

SEMIOPTICS FOR SPINOZA, SHADOW OBJECT 4

BY RAFAEL LOZANO-HEMMER

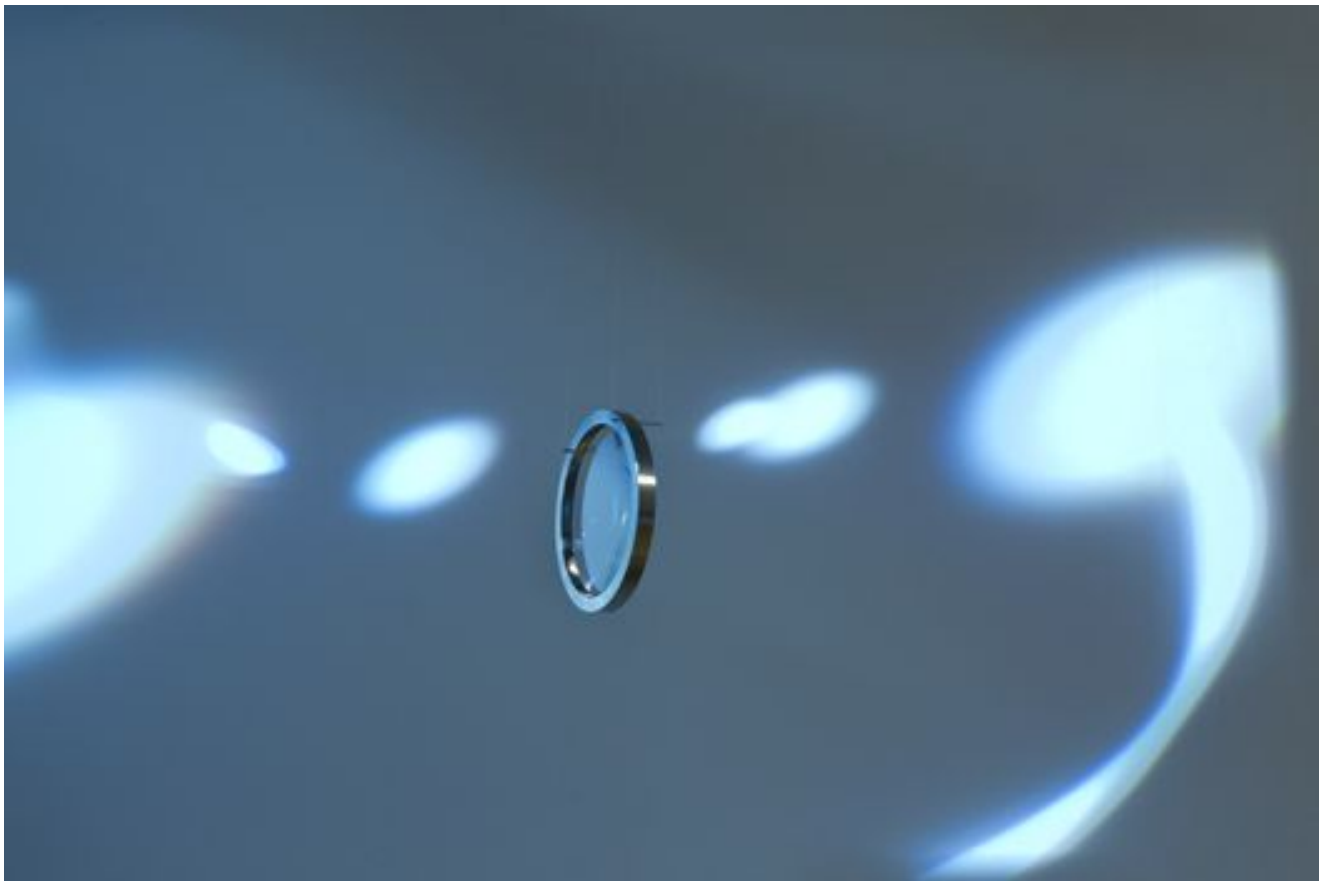


TABLE OF CONTENTS

GENERAL IMPORTANT INFORMATION

Technique	3
Description	3
Operation	3
Maintenance	4
Placement Instructions	4

DETAILED TECHNICAL INFORMATION

Normal Software Operation	11
Manual Software Calibration	12
Remote Access to Artwork's Computer	21
Preliminary Troubleshooting Steps	22
Troubleshooting Assistance	23
Support (Contact Us)	24

APPENDIX I - INSTALLATION

Description of Components	25
Wiring Diagrams and Connections	29
Close-Up Photos of Mounting	32

APPENDIX II - TECHNICAL DATA SHEETS

Microsoft Kinect 360	33
Installation Photos	34

GENERAL IMPORTANT INFORMATION

This short section must be read for proper operation.

SEMIOPTICS FOR SPINOZA, SHADOW OBJECT 4 (2012)

BY RAFAEL LOZANO-HEMMER

Technique

Computer, Microsoft Kinect 360 camera, projector, metal, motor, Arduino processor, 150 mm lens, custom ring mount.

Description

One plano-convex lens, similar to the telescopes made by philosopher Baruch Spinoza for Huygens, is suspended from a thread and moves (with the airflow and with a small motor.) Projected onto the wall behind the lens is a set of artificial lens flares, generated by equations that take into account the orientation of the lens in relation to the public. The project generates fake light beams that go through the real lens, and the lens then diffracts and scatters the fake light creating light source aberrations, directional haze, and chromatic distortion. The piece is a mobile that can be moved by touching it lightly or by blowing upon, which changes the corresponding lens flares.

Please DO NOT TOUCH the lens, as a motor is rotating it slowly.

Operation

Please refer to [Appendix I - Installation](#) for detailed system information and wiring diagram.

1. Connect the computer, the Kinect, the motor driver, and the projector to electrical power. Use the supplied power cables.
2. Depending on the version of artwork you have, you might find a small black box on which to connect a power cable. On that box, you will also find a button that is used to turn the artwork ON/OFF.

If your version does not have this box, you will either use the computer's power button to turn the artwork ON/OFF, or you must set the computer to an automatic power schedule.

3. To turn the piece ON, press the power button on the computer for one second, then release it. Important note: Please do not push the button again as this will shut down the piece. Wait at least two minutes before pressing it again as the computer might take this long to boot. After two minutes (maybe faster), you should see the piece.
4. To turn the piece OFF, press the power button on the side of the small box, or the computer button.

Wait about 60 seconds until everything (including the projector) finishes its shutdown routine. The projector is ready to be used again once the lens door is closed, or the indicator LEDs on the projector show that the cool down process is done.

5. If the piece doesn't start within two minutes, try to turn on the piece again. If it still doesn't turn on, then hold the power button all the way down for 10 seconds. Then, wait at least three seconds and press the power button all the way down for one second, and you should be up and running again.

Maintenance

Please do not clean the Kinect camera surfaces with Windex or soap. Use a lint-free cloth and a LCD screen liquid cleaner, such as Kensington Screen Guardian, typically found in computer stores. When cleaning the Kinect, avoid applying too much pressure on its surface otherwise the Kinect could swivel and it will mess up the alignment of the artwork.

The metal support structure can be cleaned with regular all-purpose cleaner. Do not use harsh cleaners or rough sponges.

We recommend cleaning the piece every two months at least.

Placement Instructions

The artwork comes in two versions: Version 1 is a metal structure that contains all elements (camera, projector, computer, motor), and Version 2 is comprised of a larger projection, in which one or more projectors can be hung separately.

Version 1:

The projector structure should be hung parallel to the floor, or perfectly horizontal, in order for the projection to look straight.

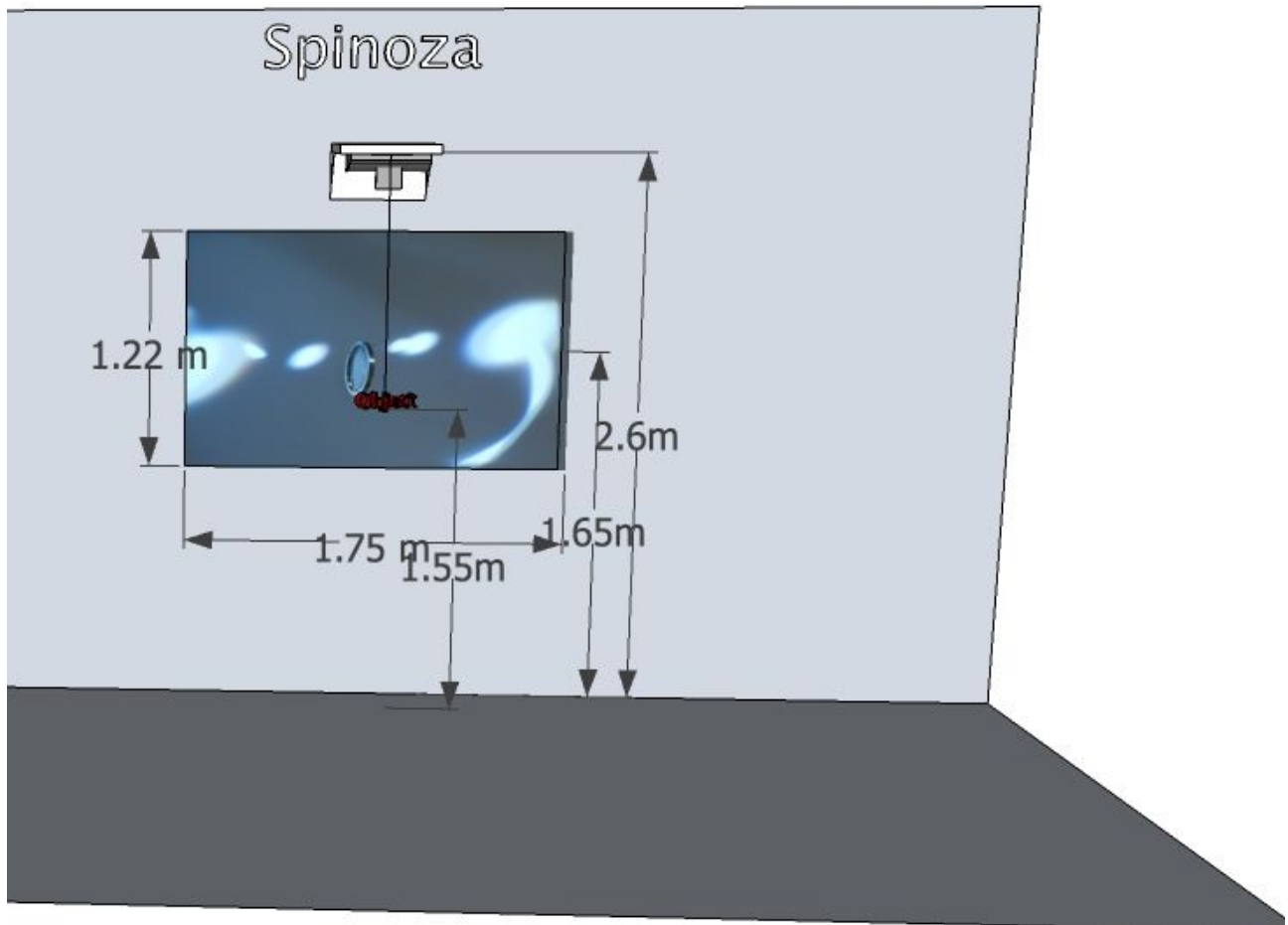
You might have to use the projector's remote control to adjust the key-stoning of the image. You can also adjust the projector's tilt by loosening the four screws that attach the projector to the metal structure, on the top side of the structure.

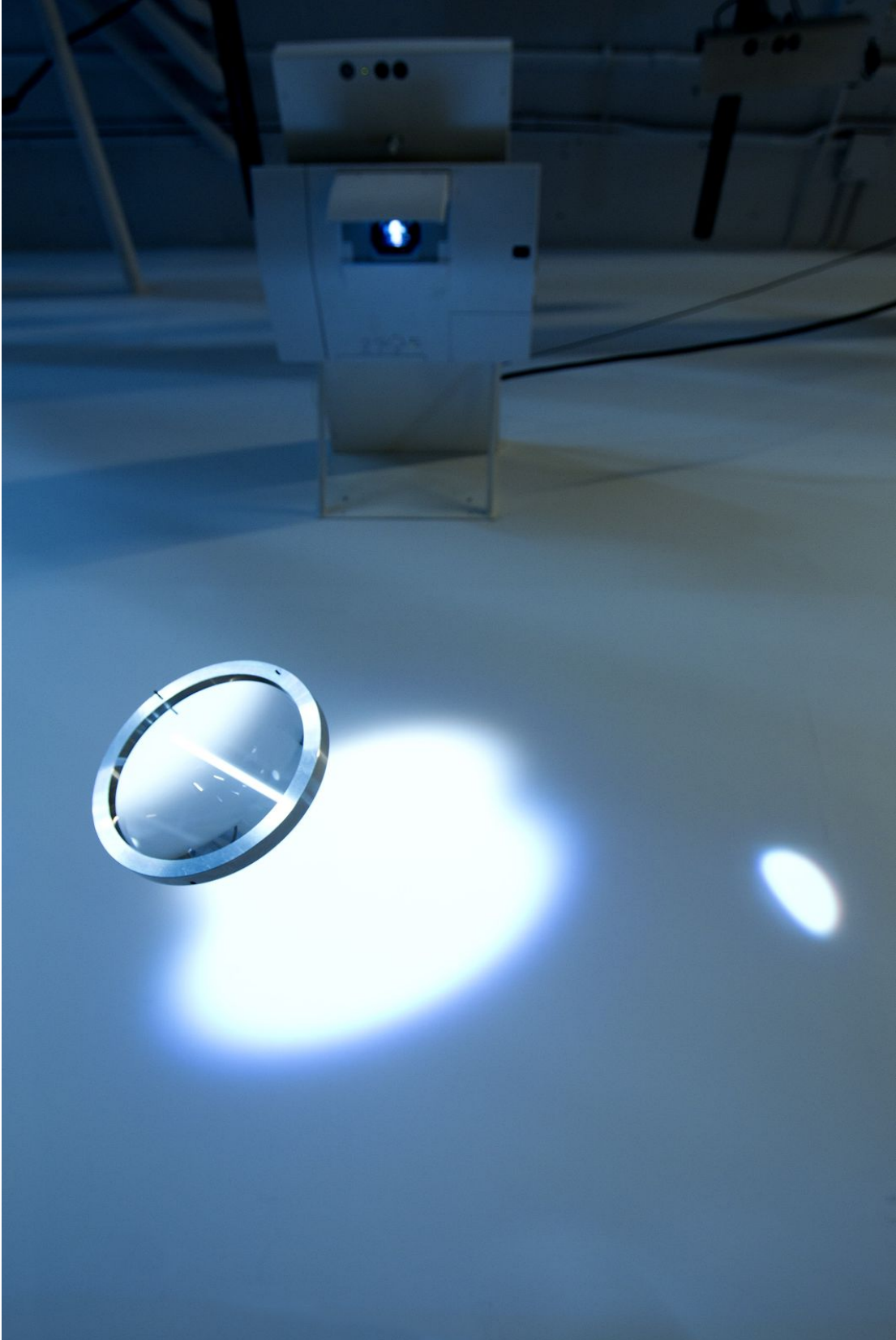
The ceiling height of the exhibition space should be at least 290 cm (114"). Mount the projector structure so that the top of the wall plate is at 290 cm.

Now attach the lens to a nylon string and hang it on to the hook. The hook is located close to the camera lenses.

Ensure that convex side of the lens faces in to the ring otherwise the lens will be off-balanced.

For Version 1, please consult the following images.





Version 2:

For this version, the projectors are mounted separately from the Kinect camera mount. This means that any type of projector can be used and any image size can be achieved. We usually use three projectors to fill three walls of a room.

The Kinect camera mount attaches to a pole lowered from the ceiling in the middle of the room.

Depending on where the computer is installed, a Black Box USB extender might be needed for the Kinect camera USB connection.

The projectors should use an ultra short throw lens which allows them to be placed very close to the projection walls. This will prevent people from casting any shadows onto the walls. Please consult the [Appendix](#) for brand and model suggestions.

In the following installation photos, the distance between the Kinect and top of the lens is 149 cm, and the distance from the bottom of the lens to the floor is 135 cm.

For Version 2, please consult the following images.



Kinect camera hung from a square tube, with lens.



Museum setup: 3 ceiling projectors and a kinect hanging from a square pole.

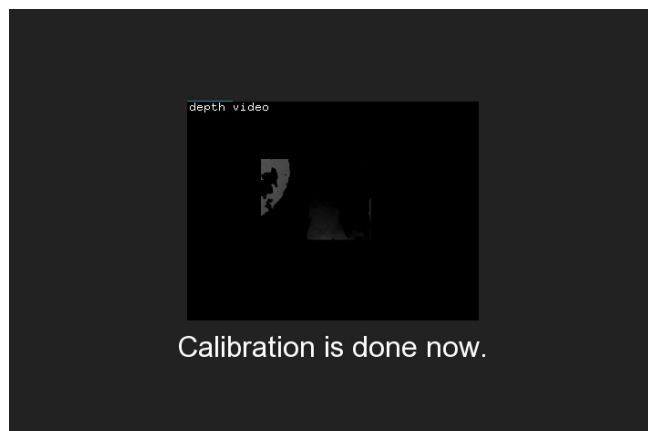
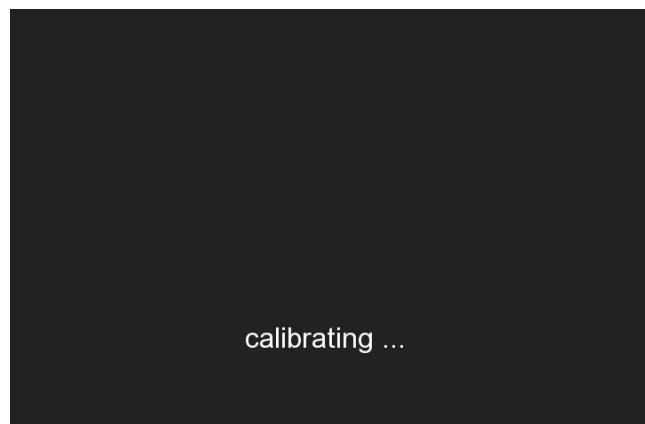
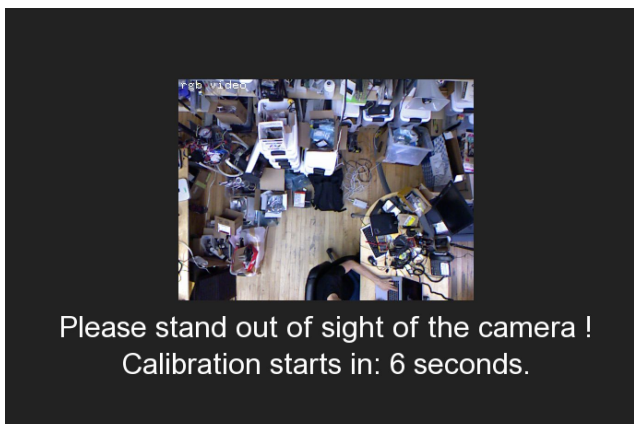
DETAILED TECHNICAL INFORMATION

Normal Software Operation

The software starts automatically after booting is done, or after clicking on the **Spinoza app** in the dock.

When the software starts, it goes through a set-up routine during which people need to stay outside of the camera viewing area.

For a short moment, a live RGB camera view will be visible, then an empty screen (in which the calibration is happening), then a grayscale depth image with the lens tracking graphics, and, finally, the generated lens flares.



Now the piece is operational and the lens flares will react to the rotational orientation of the lens.

When the lens surface is parallel with a wall, the flare will be in full bloom. But, if the lens is rotated 90 degrees from the wall, no lens flare will be visible.

Manual Software Calibration

Not much calibration is needed for this artwork. Please consult the images below.

Ensure that **near + farThreshold** are adjusted so that only the lens is detected. Place your hand below and above the lens to see if they show up in the debug live view's threshold video. They should not be visible, so that people's hands will not affect the tracking.

Depending on the distance between lens and the Kinect, **minCropX,Y,maxCropX,Y** should be adjusted, so that only the necessary area is being used for tracking and all the rest is masked out.

Based on how the three displays are arranged in **System Preferences → Displays**, you might need to change the **displayX_0,Y_0** or **displayX_1,Y_1** or **displayX_2,Y_2** values to place the correct flares onto each display.

Keyboard Shortcuts:

The key **g** on keyboard: graphic user interface pops up. These detailed settings are not really needed.

The key **v** on keyboard: reveals or hides the live depth camera view and other video used for lens tracking.

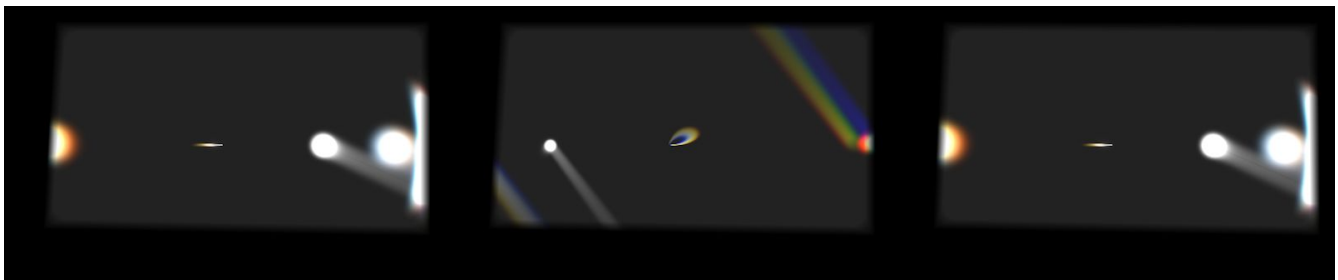
The key **t** on keyboard: reveals or hides graphics that show how the lens is being tracked.

The key **f** on keyboard: toggles between fullscreen mode and window mode.

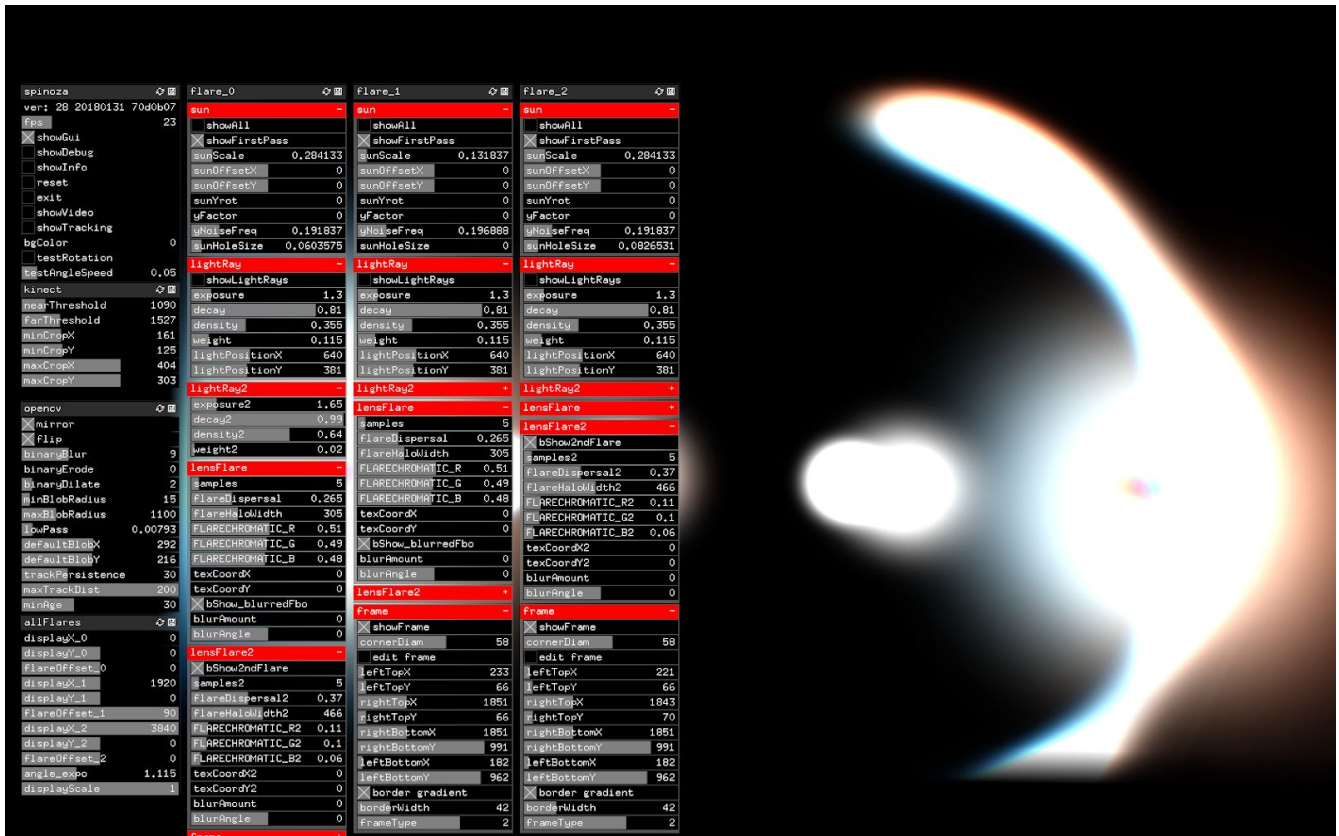
The key **i** on keyboard: shows information about the apps performance.

The keys **1-3** on keyboard: expose additional view on to different steps of the flare generation.

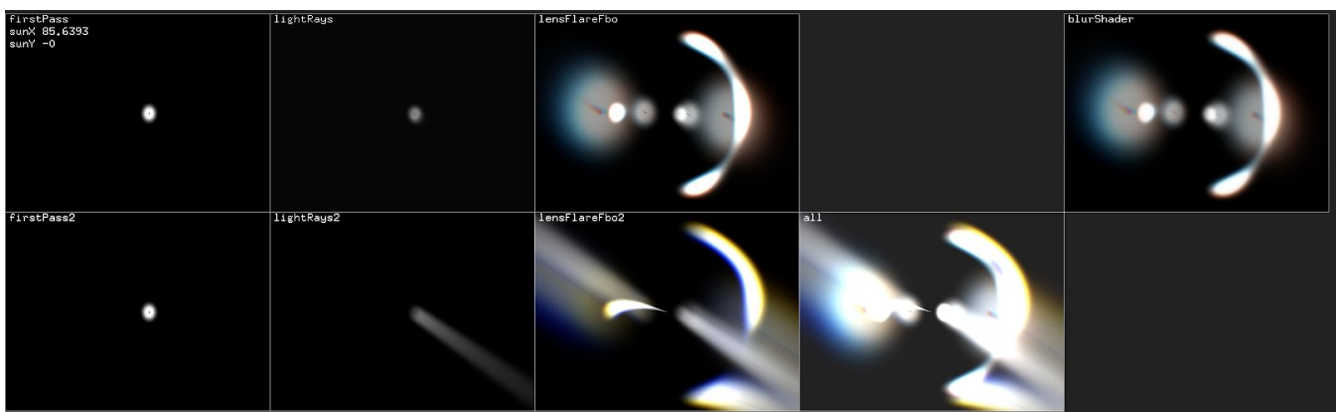
The key **0** on keyboard: hides those additional views.



All three projections with black frame to mask light spill.



All GUI elements.



Additional debug view on to the different steps used to create the lens flares.

Other GUI Elements Used for Setup:

Only when we initially setup the artwork in its location will we need some of these GUI elements.

```
spinoza
ver: 28 20180131 70d0b07
fps 23
[X] showGui
[ ] showDebug
[ ] showInfo
[ ] reset
[ ] exit
[ ] showVideo
[ ] showTracking
bgColor 0
[ ] testRotation
testAngleSpeed 0.05
```

ver: 28: this is an automatically-generated version number. Every time a new git commit is filed, the version number will change.

fps: indicates how fast the app is running.

showGui: displays GUI, like key **g**.

showInfo: like key **i**, it displays info about app performance and detected blobs.

reset: don't use.

exit: exits app; like key **esc** or **command q**.

showVideo: shows a range of live and tracked views.

showTracking: draws tracking graphics, like key **t**.

bgColor: sets background color. Should be 0.

testRotation: if selected live tracking is disabled and a fix rotational speed is applied. For debugging only.

testAngleSpeed: rotational speed for debugging.

kinect	
nearThreshold	1090
farThreshold	1527
minCropX	161
minCropY	125
maxCropX	404
maxCropY	303

near + farThreshold: the distance range from the Kinect, where the lens can be found.

minCropX,Y,maxCropX,Y: select "show video" to see how cropping limits the viewing area of the Kinect. This can be used to exclude the wall and other nearby objects from being seen by the Kinect.

mirror, flip: leave selected.

opencv	
<input checked="" type="checkbox"/> mirror	
<input checked="" type="checkbox"/> flip	
binaryBlur	9
binaryErode	0
binaryDilate	2
minBlobRadius	15
maxBlobRadius	1100
lowPass	0.00793
defaultBlobX	292
defaultBlobY	216
trackPersistence	30
maxTrackDist	200
minAge	30

binaryBlur, erode, dilate: improves tracking-image quality.

minBlobRadius, maxBlobRadius: any blob outside these thresholds gets ignored. Prevents heads or fingers from being tracked.

lowPass: smooth the rotation via a biquadratic filter.

defaultBlobX,Y: in debug mode, a small x is drawn in the live view. It should be set to the lens center.

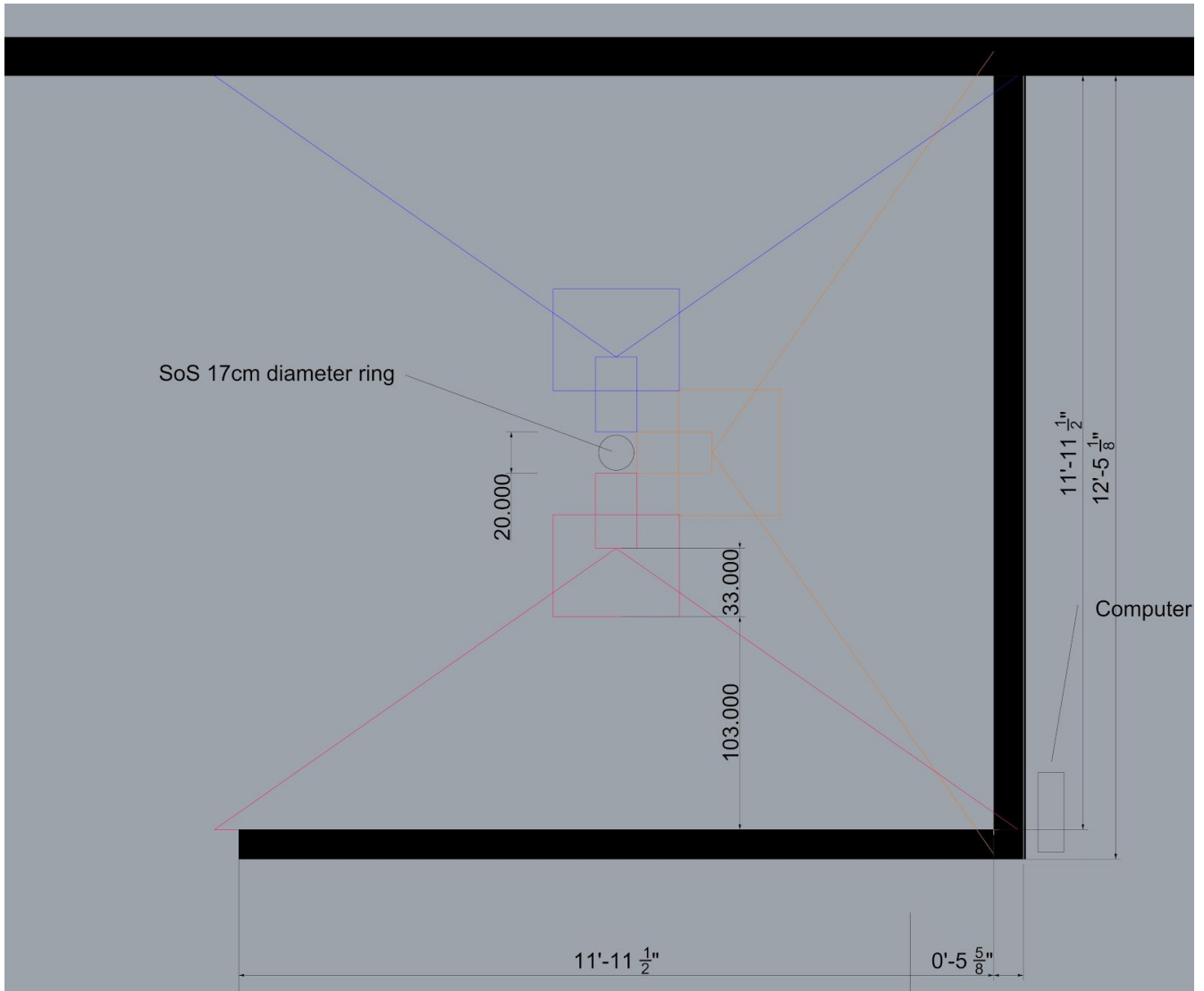
trackingPersistence: how long the app waits for a lost blob to reappear in same location.

maxTrackDist: how far a blob can travel in one frame before it is considered a different blob.

minAge: minimum amount of frames a blob needs to be detected before it is shown.

The Version 2 app can create flares on multiple projectors. Each display gets its own flare. The displays are offset so that they fully cover each display. These flares can get an angular offset so that they only appear if the lens is facing in a particular direction.

In the setup below, three HD projectors are being used. Here, each display is offset by 1920 pixels in the app. Also, the middle wall has a flare offset of 90 degrees.



Three-wall projection.

allFlares	☰
displayX_0	0
displayY_0	0
flareOffset_0	0
displayX_1	1920
displayY_1	0
flareOffset_1	90
displayX_2	3840
displayY_2	0
flareOffset_2	0
angle_expo	1,115
displayScale	1

displayX_0,Y_0: top left corner position of display 0.

flareOffset_0: angular offset to match perpendicular walls.

angle_expos: scales how the real lens rotation is applied to the angle of the projected flares. It is used to keep the flares longer on the screen.

displayScale: for debugging all three displays. Can be scaled down to be visible on one display.

Each of the three displays has their own sun, flare, light ray, and frame settings. This allows us to make each wall slightly different from the other.

flare_0	☰
sun	-
<input type="checkbox"/> showAll	
<input checked="" type="checkbox"/> showFirstPass	
sunScale	0,284133
sunOffsetX	0
sunOffsetY	0
sunYrot	0
yFactor	0
yNoiseFreq	0,191837
sunHoleSize	0,0603575

showAll: like the keys **1-4**, it shows all stages of the flare creation.

showFirstPass: draws the small white sun dot which is used to generate all other flare effects.

sunScale: determines the size of sun disc.

sunOffsetX,Y: origin of sun at angle 0.

sunYrot: rotates the sun disc around its vertical axis.

yFactor: adds a subtle change of the sun's Y position over time.

yNoiseFreq: determines how much the Y changes over time.

sunHoleSize: creates a hole inside the sun.

Each flare is created by adding two light ray effects. The GUI elements shown here influence the look, the size, and the color range of these effects.

lightRay -	
<input type="checkbox"/> showLightRays	
exposure	1.3
decay	0.81
density	0.355
weight	0.115
lightPositionX	640
lightPositionY	381
lightRay2 -	
exposure2	1.65
decay2	0.99
density2	0.64
weight2	0.02

lensFlare -	
samples	5
flareDispersion	0.265
flareHaloWidth	305
FLARECHROMATIC_R	0.51
FLARECHROMATIC_G	0.49
FLARECHROMATIC_B	0.48
texCoordX	0
texCoordY	0
<input checked="" type="checkbox"/> bShow_blurredFbo	
blurAmount	0
blurAngle	0
lensFlare2 -	
<input checked="" type="checkbox"/> bShow2ndFlare	
samples2	5
flareDispersion2	0.37
flareHaloWidth2	466
FLARECHROMATIC_R2	0.11
FLARECHROMATIC_G2	0.1
FLARECHROMATIC_B2	0.06
texCoordX2	0
texCoordY2	0
blurAmount	0
blurAngle	0



Frame-editing mode turns the black frame red and draws additional info.

In some installations, the projectors might not be installed perfectly parallel to the projection walls or the wall might not be perfectly rectangular. In these cases, the app allows us to draw a black frame around the projected flares.

This frame can also be used to make the edges blurry and less sharp. It might help to change **bgColor** to something lighter to see the difference better.

frame	
showFrame	
cornerDiam	58
edit frame	
leftTopX	0
leftTopY	1
rightTopX	1917
rightTopY	0
rightBottomX	1915
rightBottomY	1199
leftBottomX	1
leftBottomY	1199
<input checked="" type="checkbox"/> border gradient	
borderLeft	10
borderRight	0
borderTop	10
borderBottom	10
frameType	2

showFrame: enables use of the frame feature.

Select **frameType 0** to not use any frame or type 2 to get the above shown gradient.

Select **edit frame** to alter the positions of the frame corners. The black frame will turn red.

Now you can change the X and Y location of each corner until you get a projected image that is aligned well with the wall. Use the keys **l, k, j, h** to pick which corner you want to adjust and use the arrow keys to fine-tune the adjustment.

Use the slider for more abrupt adjustments.

By selecting **border gradient** and adjusting **borderLeft, right, top, bottom**, you can decide whether your frame needs a gradient effect or not and how wide the gradient border is on each edge.

Remote Access to Artwork's Computer

There is a software installed on the computer running this artwork that allows the studio to connect remotely to the artwork. This feature is helpful when you require assistance from the studio, as we can remotely connect to it, do a quick inspection, and do a debugging session of your components, if needed. In order to enable this feature, the computer has to be connected to the internet at all times. Depending on the computer's operating system (Windows 7/8/10, OSX), the procedure to set the computer online will vary. Please look online for tutorials, if necessary.

Preliminary Troubleshooting Steps

After pressing the button, nothing seems to happen.

Do you hear any sound coming from the computer? If so, the computer is running and the projector should display the piece shortly. If not, check if the projector is powered correctly and try to turn it on with the remote control. Also, check that the projector's source is set to the same port where the cable is plugged in (HDMI, VGA, DVI, etc.)

The piece doesn't react when the lens is rotating.

Ensure that the Kinect is well connected to the power and to the computer. You might be able to see a light inside the Kinect lenses.

The projected 3D shadow is not reacting or is not visible.

Connect a USB mouse and a keyboard to the USB port on the small box. Press the ESC button to exit the software. Find the app's icon in the dock and restart the app.

The motor that rotates the lens is too loud.

There is a cable that runs from the motor to a small black box. This box also has two USB cables entering it. One USB cable connects the box to the computer and the other connects to the projector. Disconnect the USB that runs to the projector. This will shut down the power to the motor.

Upon startup of the app, the RGB video feed is black.

This means that the Kinect camera was not initialized correctly. The software is programmed to automatically try to reconnect to the Kinect camera. If this was successful, you should see the RGB video. If not, use the "restart" button in the GUI to restart the softwares. Or, shut the computer down and restart it.

The lens fell off.

If the glass lens broke or chipped please buy a new one. See the [Appendix](#) for details on the brand and model required for replacement.

Use new fishing wire and feed it through the two holes in the lens mounting ring. Ensure that the distance between the lens and the Kinect camera mount is the same as before.

Troubleshooting Assistance

Prior to contacting the Antimodular Studio with a problem about your artwork, please ensure that you went through the preliminary troubleshooting steps outlined in the previous section.

The troubleshooting process will vary depending on the problem. In order to make the process easier, it is recommended that you collect and send the following information to the studio:

- Date and time when the problem first happened;
- Description of the problem;
- Actions taken so far and conclusions;
- Detailed photographs (or videos) displaying the problem;
- Detailed photographs (or videos) of the suspected faulty component;
- Detailed photographs (or videos) of the whole artwork and its surroundings;
- Personnel involved.

Support (Contact Us)

If you would like support for the piece, please feel free to call Lozano-Hemmer's studio in Canada:

Antimodular Research
4060 St-Laurent, studio 107
Montréal Québec H2W 1Y9 Canada
Tel 1-514-597-0917
Fax 1-514-597-2092
info@antimodular.com
www.antimodular.com

APPENDIX I - INSTALLATION

Description of Components

This artwork requires the following components:



For the single-projector version: Computer, Apple Mac Mini, i7, 3 Ghz, 4GB RAM



For the multi-projector version: Hackintosh, i7, 4Ghz, 16 GB RAM, GTX1070
with macOS 10.12.6, 250 GB hard drive



Microsoft Kinect 360 camera



USB 2.0 over Cat5 extender for Kinect-Computer connection, Black Box IC280A-R2 (only needed for Version 2)



www.pololu.com

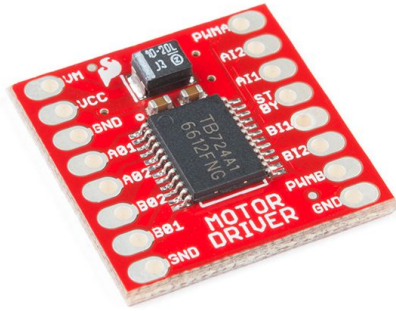
25D mm brushed DC motor, 499:1 ratio



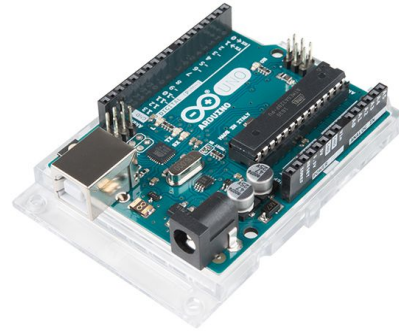
www.pololu.com

1000:1 Micro DC motor

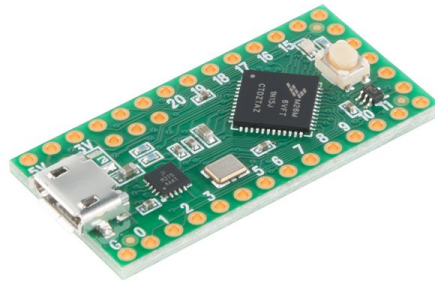
Metal geared motor, 5 Volt DC, 11-55 RPM



TB6612FNG DC driver from Sparkfun.com



Arduino Uno



Teensy LC

Motor driver TB6612FNG controlled by Arduino or Teensy

Projector Hitachi cp-aw251n with remote HL02772

Projector-computer control for Hitachi based on rs232 level shifter, digikey 28560D-ND



Panasonic projector + ET-DLE030



Epson projector + ELPLX01

Projector with Ultra Short Throw Lens (for Version 2)

Active USB extension cable for all other long USB connections



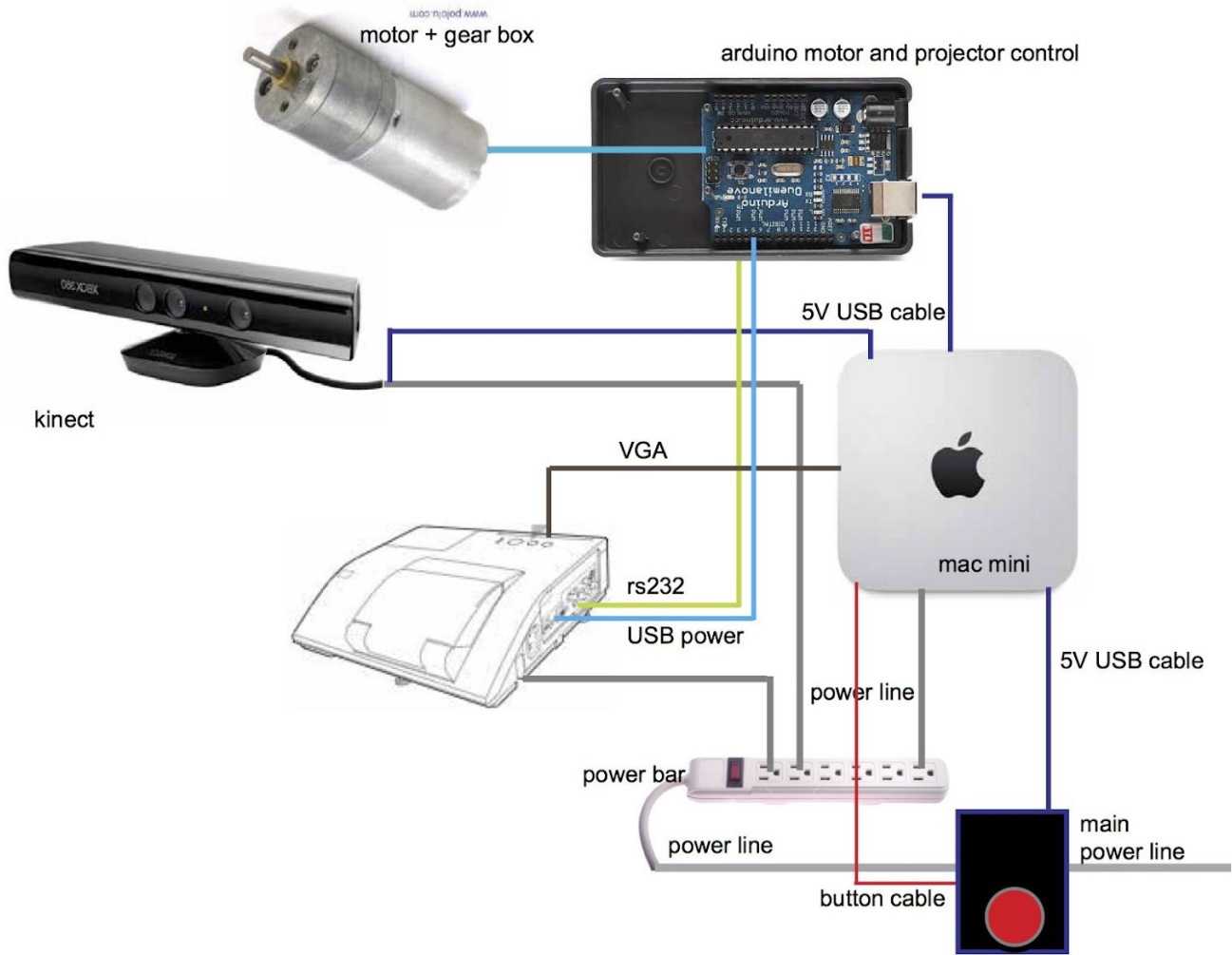
Logitech wireless RF keyboard



150mm Dia x 300mm Focal Length, PCX Condenser Lens
(www.edmundoptics.com Stock #27-509)

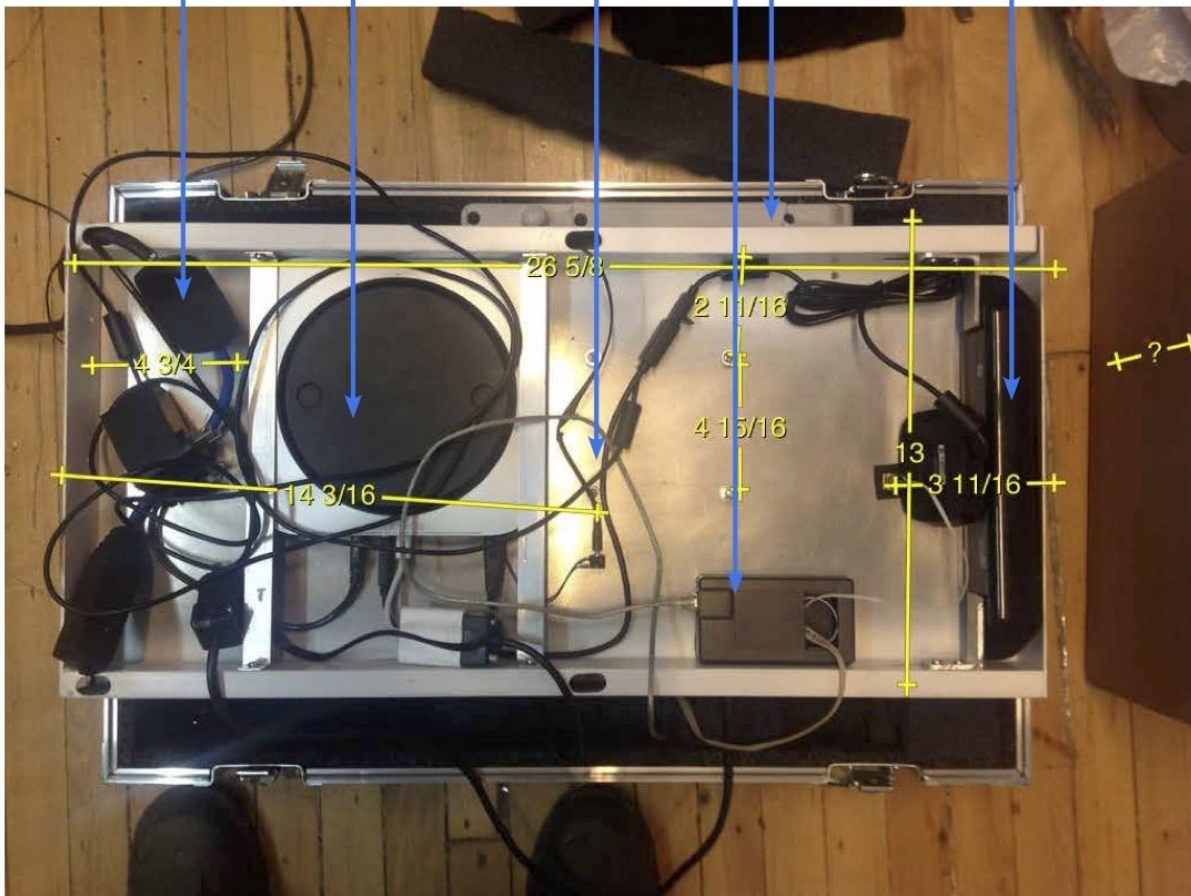
Wiring Diagrams and Connections

In order for the piece to run properly, the computer should be connected according to the following diagrams.



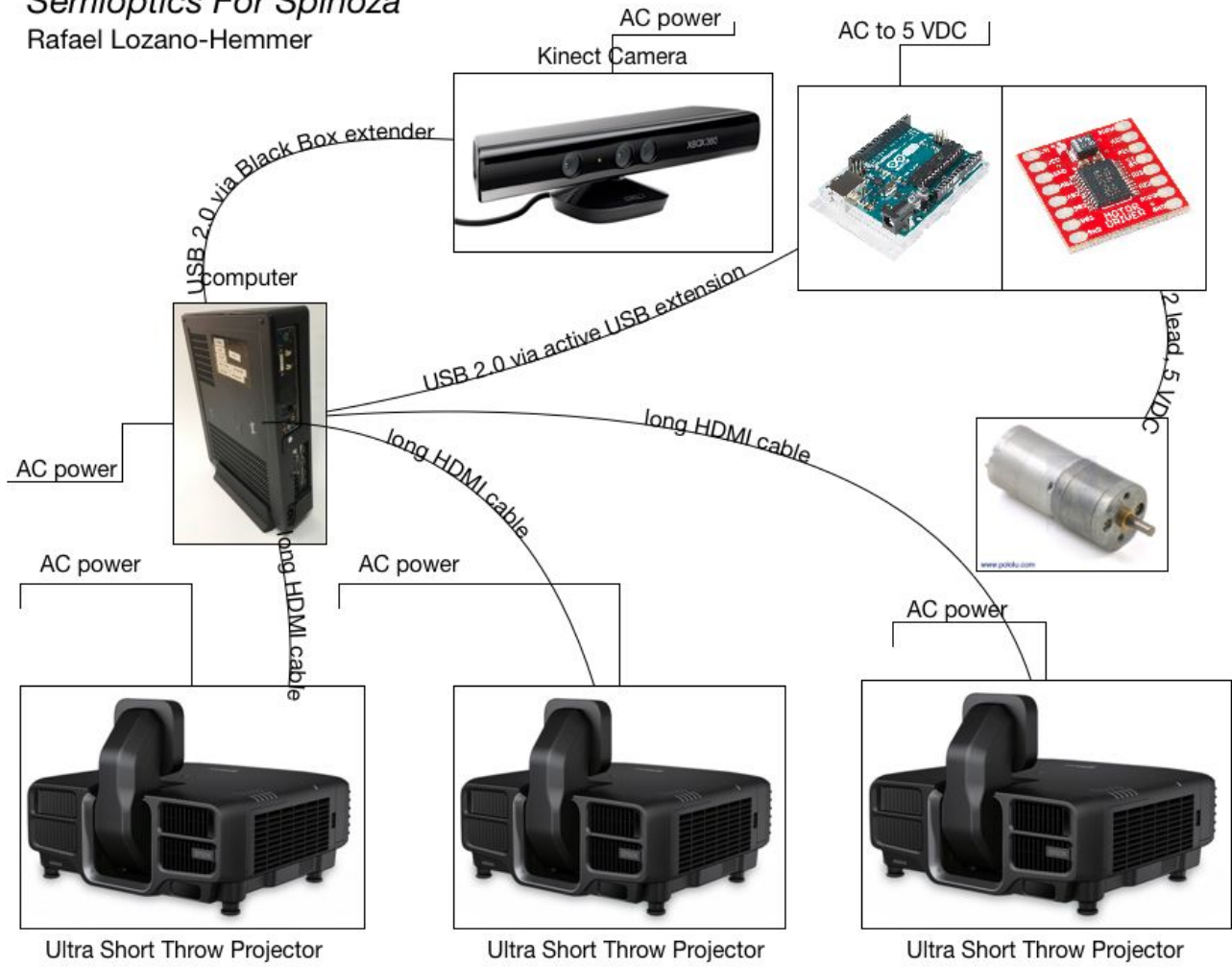
General setup of all the equipment for Version 1

power lead connection
 mac mini computer
 external power button connection
 arduino motor + projector control
 projector
 kinect camera



Wiring of Version 1 with all the equipment integrated onto one bracket.

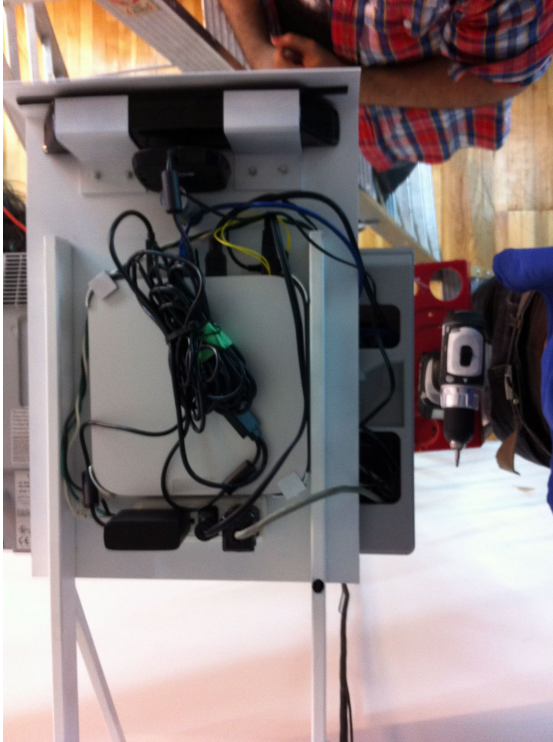
Semioptics For Spinoza
 Rafael Lozano-Hemmer



Wiring diagram multi-projector version (Version 2)

Close-Up Photos of Mounting

The following are images of the all-integrated mounting version.



All equipment is mounted onto the bracket.



Six screws hold the bracket in place.

APPENDIX II - TECHNICAL DATA SHEETS

Microsoft Kinect 360

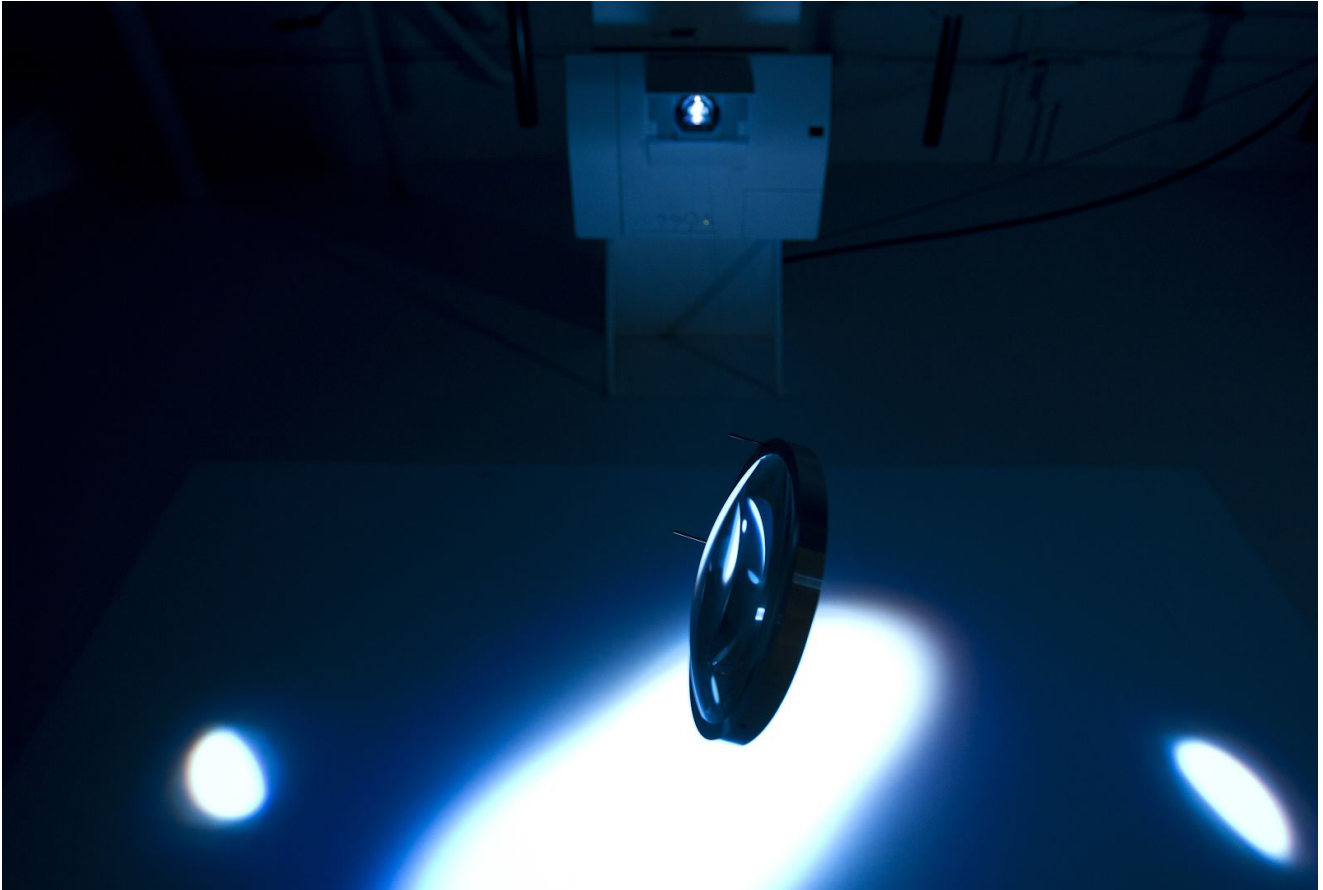
Color Camera	640 x 480 32-bit colour at 30 frames/second
Depth Camera	320 x 240 16-bit depth at 30 frames/second
Minimum Depth Distance	1.2 m
Maximum Depth Distance	3.5 m
Horizontal Field of View	57 degrees
Vertical Field of View	43 degrees
USB Standard	2.0

Installation Photos

Version 1: Single-projector version



Version 1: Single-projector version



Version 2: Multi-projector version

