. & Karvinen, T. (2014). *Make: Getting Started with Sensors: Measure the World with Electronics, Arduino, and Raspberry Pi*. Sebastopol, CA: Maker Media, Inc.
- Kim, S.Y., Prestopnik, N., & Biocca, F.A. (2014). *Body in the interactive game: How interface embodiment affects physical activity and health behaviour change*. Computers in Human Behavior, 2014(36), 376-384.
doi:10.1016/j.chb.2014.03.067
- Kwastek, K. (2013). *Aesthetics of Interaction in Digital Art*. Cambridge, MA: The MIT Press.
- Levin, G. (2014). *New-Media Master Class with Golan Levin*. Retrieved January 15, 2015 from <https://colab.aut.ac.nz/workshops/new-media-master-class-with-golan-levin/>
- Lopes, D.M. (2010). *A Philosophy of Computer Art*. New York, NY: Routledge.
- Manovich, L. (2002). *The Language of New Media*. New York, NY: Leonardo.
- Matyka, M., & Ollila, M. (2003). A Pressure Model for Soft Body Simulation. *Proceedings of Annual SIGRAD Conference*. Umeå, Sweden.
- Maurice Merleau-Ponty's approach to consciousness*. (n.d.). Retrieved March 20, 2015, from <http://www.slideshare.net/DMLab/maurice-merleau-pontys-approach-to-consciousness>
- McCarthy, A. (2001). *Ambient Television: Visual Culture and Public Space*. Durham, NC: Duke University Press.
- Michelis, G. (2013). Third International Conference on Arts and Technology, ArtsIT 2013, Milan, Italy, March 21-23, 2013. Selected papers, retrieved February 15, 2015 from <http://artsit.org/2013/show/home>
- Microsoft. (n.d.). *Kinect for Windows*. Retrieved June 1, 2015 from <https://www.microsoft.com/en-us/kinectforwindows>

Moustakas, C. (1990). *Heuristic Research: Design, Methodology, and Applications*. London, UK: SAGE Publications, Inc.

OpenFrameworks. (n.d.). Retrieved January 10, 2015 from <http://openframeworks.cc>

Perevalov, D. (2013). *Mastering openFrameworks: Creative Coding Demystified*. Birmingham, UK: Packt Publishing.

Philosophy of Technology 2: Embodiment, Disembodiment and Immersion. (n.d.). Retrieved June 14, 2015 from <http://www.slideshare.net/DMLab/applied-philosophy-2-embodiment-immersion>

Pritika, L. (2013). *Transvirt: Doing the digital body*. Retrieved May 14, 2015 from <http://aut.researchgateway.ac.nz/handle/10292/5537>

RelaxLine. (2010). *BellyBio Interactive Breathing*. Retrieved April 8, 2015, from <https://itunes.apple.com/us/app/bellybio-interactive-breathing/id353763955?mt=8>

Rush, M. (2001). *Video Art*. London, UK: Thames & Hudson.

Sims, R. (n.d.). *Interactivity: A Forgotten Art?* Retrieved June 14, 2015, from <http://www2.gsu.edu/~wwwitr/docs/interact/>

Somniacs – Beyond Reality. (n.d.). *Birdly*. Retrieved March 8, 2015 from <http://birdly.zhdk.ch>

Stern, N. (2013). *Interactive Art and Embodiment: The Implicit Body as Performance*. Canterbury, UK: Gylphi Limited.

Viljoen, M. (2010). Embodiment and the experience of built space: the contributions of Merleau-Ponty and Don Ihde. *South African Journal of Philosophy*, 29(3), 306-329. Retrieved February 15, 2015 from <http://www.ajol.info/index.php/sajpem/article/view/59153>

What is Arduino? (n.d.). Retrieved July 19, 2014 from <http://www.arduino.cc>

What is Somaesthetics? (2013, November 25). Retrieved May 1, 2014 from <https://embodimentblog.wordpress.com/2013/11/25/what-is-somaesthetics/>

APPENDIX A

This is content of the configuration file **config.xml** that accompanies interactive application. Descriptive comments are marked in green.

```
<configuration>

  <!--
  Tag "is_fullscreen" has two settings:
  - true: open application in fullscreen mode,
  - false: open application in window mode.

  Window mode = useful for testing. Fullscreen mode = useful for installation.
  -->
  <is_fullscreen>>false</is_fullscreen>

  <!--
  Tag "is_help_displayed" has two settings:
  - true: help is displayed on the screen when application starts,
  - false: help is disabled initially.
  -->
  <is_help_displayed>>false</is_help_displayed>

  <!--
  ARDUINO sensor:
  is_arduino_connected: switch on/off querying/connecting Arduino,
  is_arduino_input_displayed: if true, it displays sensor reading plot.
  -->
  <is_arduino_connected>>false</is_arduino_connected>
  <is_arduino_input_displayed>>true</is_arduino_input_displayed>

  <!--
  Arduino's basic connection settings:
  device: specifies port of connection,
  speed_baud: defines speed of connection,
  mic_threshold: if above that number, the pressure parameter is increased.
  -->
  <arduino>
    <device>COM3</device>
    <speed_baud>9600</speed_baud>
```



```

    <mic_threshold>5</mic_threshold>
</arduino>

<!--
LeapMotion controller:
is_leap_motion_connected: switch on/off quering LeapMotion controller.
-->
<is_leapmotion_connected>>false</is_leapmotion_connected>

<!--
Kinect:
is_kinect_connected: switch on/off quering Kinect controller.
-->
<is_kinect_connected>>false</is_kinect_connected>

<!--
BoundingBox defines the boundaries of computational domain for physics simulation.
is_bounding_box_visible: switch on/off displaying of the bounding box.
-->
<bounding_box>
    <x_min>-6.5</x_min>
    <x_max> 6.5</x_max>
    <y_min>-4.5</y_min>
    <y_max> 4.5</y_max>
    <z_min>-4.5</z_min>
    <z_max> 4.5</z_max>
</bounding_box>
<is_bounding_box_visible>>false</is_bounding_box_visible>

<!--
Specify camera device id to use as texture on the softbody objects.
-->
<camera_device_id>0</camera_device_id>

<!--
Tag "camera" specifies basic OpenGL camera parameters.
-->
<camera>
    <distance>10.0f</distance>
    <fov>60</fov>
</camera>

<!--

```

```

Tag "time_step" defines time step for physics simulator.
Too big value may crash the numerical code, please test stability.
-->

<time_step>0.001</time_step>

<!--
This block of parameters defines whether mesh lines are drawn, vector normals
are displayed and also whether textured faces are displayed. By manipulating those
parameters, one can display only lines of meshes, or clean textured object, or
combination of those.
-->

<are_mesh_lines_displayed>>false</are_mesh_lines_displayed>
<are_vector_normals_displayed>>true</are_vector_normals_displayed>
<are_textured_faces_displayed>>true</are_textured_faces_displayed>

<!--
Soft Bodies:
This block defines all soft body objects, their names, simulation parameters.
Each object/body has separate sets of parameters.
-->

<soft_bodies>
  <body>
    <name>Bunny</name>
    <file>lofi-bunny.ply</file>
    <!--
      Parameters:
      gravity: defines gravity direction acting on the object,
      position: defines initial position,
      mass: defines mass of each point,
      pressure: defines pressure parameter (how much air is inside),
      time_to_die: how many seconds the object is active, then disappears,
      time_to_fadeout: how long object disappears in seconds,
      is_active_at_start: whether object is displayed at the start of simulation or
      waits for use input on kienct tracking to be displayed,
      max_pressure: safety boundry, how much air we can pump in,
      min_pressure: safety boundry, how much air we can deflate.
    -->

    <gravity>
      <x>0.0</x>
      <y>0.0</y>
      <z>0.0</z>
    </gravity>
    <position>

```

```

        <x>0.0</x>
        <y>0.0</y>
        <z>0.0</z>
    </position>
    <mass>1.0</mass>
    <spring_coefficient>221100</spring_coefficient>
    <damping_factor>30</damping_factor>
    <pressure>150000</pressure>
    <time_to_die>111</time_to_die>
    <time_to_fadeout>0.5</time_to_fadeout>
    <is_active_at_start>false</is_active_at_start>
    <max_pressure>2000000</max_pressure>
    <min_pressure> 150000</min_pressure>
</body>
<body>
    <!--
    Here is another body
    Here is another body
    -->
</body>
</soft_bodies>
</configuration>

```

APPENDIX B

On 8th July 2015, at the QUT (www.qut.edu.au) in Brisbane, Australia, the Science and Engineering Faculty Roadshow Expo was held. I accepted Dr. Tomasz Bednarz's suggestion to exhibit this artwork as a cross-section of art and science.

A pamphlet that includes a description of my work is shown in Figure 47. Figures 50-53 include photographs of audiences' interaction with my installation at the event. The installation was made on COW computer available at the QUT, with touch panel. I was informed that my work had attraction to many people, especially children.

Art + Science: Interactive Bunny

This work shows our interactive **Art + Science** system that conceptualizes reflective behavior on ourselves through a mirror.

Art: Spectators reflect and manipulate themselves on the projected physical model of a balloon, in this case a bunny shaped 3D mesh acting as a mirror.

Science: Balloon physics model code was implemented. The vertices of the meshes are represented as mass points governed by Newton's second law of motion, and the edges connecting these points as elastic links. The relative positions of the mass points and object deformations changed due to external forces, such as gravity, force of spring and pressure.

Interactions: During development of this project, we investigated three key modes of interaction, which are through human breath, keyboard and mouse actions: grabbing, pulling, releasing.

Authors: June Kim (AUT) and Tomasz Bednarz (QUT)

KEY STROKES

space bar	restart the simulation
f	switch on full screen mode
b	blow air inside the balloon
c	zero gravity
x/X	gravity in negative and positive X direction
y/Y	gravity in negative and positive Y direction
z/Z	gravity in negative and positive Z direction
p	pause simulation

Figure 47. Pamphlet for the Interactive Installation, experiment.

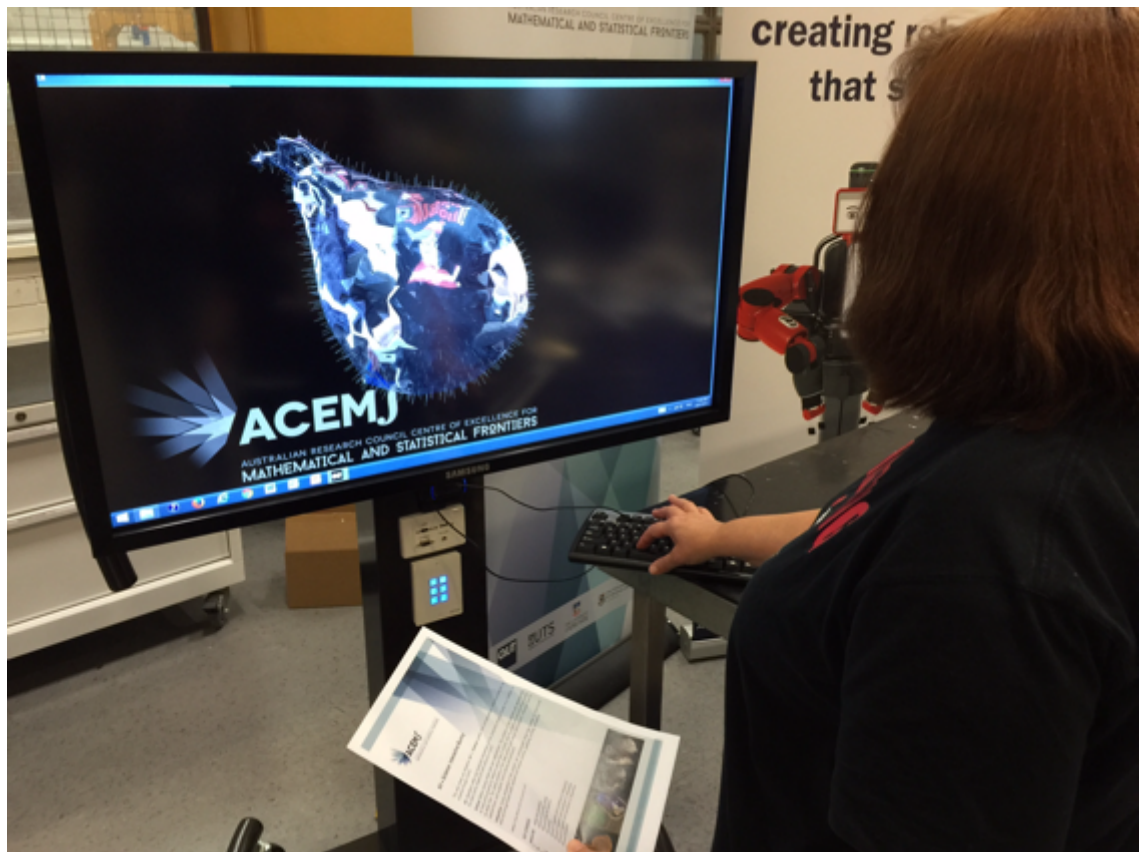


Figure 48. Interactions with the bunny using key strokes, at the QUT



Figure 49. Interactions with the bunny using touch screen, at the QUT.



Figure 50. Interactions with the bunny, using mouse, at the QUT.



Figure 51. Interaction with the bunny.

Figure 31