

Personal Project 1

Fog Screen



Paul Craven

Contents

Research

Interactive Art Installations

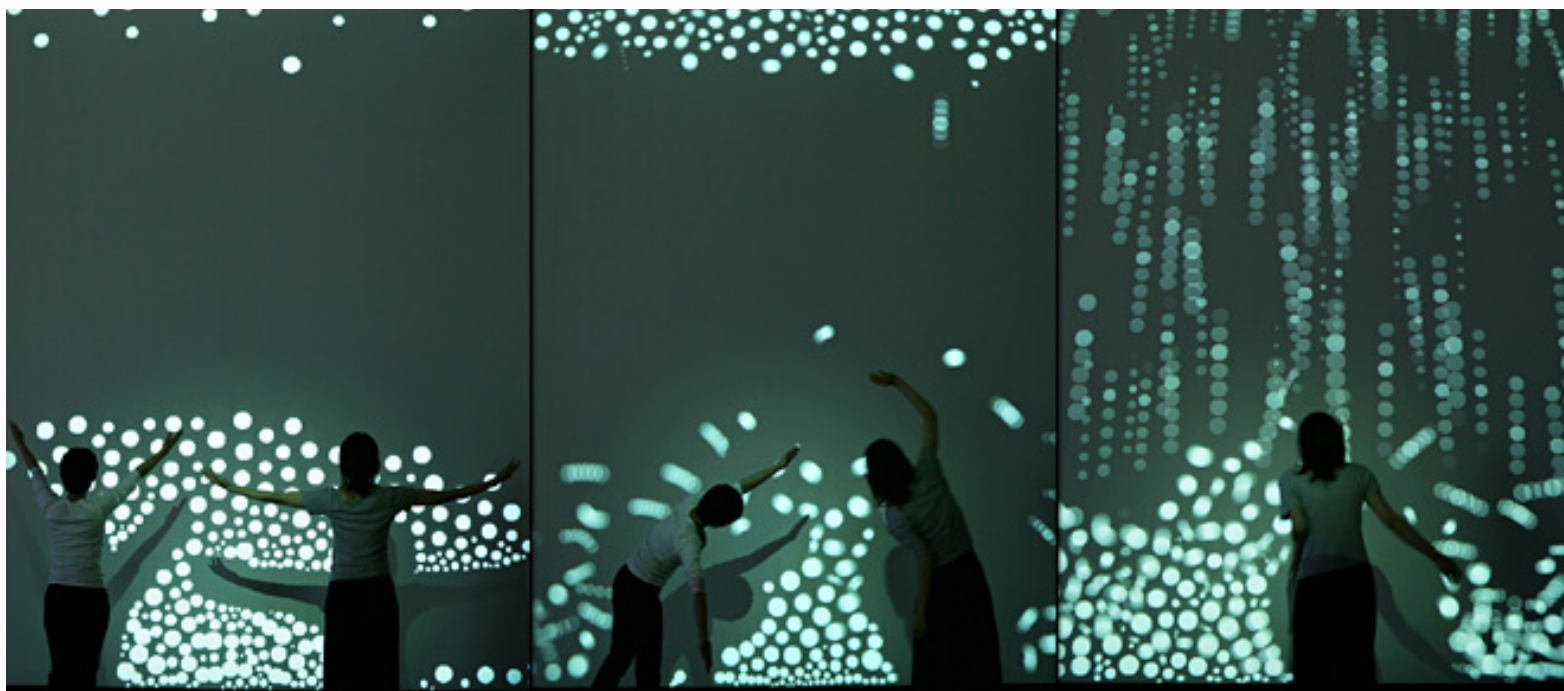
I have always enjoyed the of act of being part of the art and that it can be different for different people with different interactions keeping art new and almost fluid. My first research topic was interactive Art installations.

Some of my favourite interactive installations are mentioned here.

Footfalls (2006, by Tmema [Golan Levin & Zachary Lieberman]) is an interactive audiovisual installation in which the stomping of the visitors' feet creates cascading avalanches of bouncy virtual forms.

Footfalls was commissioned in July 2006 by the NTT InterCommunicationsCenter (ICC), Tokyo, for their solo exhibition of Tmema projects.

In Footfalls, stepping and stomping sounds produced by the visitors' feet are detected by microphones under the floor, and used to govern the size and number of virtual objects that fall from a six-meter high projection. The harder the visitors stomp, the more items fall. Using their silhouettes, visitors can then "catch" and "throw" these projected objects around.



Rain Room, by Random International, at the Curve Gallery, Barbican Art Gallery, in London.



Rain Room featured a 100 square metre room where it rained everywhere in the room except for the area in which you were standing. It was an intricate circuit of solenoids and pressure regulators combined with state of the art 3D tracking cameras. The images of people walking in it alone are mesmerising, I can only imagine what it must have felt like to walk between the raindrops.



Projection Mapping



One of my favourite projection mapping projects was a record breaking projection of a giant Namco Pac-Man Game using a custom made giant controller to play the game of Pac-Man projected across the river Thames and onto the side of the Millenium Mills in London Docklands. The Gadget Show teamed up with Namco Bandai the makers of Pac-Man and Rith (Running in the Halls) who are an interactive agency that designs and develops apps, websites and large scale interactive installations.

Together they broke the Guinness World Record for the world's largest architectural projection mapped game.

I love the idea of a giant arcade game that actually works and making the controller giant too was a great effect that would make it feel as though you were tiny playing this game.

Water Screens

Looking at new materials to project onto.
I found myself researching projection screens
and techniques for projecting.

Water Screens are some of the most
impressive shows that i have seen but they
take up massive amounts of space and there
usually needs to be large body of water in the
general vicinity.
However beautiful they look...they are certainly
the most impractical forms of projection
around.



Fog Screens

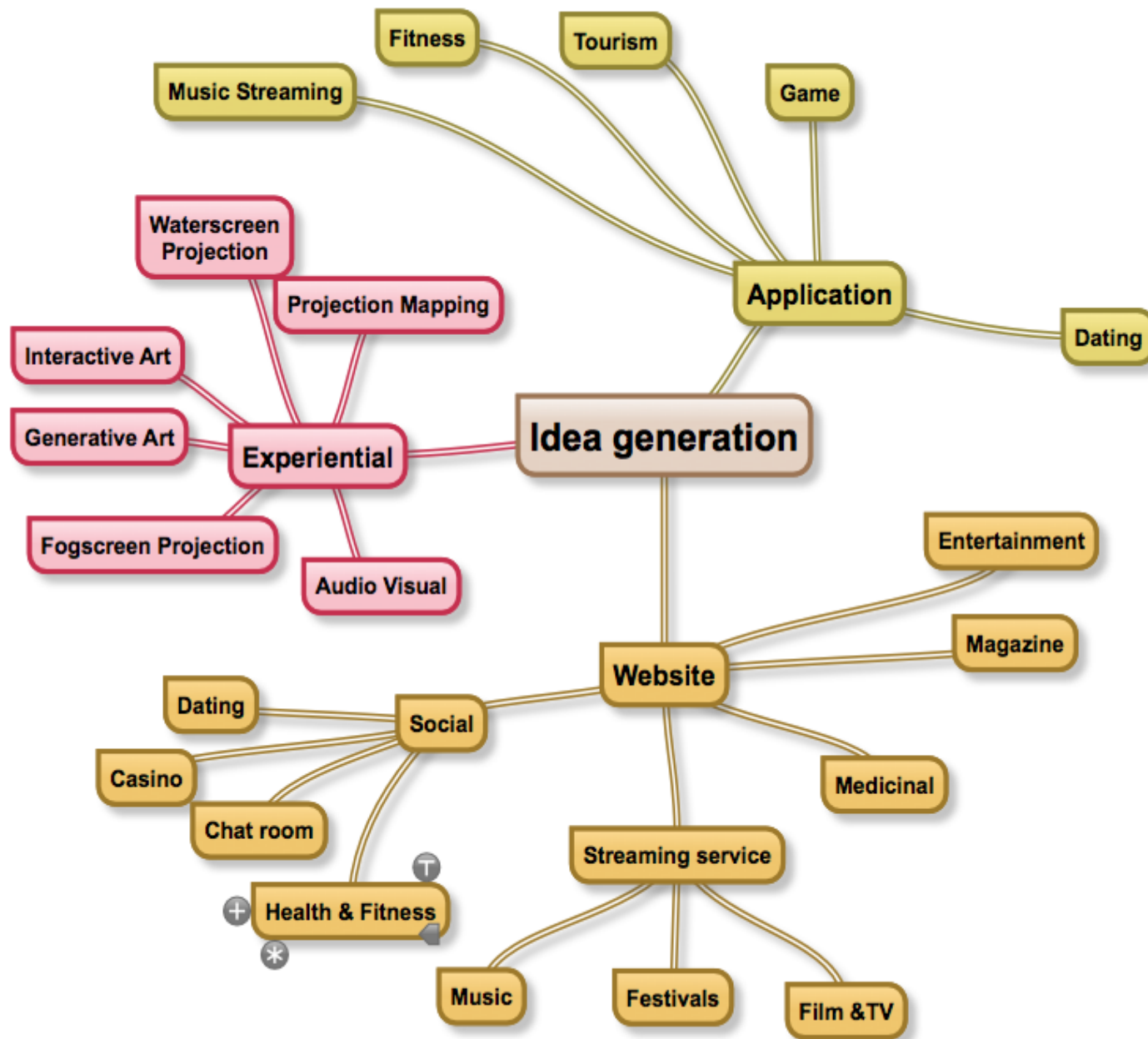


Fogscreens are by far the most exciting projection displays that are easily accessible. The only drawback to these are the prices.



A decent interactive fogscreen would set you back over £100,000 to buy and around £3,000 - £5000 per day to rent. These kind of prices make the fogscreen rare for nightclubs to have. I see this as a great aid to the VJ in a club if it were affordable. They are also very heavy and hard to manage,

Concept Development



Concept 1

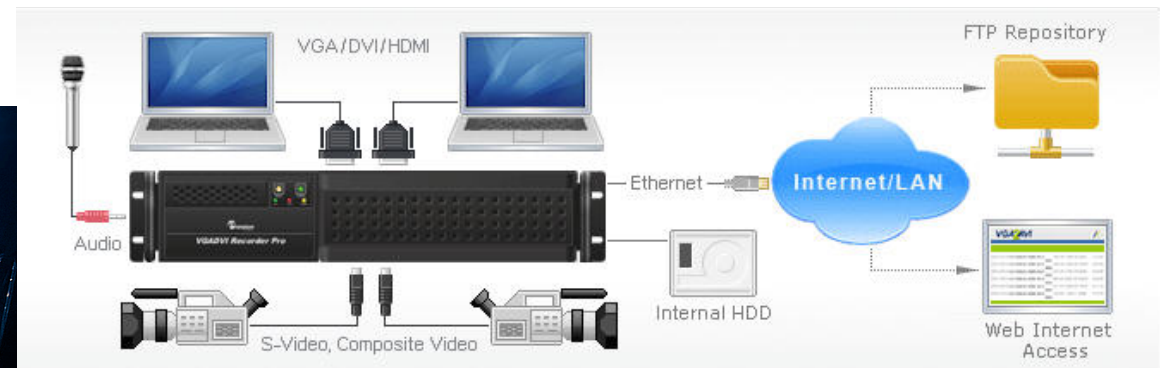
As commercial fogscreens are heavy and expensive there is a need for an interactive portable fogscreen that can be taken out to venues around Newcastle and display visual art in real time from the VJ in a new and exciting way. This is something that I haven't seen in a small venue and I feel that this would be a new way to attract people to nightclubbing in Newcastle. It could be interactive using Xbox Kinect technology to track people and generating new real time art.



Concept 2

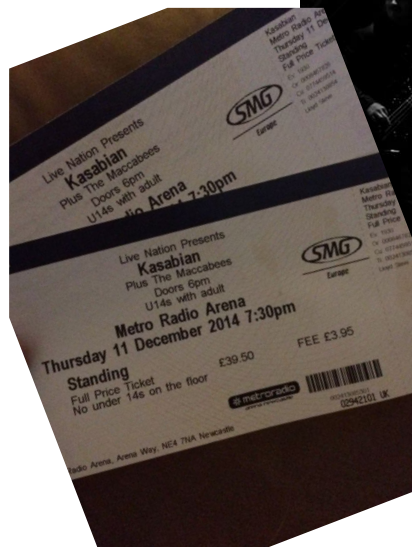
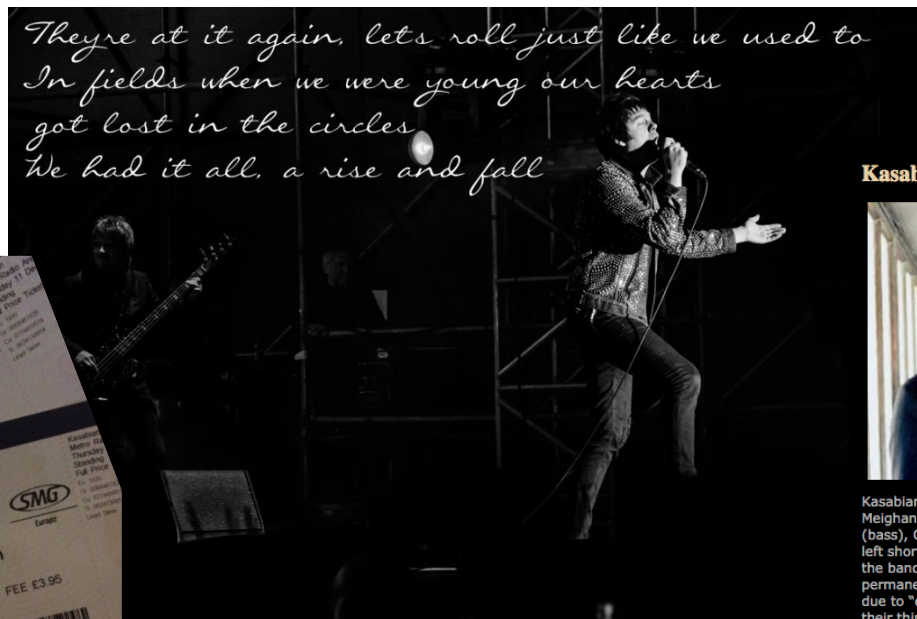
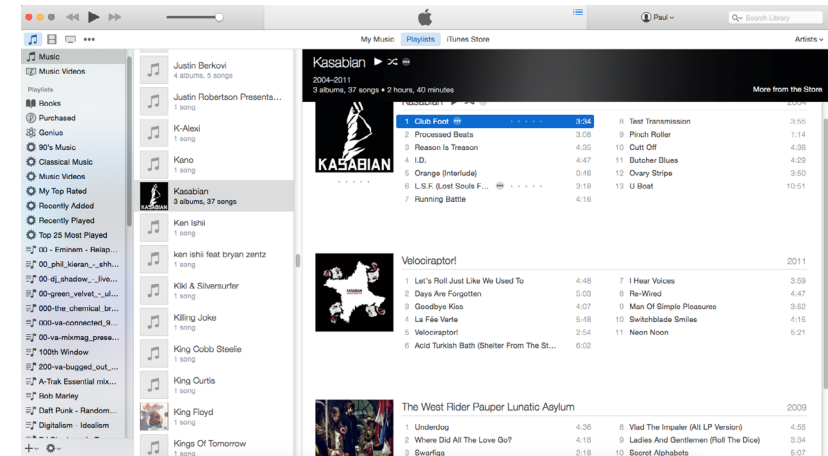
If like me when you go to a nightclub there is no chance that you get a glimpse of the DJ and you spend more time looking at the visuals whilst dancing, plus the fact that nights out are expensive. You might grudge paying £20 to see a DJ that you haven't seen before when you might not even like them on hearing them. Or maybe you are a student and would like to have the DJ in your flat party....Then a streaming service is what is needed. There is 1 dedicated dance event streaming team and they are called boiler room, but they only do big name events.

I aim to change that by streaming local events in Newcastle upon Tyne. I will focus on delivering a new brand with a reliable service that will be free for people to stream to their home computers or handheld devices. It will focus on the audio visual art and music scene in the North East.



Concept 3

Travel application based on music taste. The application would be able to suggest places to visit based on your musical tastes. It takes a look at your spotify or itunes lists and based on what you listen to it will suggest gigs or places of interest to visit. If for instance you listen to Kasabian the app would intuitively notify you about the latest tour dates and where you can buy tickets from it would also supply information on where they are from and where they played their first gigs etc. Places that a Kasabian superfan might want to visit?



Kasabian Biography



Kasabian are an English rock band from Leicester, formed by Tom Meighan (vocals), Sergio Pizzorno (guitar and vocals), Chris Edwards (bass), Chris Karloff (Keyboards and Guitar) and Ash Hannis (drums) who left shortly before the release of the first album. During the recording of the band's second album, the drummer Ian Matthews became a permanent member. Shortly after this, Chris Karloff was asked to leave due to "creative differences". He was replaced during the recording of their third album by Jay Mehler.

The band members are from the Leicestershire villages of Blaby and Countesthorpe. They met while attending Leysland High School and Countesthorpe Community College both in the village of Countesthorpe.

The band took their name from Linda Kasabian, the young pregnant girl who drove the car in which the killers escaped after the murder of Leno and Rosemary LaBlanca, the second of the Tate-LaBlanca murders.

The band has been compared with the likes of Primal Scream, with whom they share an electronica-indie rock fusion, and The Stone Roses and Oasis, with whom they share confidence, swagger and rough vocal style.



Further Development

Kinect Hack

I borrowed the Kinect sensor from the uni and then started researching hacks to access the different sensors within the Kinect.

First you need to install Xcode from the Mac App Store. Once installed, open the application and select the menu option Xcode / Open Developer Tool / More Developer Tools... Clicking this will take you to the Apple Developer website. login using your Apple ID and password and continue until you see the 'Downloads for Apple Developers' screen. Find the Xcode 6 Developer Tools.

Next In the list of downloads, search for the most recent version of Command Line Tools (OS X Mavericks) for Xcode6. Click on the blue disk image link to the right of the description to download.

Install OS X Mavericks Command Line Tools for Xcode6. Once you've downloaded the Command Line Tools, simply open the disk image and install the package.

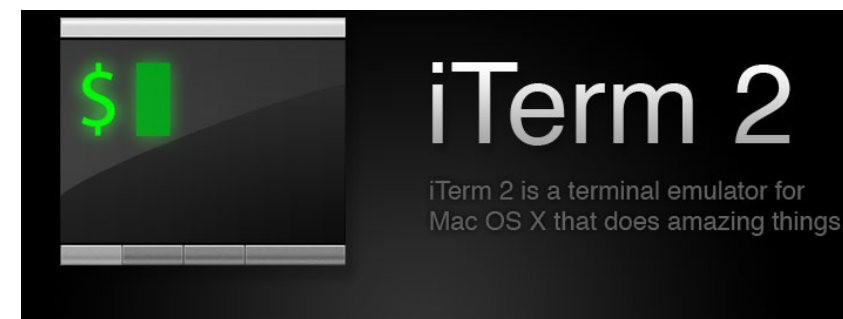
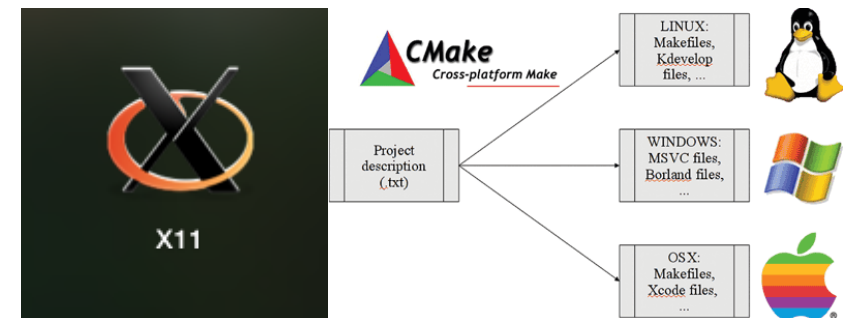
Downloads for Apple Developers Hi, Paul Craven | My Profile | Sign out

Search: Xcode 6 1 - 12 of 12 Page 1 of 1

Categories	Description	Release Date
<input checked="" type="checkbox"/> Applications (13) <input checked="" type="checkbox"/> Developer Tools (211) <input checked="" type="checkbox"/> iOS (16) <input checked="" type="checkbox"/> OS X (71) <input checked="" type="checkbox"/> OS X Server (9) <input checked="" type="checkbox"/> Safari (1)	Xcode 6.1.1 This is a release version of the complete Xcode developer toolset for Mac, iPhone, iPod touch, and iPad. It includes the iOS 8.1 SDK, OS X 10.10 SDK and OS X 10.9 SDK. Xcode 6.1 requires OS X Yosemite or OS X Mavericks.	Dec 2, 2014
	Command Line Tools (OS X 10.10) for Xcode - Xcode 6.1.1	Dec 2, 2014
	Command Line Tools (OS X 10.9) for Xcode - Xcode 6.1.1	Dec 2, 2014
	Xcode 6.1	Oct 22, 2014
	Hardware IO Tools for Xcode - Xcode 6.1	Oct 16, 2014
	Command Line Tools (OS X 10.9) for Xcode - Xcode 6.1	Oct 16, 2014
	Command Line Tools (OS X 10.10) for Xcode - Xcode 6.1	Oct 16, 2014
	Audio Tools for Xcode - Xcode 6.1	Oct 16, 2014
	Auxiliary Tools for Xcode - Xcode 6.1	Oct 16, 2014
	Graphics Tools for Xcode - Xcode 6.1	Oct 16, 2014

Next up is a list of applications/tools that you'll need to install.

1. Download XQuartz, an open source version of the X.org X Window System for Mac.
2. Download CMake, a cross-platform build automation system. Select the download link beside Mac OSX 64/32-bit Universal (for Intel, Snow Leopard/10.6 or later), Part way through the CMake installation you'll be asked to Install Command Line Tools, make sure you do this by clicking the Install Command Line Links button. If you get an error at this point, you'll need to follow all the uninstallation instructions and then reinstall.
3. Download I term2, which allows you to compile, install and upgrade software via Terminal app.



Kinect Hack Cont'd

At this point we've installed our primary applications, so now we just need to install a couple of dependencies for later on.

Libtool: A generic library support tool.

In Terminal type in:

```
Last login: Wed dec 10 23:37:10 on ttys000
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]sudo port install libtool[/stag_code]
```

Press Return, and then enter your password when prompted. If all went well, we should see something similar to the following appear in Terminal:

```
Last login: Wed dec 10 23:37:10 on ttys000
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]sudo port install libtool[/stag_code]
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]→ Fetching archive for libtool
→ Attempting to fetch libtool-2.4.2_3.darwin_13.x86_64.
tbz2 from http://jog.id.packages.macports.org/macports/
packages/libtool
→ Attempting to fetch libtool-2.4.2_3.darwin_13.x86_64.
tbz2.rmd160 from http://jog.id.packages.macports.org/
macports/packages/libtool
→ Installing libtool @2.4.2_3
→ Activating libtool @2.4.2_3
→ Cleaning libtool
→ Updating database of binaries: 100.0%
→ Scanning binaries for linking errors: 100.0%
→ No broken files found.[/stag_code]
```

Next we need to install

Libusb: A usb support tool for applications.

In Terminal, type in the following:

```
Last login: Wed dec 10 23:37:10 on ttys000
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]sudo port install libusb +universal[/
stag_code]
```

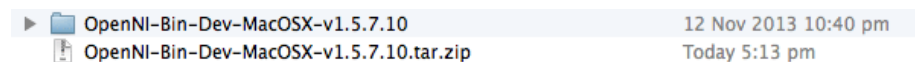
Click Return and then enter your password when prompted. If everything installed correctly, you should see something similar to this:

```
Last login: Wed dec 10 23:37:10 on ttys000
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]sudo port install libusb +universal[/
stag_code]
Pauls-MBP:~ paulcraven$ [stag_code data_line=""
style="markup php"]→ Fetching archive for libusb
→ Attempting to fetch libusb-1.0.18_0+universal.dar-
win_13.i386-x86_64.tbz2 from http://jog.id.packages.
macports.org/macports/packages/libusb
→ Attempting to fetch libusb-1.0.18_0+universal.dar-
win_13.i386-x86_64.tbz2 from http://packages.macports.
org/libusb
→ Attempting to fetch libusb-1.0.18_0+universal.dar-
win_13.i386-x86_64.tbz2 from http://mse.uk.packages.
macports.org/sites/packages.macports.org/libusb
→ Fetching distfiles for libusb
→ Attempting to fetch libusb-1.0.18.tar.bz2 from http://
aarnet.dl.sourceforge.net/libusb
→ Verifying checksums for libusb
```

More Kinect Hack Cont'd

Now we are moving into Kinect territory. OpenNi is the first Kinect-specific software we need to install. OpenNi is created by a group of companies including PrimeSense (now owned by Apple), who developed the original Microsoft Kinect. The OpenNi software itself contains a stack of middleware libraries and applications that will help get me going on my Kinect journey!

1. Create a new folder on your Mac called 'Kinect'. This will help keep things organised (I'll be referring to this folder later on).
2. Download OpenNI v1.5.7.10 SDK for Mac OS X.
3. Move the file you just downloaded to your Kinect folder and double click to uncompress and reveal the SDK folder.



Now I am ready to install the OpenNi SDK on my Mac. Open your terminal app again, we will now navigate to the OpenNi SDK folder using the command. Start by typing in cd, add a space, and finally drag your SDK folder from the Finder onto the Terminal window and press Return. This will automatically enter the file path for you, and look something like this:

```
[stag_code data_line="" style="markup php"] Pauls-MBP:~ paulcraven$ cd /Users/glen/Sites/kinect/OpenNI-Bin-Dev-MacOSX-v1.5.7.10
Pauls-MBP:OpenNI-Bin-Dev-MacOSX-v1.5.7.10 glen$[/stag_code]
```

This is the good part now press Return and then enter your password to install OpenNi SDK:

```
[stag_code data_line="" style="markup php"]sudo ./install.sh[/stag_code]
```

If everything installed correctly, then you'll see the following text:

```
[stag_code data_line="" style="markup php"]Installing
OpenNI
*****
copying shared libraries...OK
copying executables...OK
copying include files...OK
creating database directory...OK
registering module 'libnimMockNodes.dylib'...OK
registering module 'libnimCodecs.dylib'...OK
registering module 'libnimRecorder.dylib'...OK
creating java bindings directory...OK
Installing java bindings...OK
*** DONE **[/stag_code]
```

More Kinect Hack Cont'd

I need to quickly complete this step to prevent errors during the SensorKinect installation.

Open Terminal on your Mac: Applications / Utilities / Terminal.
Next type in the following and press Return:

```
[stag_code data_line="" style="markup php"]sudo ln -s /usr/local/bin/niReg /usr/bin/niReg[/stag_code]
```

I will install SensorKinect this is simply a module that helps the Kinect interact with OpenNI. I need it for my Kinect to work on my Mac.

1. Visit the SensorKinect project on GitHub and click the Download ZIP button that looks like this:



2. Once downloaded, move the SensorKinect-unstable.zip file to your Kinect folder and double-click to uncompress it.

3. Navigate to the SensorKinect093-Bin-MacOSX-v5.1.2.1.tar.bz2 file inside the Bin folder, and double-click to uncompress it.

Now we open the terminal app again and type: cd the space and then drag the folder Sensor-Bin-MacOSX-v5.1.2.1, this will navigate to this folder and we should see the following:

```
[stag_code data_line="" style="markup php"]Paul'sMBP:~paulcraven$ cd /Users/glen/Sites/kinect/SensorKinect-unstable/Bin/Sensor-Bin-MacOSX-v5.1.2.1
Paul's MBP:Sensor-Bin-MacOSX-v5.1.2.1 glen$ [/stag_code]
```

Hit return and if it all went ok you will see:

```
[stag_code data_line="" style="markup php"]Installing PrimeSense Sensor
*****
creating config dir /usr/etc/primesense...OK
copying shared libraries...OK
copying executables...OK
registering module 'libXnDeviceSensorV2KM.dylib' with OpenNI...OK
registering module 'libXnDeviceFile.dylib' with OpenNI...OK
OK
copying server config file...OK
setting uid of server...OK
creating server logs dir...OK
*** DONE ***[/stag_code]
```

Finally we need to install Nite this is some amazing middleware that's made by the same team behind OpenNI. It monitors depth, colour, infrared and audio to track your body and hands for use as input devices.

1. Download NiTE v1.5.2.21 for Mac OS X.

2 Once downloaded, move the NITE-Bin-MacOSX-v1.5.2.21.tar.zip file to your Kinect folder and double-click to uncompress it.

Kinect Hacked

Finally we need to install Nite this is some amazing middleware that's made by the same team behind OpenNI. It monitors depth, colour, infrared and audio to track your body and hands for use as input devices.

1. Download NiTE v1.5.2.21 for Mac OS X.
- 2 Once downloaded, move the NITE-Bin-MacOSX-v1.5.2.21.tar.zip file to your Kinect folder and double-click to uncompress it.

And just like before, open Terminal and type in cd, followed by a space, and then drag the NITE-Bin-Dev-MacOSX-v1.5.2.21 folder into the Terminal window. Hit Return to navigate to that directory, and you should see something like this:

```
[stag_code data_line="" style="markup php"]Paul'sMBP:~
paulcraven$ cd /Users/glen/Sites/kinect/ NITE-Bin-Dev-
MacOSX-v1.5.2.21 Paul's MBP:Sensor-Bin-MacOSX-v5.1.2.1
glen$ [/stag_code]
```

After this, we can install NiTE by typing in the following, and then hitting Return:

```
[stag_code data_line="" style="markup php"]sudo ./
install.sh[/stag_code]
```

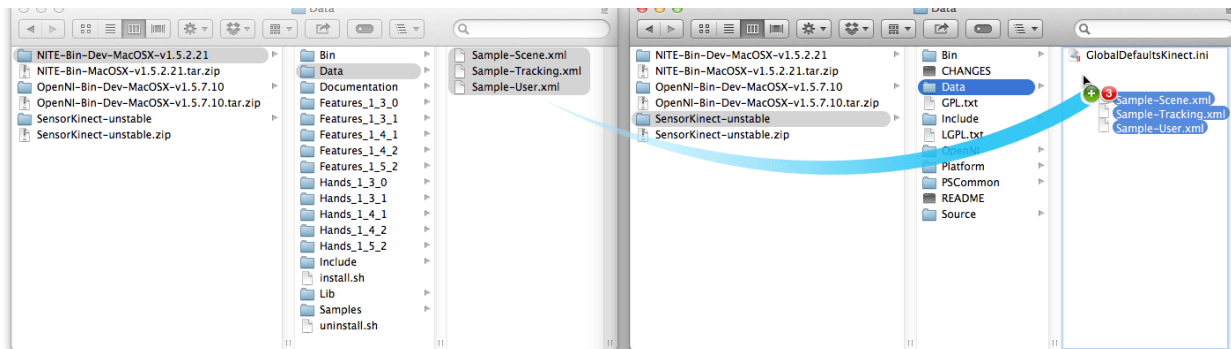
A successful install will look something like this:

```
[stag_code data_line="" style="markup php"]Installing
NITE *****
Copying shared libraries... OK
Copying includes... OK
Installing java bindings... OK
Installing module 'Features_1_3_0'...
ls: Features_1_3_0/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_3_1'...
ls: Features_1_3_1/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_4_1'...
ls: Features_1_4_1/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_4_2'...
ls: Features_1_4_2/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_5_2'...
Registering module 'libXnVFeatures_1_5_2.dylib'... OK
Copying XnVSceneServer... OK
Installing module 'Features_1_5_2'
ls: Hands_1_3_0/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_5_2'
ls: Hands_1_3_1/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_5_2'
ls: Hands_1_4_1/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_5_2'
ls: Hands_1_4_2/Bin/lib*dylib: No such file or directory
Installing module 'Features_1_5_2'
registering module 'libXnVHandGenerator_1_5_2.dylib'...
OK
Installing .NET wrappers...
Package exported to: /Library/Frameworks/Mono.
framework/Versions/3.2.0/lib/mono/2.0/XnVNite.net.
dll -> ../gac/XnVNite.net/1.5.2.21__399870656fc6c787/
XnVNite.net.dll
Installed Bin/XnVNite.net.dll into the gac (/Library/
```

Kinect Hacked

Finally we can transfer some sample files and test out our installations.

Copy the sample xml files from NiTE / Data over to your SensorKinect / Data folder like so (hold the Option key while dragging to copy files in Mac OS X):



And now finally, to run your first demo, type the following into your Terminal window and press Return:

```
[stag_code data_line="" style="markup php"]./Sample-PointViewer[/stag_code]
```

If everything is setup correctly, PointViewer should open in a new screen. This is a tracking demo, so stand back, wave your hand around in the air, and pretty soon you should notice a white tracking line appear.

In a new finder window, locate your x64-Release folder by following this path: NiTE / Samples / Bin / x64-Release. Open your Terminal app again: Applications / Utilities / Terminal. Type cd, followed by a space, and then drag the x64-Release folder onto the Terminal window to fill in the path and press Return. You should see something like this:

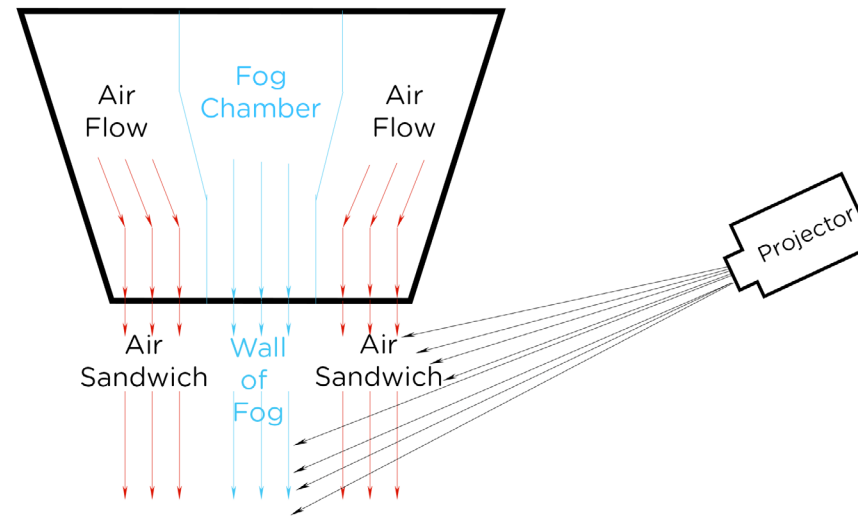
```
[stag_code data_line="" style="markup php"]Paul'sMBP:~
paulcraven$ cd /Users/glen/Sites/kinect/ NITE-Bin-Dev-
MacOSX-v1.5.2.21/Samples/Bin/x64-Release
Glens-MacBook-Pro:x64-Release glen$ [/stag_code]
Paul's MBP:Sensor-Bin-MacOSX-v5.1.2.1 glen$ [/stag_
code]
```



Video available at <http://youtu.be/O2KsTq2n4EM>

Fogscreen

The FogScreen is a high-tech version of the technology in a cool air humidifier. Tap water is pumped into the fog tank where it is blasted with ultrasound, turning it instantly into a thick fog made of tiny water particles 2-3 microns in diameter. The tank's internal design plus 3 sets of fans work together to create a very thin wall of mist about half of an inch thick. One set of fans blows the fog downwards while the other two sandwiches the fog between air curtains so that it becomes a smooth projection screen.



The Build

What do I Need?

A list of parts and tools to complete this task are:

Tools:

Jig saw
Circular Saw
Drill
Clamps
Hand Saw
Electric Sander
Paint Brushes



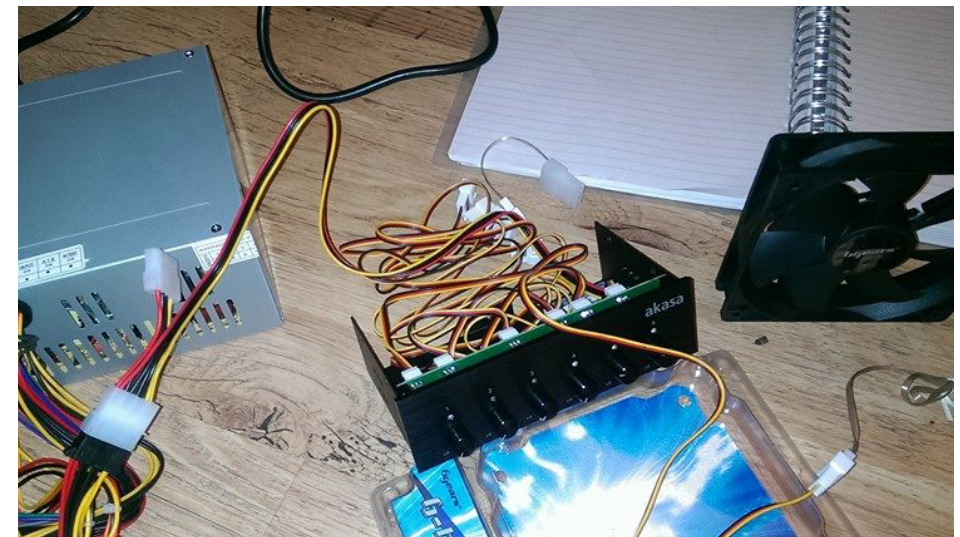
Materials:

1.2m x 2.4m x 18mm plywood
Wood Screws
Wood Glue
Black Satin Paint
Straws



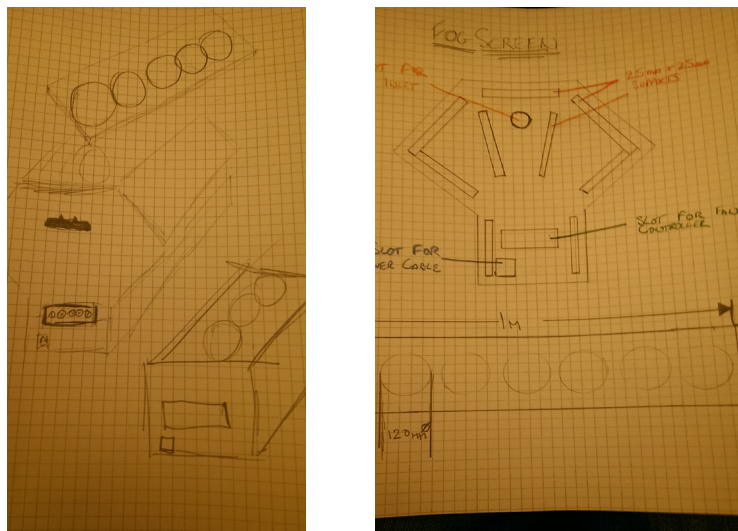
Electrical:

18 x 120mm case fans
6 fan controller
400w power supply



The Exterior

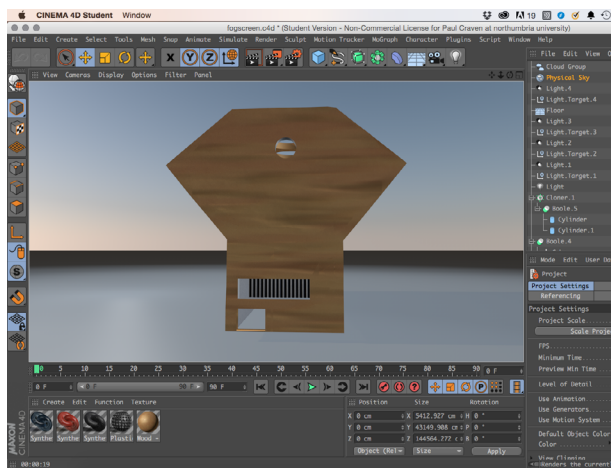
Because of the price of a Fogscreen I have decided to build my own. First I drew a few designs on paper to see how it would look.



I took an 8ft by 4ft piece of 18mm ply wood. I cut it into 8 x 1m by 19cm boards with 120mm diameter holes for the fans cut into 3 of the boards and 2 600mm x 600mm end boards for me to construct my fogscreen.

I cut the end shapes out of the 600mm boards and screwed short lengths of 25 x 25mm pine supports to the inside of the ends, these will attach the sides to the end boards. I have cut a 40mm hole in one end for the smoke inlet.

I then took this drawing and using Maxon C4d I created a 3D model of my design



The Exterior

The next stage was to finish the exterior, I filled with wood filler the holes to smooth them out then I painted the outside of the fan boards before fitting it together so that when I connect the fans I don't get paint on them. Then taking some clamps and vices I clamped the exterior of the box together held together with wood screws and wood glue.

On the ends I placed some strong handles so that it can be portable. I cut the holes for the fan controller and the power cables.

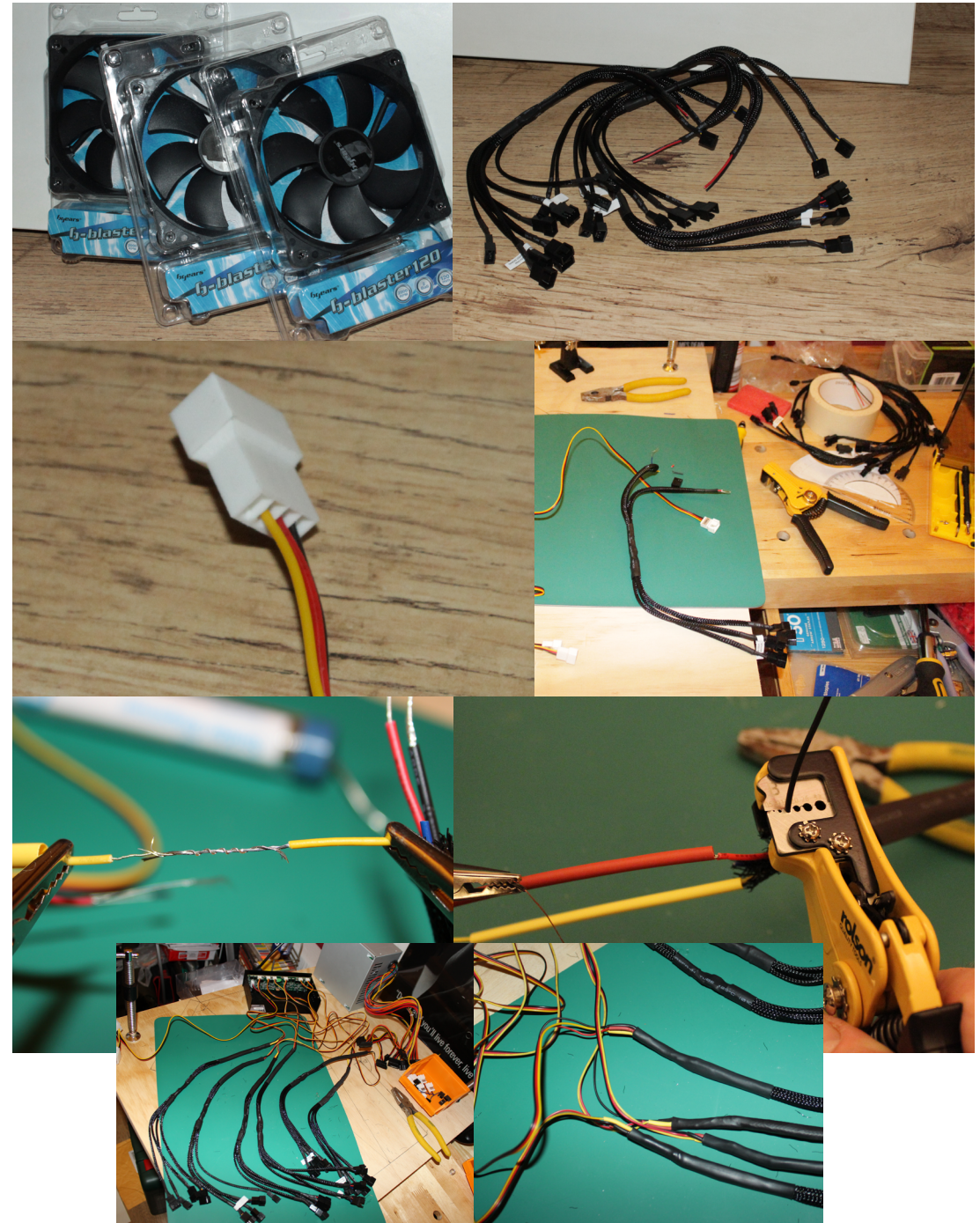


The Fans

As I wanted to connect 18 fans at the same time I decided to use a 6 fan controller to control the airflow and allow for a bit of control over the fog output. Each output of the controller could take 20 watts so decided to attach 3 fans to each channel. Each fan was 4 watts so this would allow for 5 fans to be connected per channel if needed.

The fans had to be connected to some splitter cables, however the splitter cables and the cables from the controller had different fittings this meant that i had to strip the wires back and solder them to the splitter cables, and then heat shrinking cables again. This was very time consuming.

I then connected all of the fans to the controller to test them.



The Inside

The next stage was to finish the inside of the box and install all of the fans and wire it up correctly this again was very time consuming.

First I placed the power supply in the bottom with the controller above it. Then I had to fit the straws to give the airflow some direction, the air needs to be pushed through a honeycomb, I made my own with 2000 drinking straws glued together.

Then I fitted the fans to the boards.

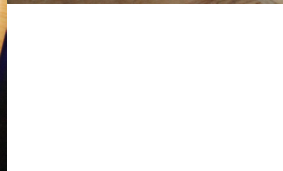


The Finishing Touches

I put everything together with the funnel shape inside the box this would push the airflow into a smaller gap increasing my airflow by forcing it through a smaller exit hole. This would also push the fog through in a laminar flow giving an almost flat surface to project onto.



The Finished Product



Evaluation

Conclusions

Overall I am happy with how the fogscreen turned out, it worked as well as I expected it to. I had a few setbacks along the way but I feel that I have product which can be used in various settings and can be easily made interactive with Kinect or Arduino. This will give me some scope to move this project along further in the next semester if needed. I will look further into interactive art installations and see what possibilities are available in the future,