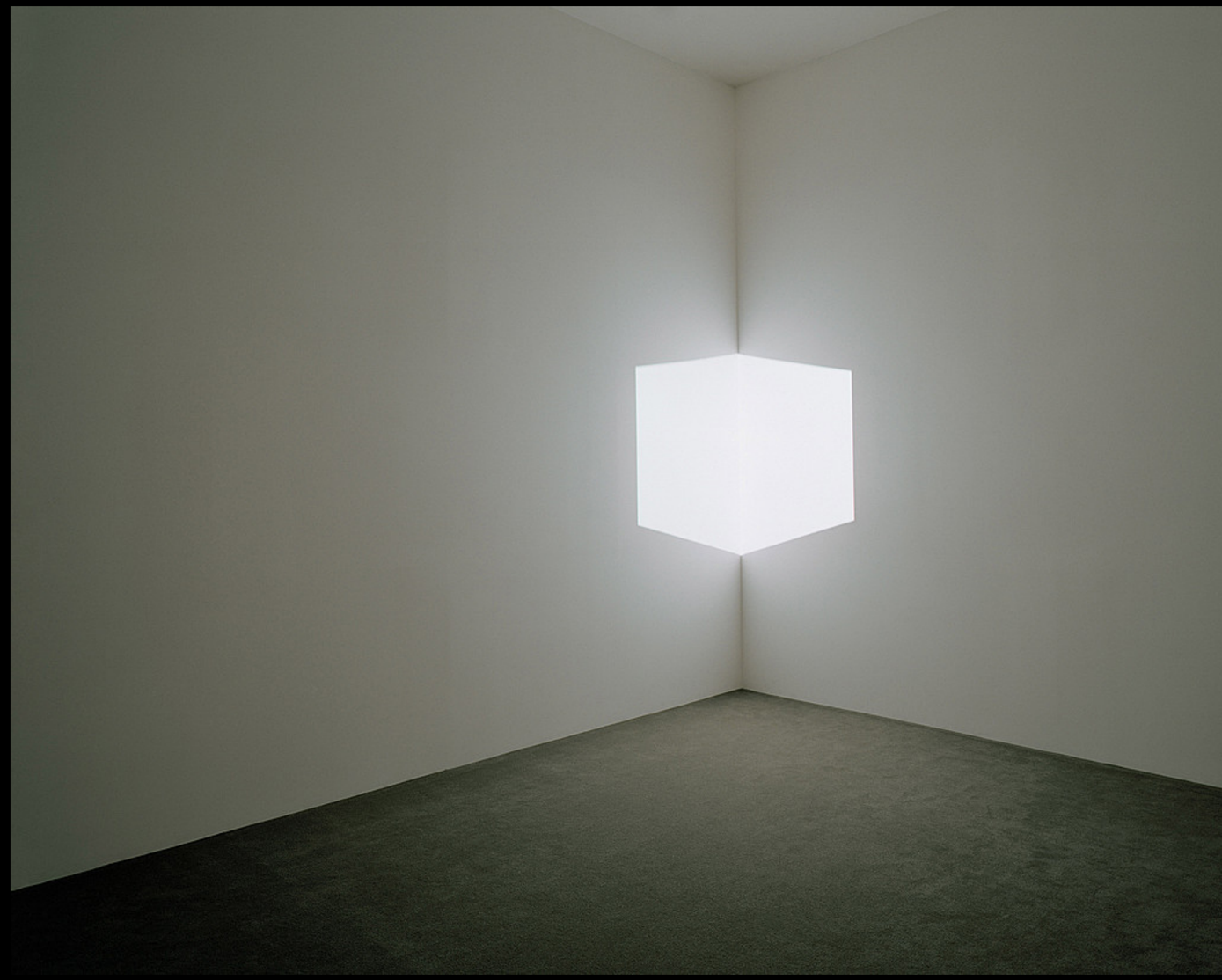
IMCA 221
Programming for Artists
Winter 2025

lee wilkins
l.wilkins@concordia.ca

Class is on Zoom even in the classroom for sharing, find the details on Moodle https://moodle.concordia.ca/

Download the slides!

James Turrell
Afrum 1

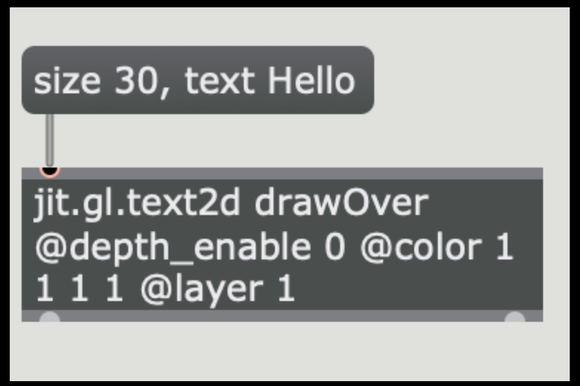


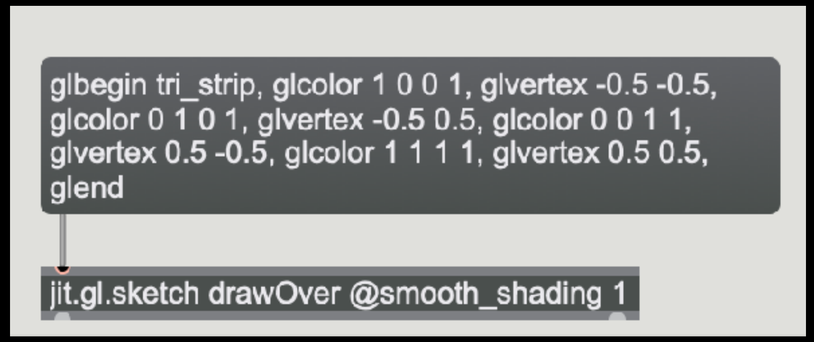
2d drawing and drawing over video

There are a number of ways to draw in 2d and lots of opinions! Jit.lcd is a very common object but may not be supported for much longer. jit.sketch is a bit harder to use, but more likely to be supported for longer.

Camera draw patch

Here are some examples of drawing or writing text. Check out details here https://docs.cycling74.com/max7/tutorials/jitterchapter40

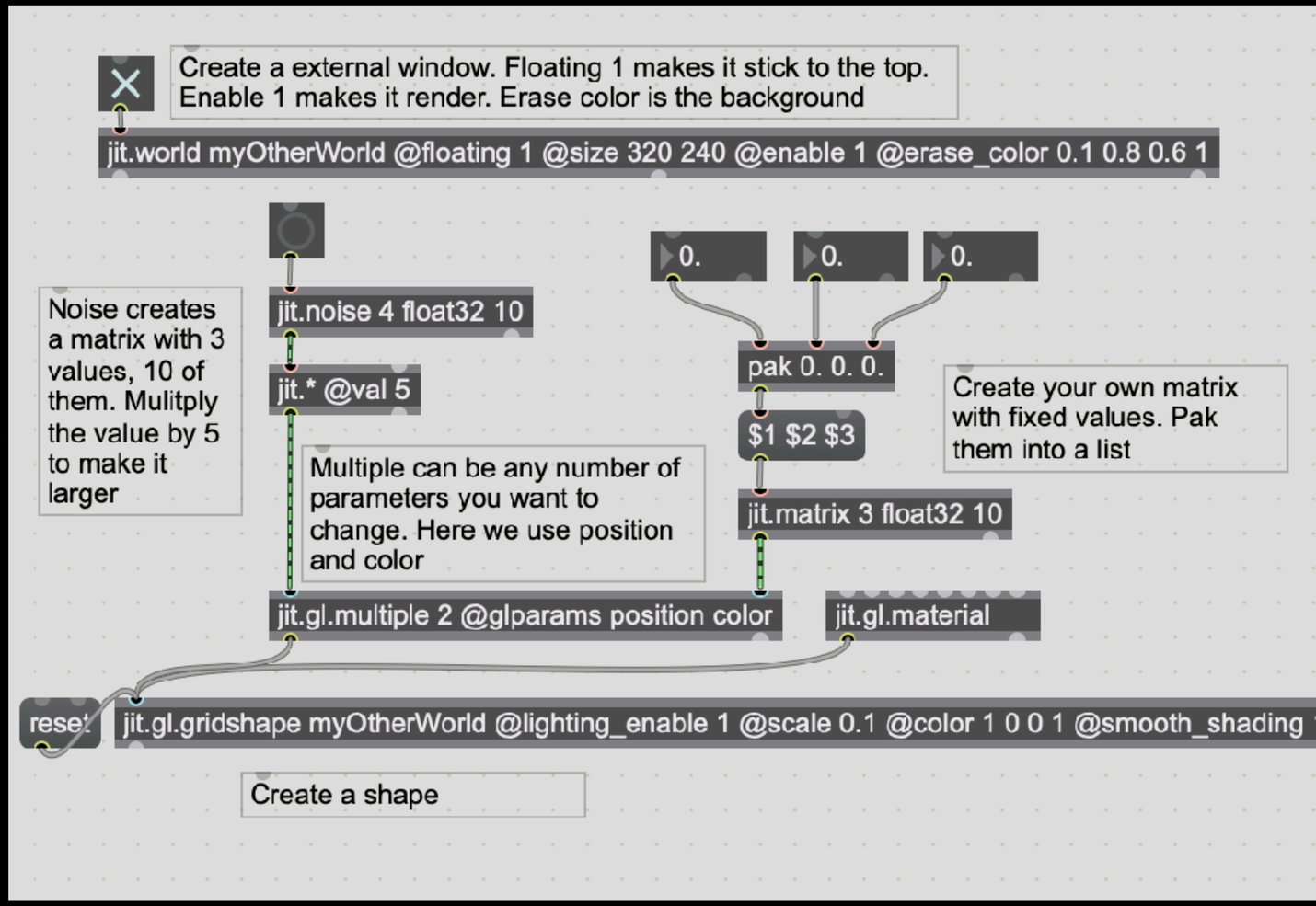




3D Drawing patch

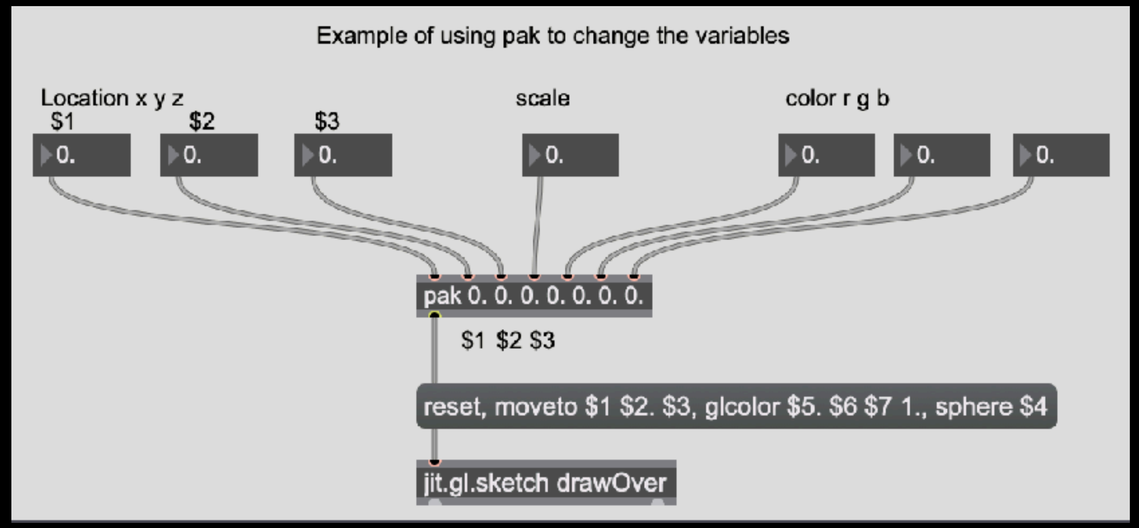
jit.gl.multiple is a way of creating many shapes. You can set how many parameters you want to change and then use a matrix to send the information. Here's an example using noise.

Explore this concept more - https://www.youtube.com/watch?v=kCNbW-tahuc



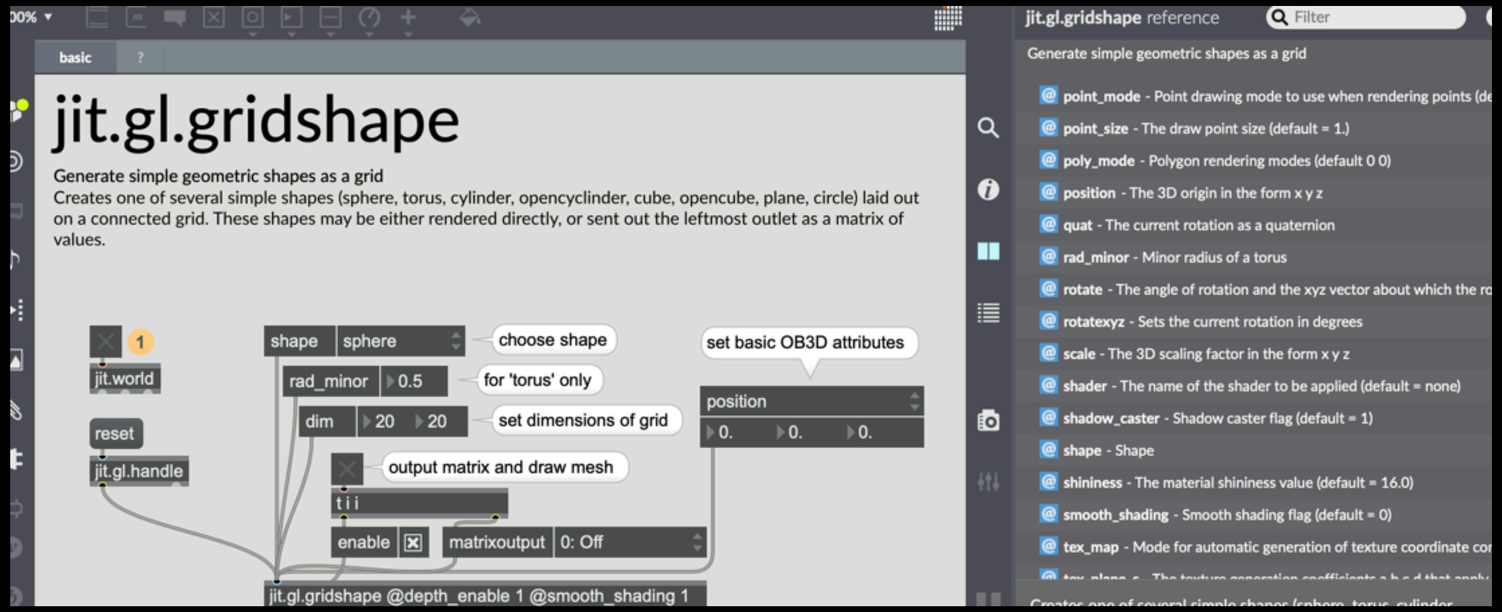
Camera draw patch

Here are some examples of paking a list of elements to change the size and position of a sphere



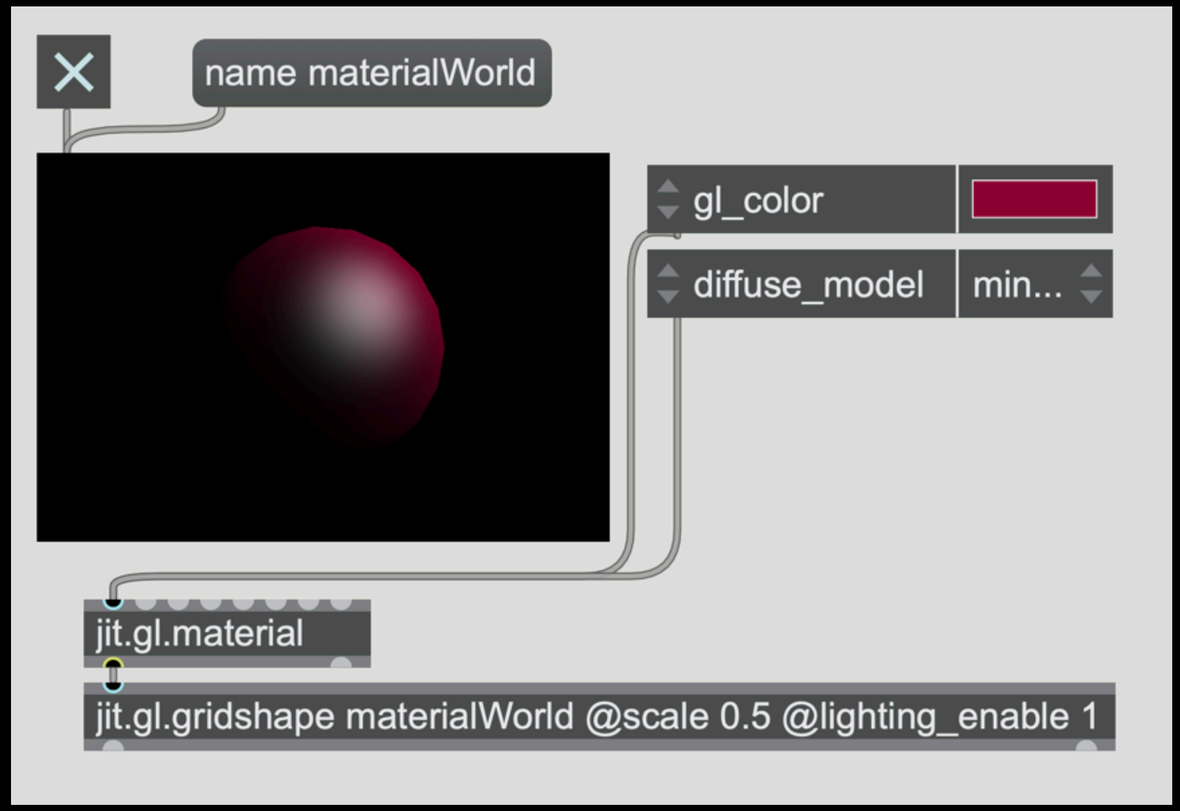
3D Drawing patch

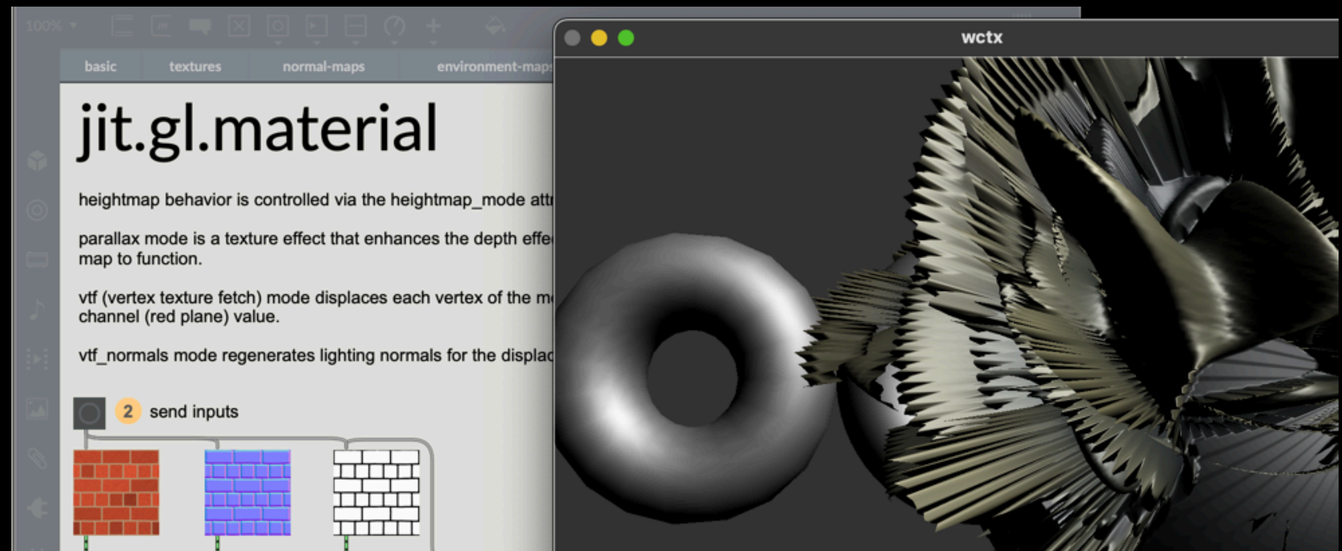
Check the help file to fully understand what you can do with this and begin to explore.



3D Drawing patch

You can add materials to your shapes with jit.gl.material. There are many attributes you can change to play with materials. Check out the material help file for so so much more!



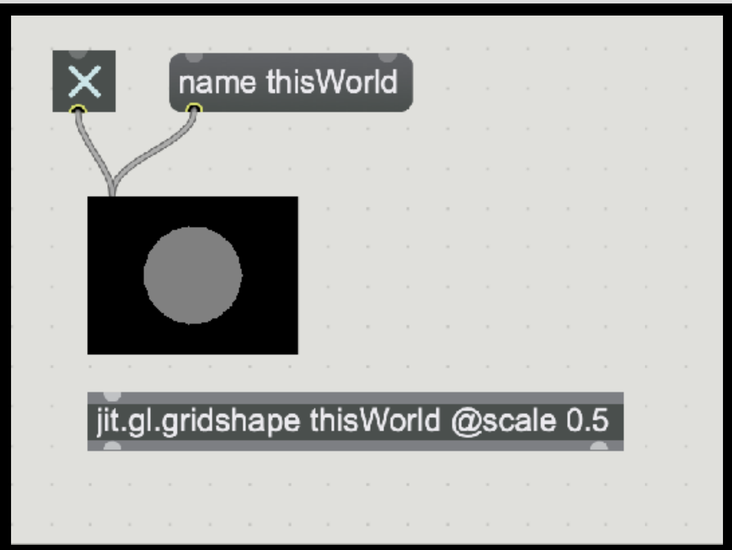


3D Drawing patch

There are a number of tools to make drawing in 3D easier. They begin with jit.gl for graphic language.

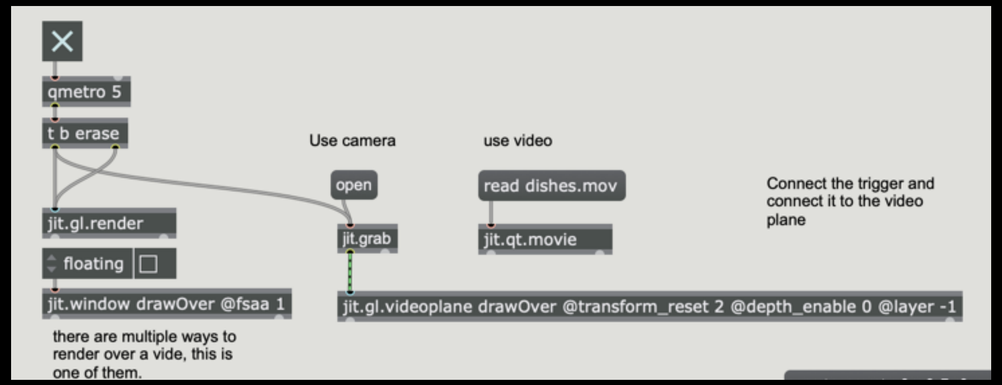
Lets start by making a pworld or jit.world. We need to give it a name so we can reference it. I've named mine "shapeWorld"

jit.gl.gridshape is used to make simple shapes. You can use attrui to change some parameters

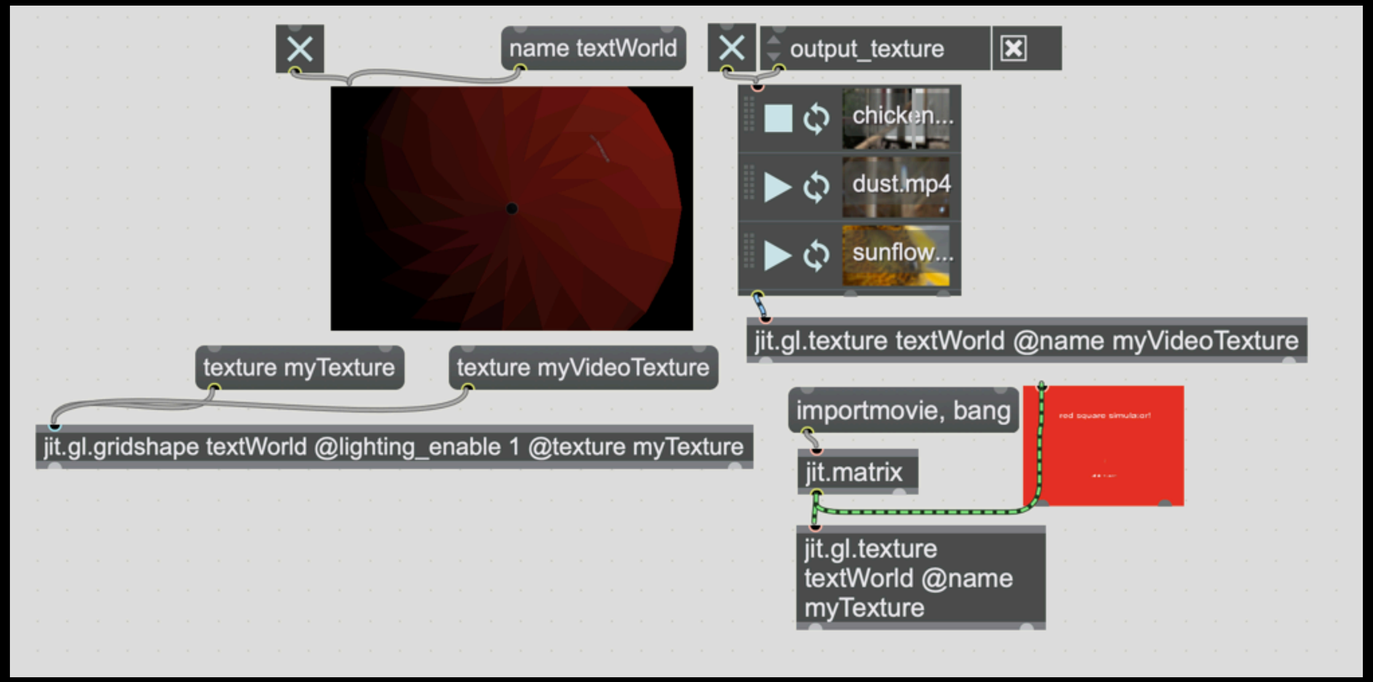


cameradraw patch

Draw over a video using jit.videoplane like this, you need to keep rendering the video



3D Drawing patch
You can add textures as movies or images.

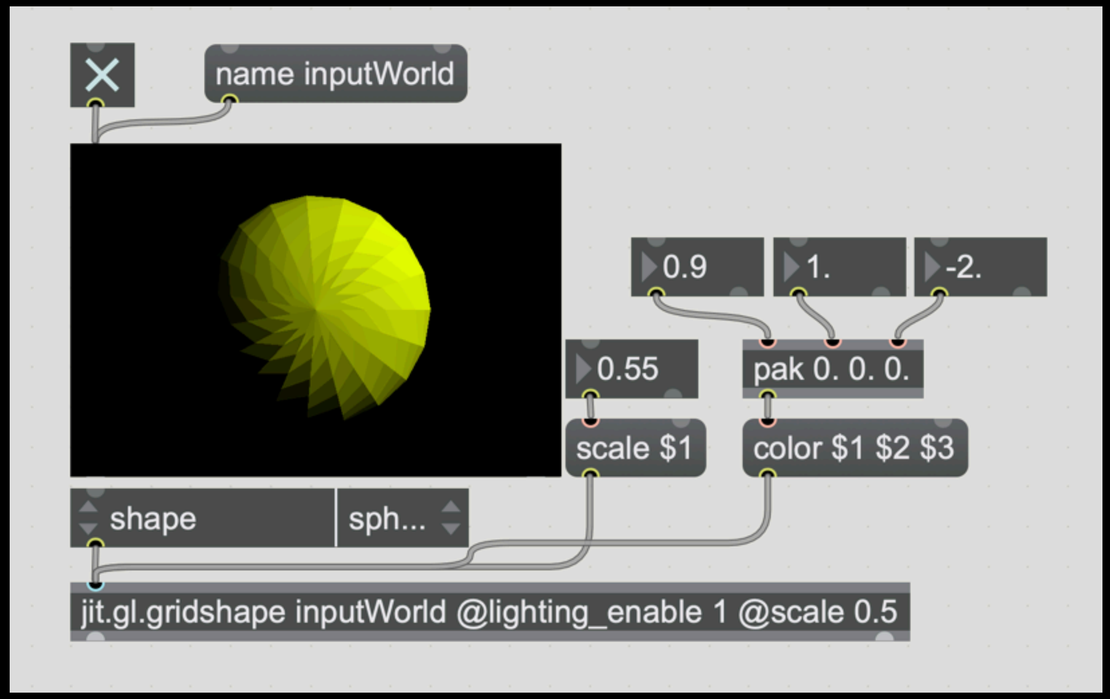


3D Drawing patch

Here is an example of using inputs (music? Sensors?) to change the attributes of a shape.

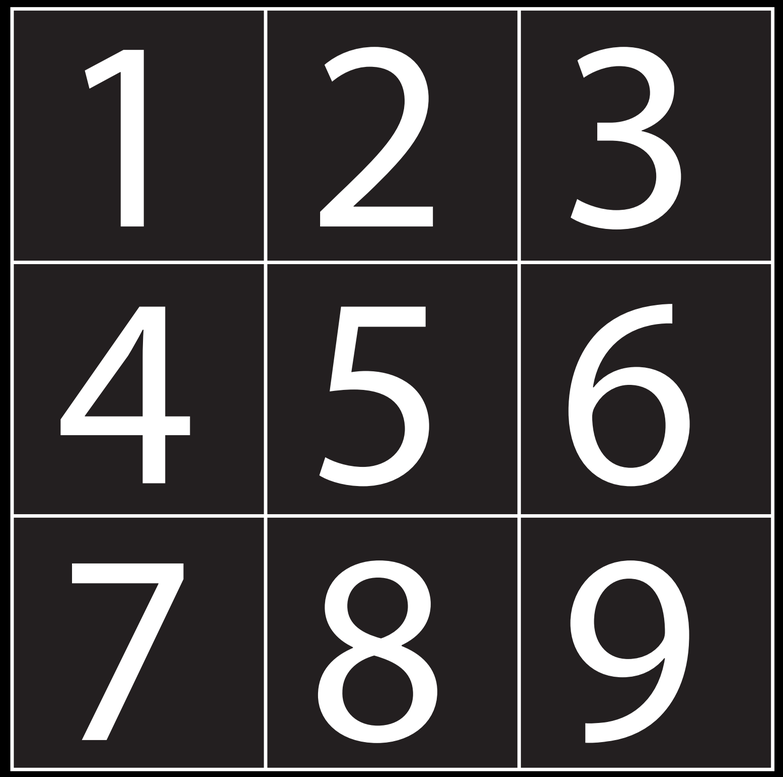
Using $1 holds a variable. We can use pak (or pack) to make a list of variables we can access. For example, color has 3 arguments (red, green blue) so we need to pack a list of 3 elements and then access them

.



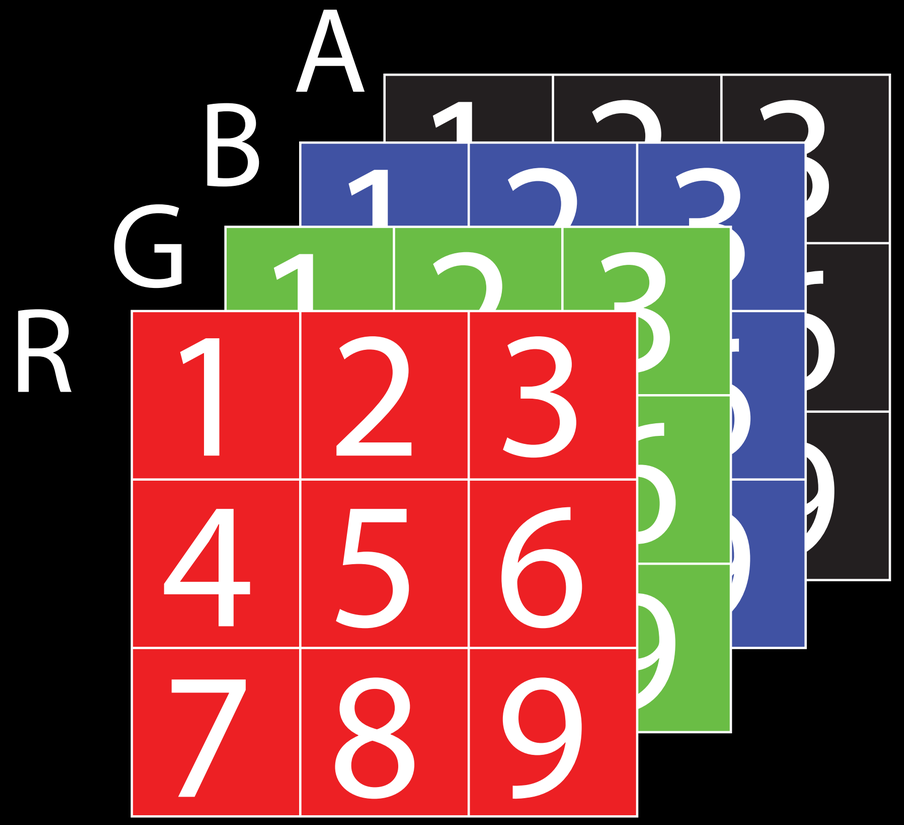
Matrix

Jitter uses a matrix to display graphics. Matrices are grids with cells. Each cell is labeled like this. It is not like a cartesian plane.



Matrix

Each matrix can have multiple planes. Planes are basically maps that describe each pixel. There are 4 here: RGBA, Red, Green, Blue, Alpha. Alpha is transeparcy. They are



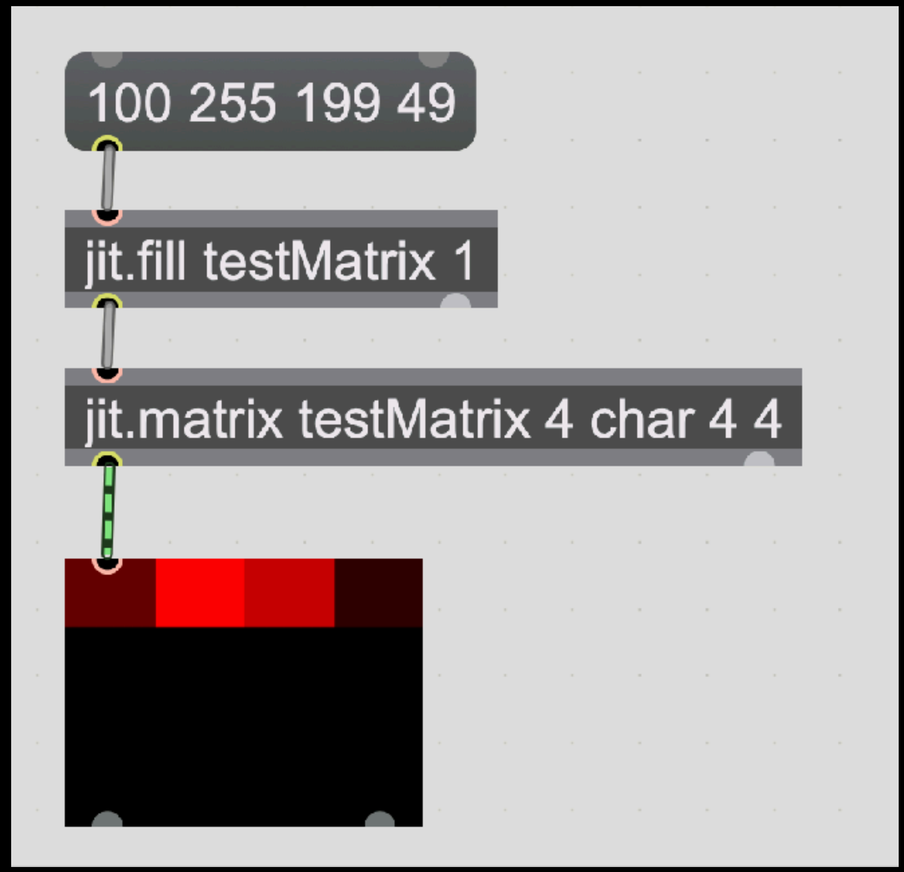
Simple Matrix

You can name each matrix. I created a 4 plane matrix that has 4 x 4 pixels.

I used jit.fill to fill the first plane of my matrix (red)

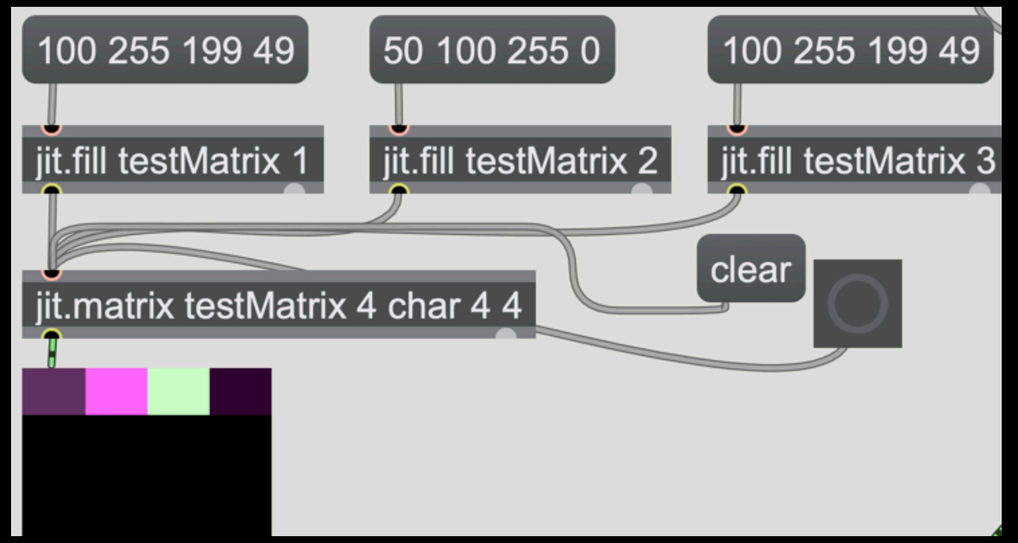
A message with a list of numbers fills each pixel between 0 - 255.

You can also use floats between 1. And 0. But dont forget to use the .



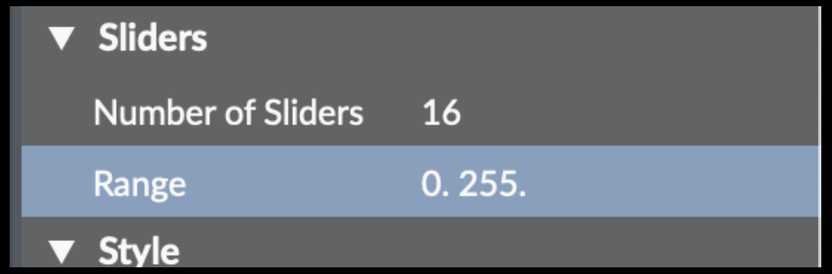
Simple Matrix patch

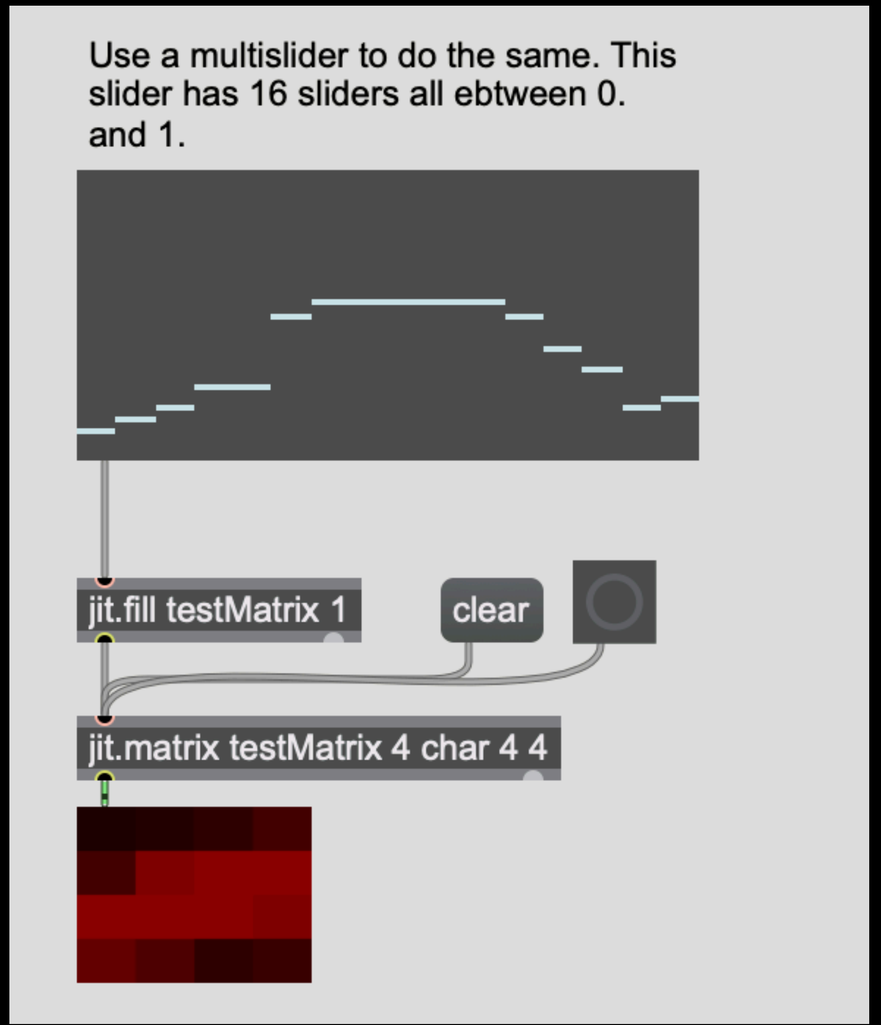
Repeat each plane.

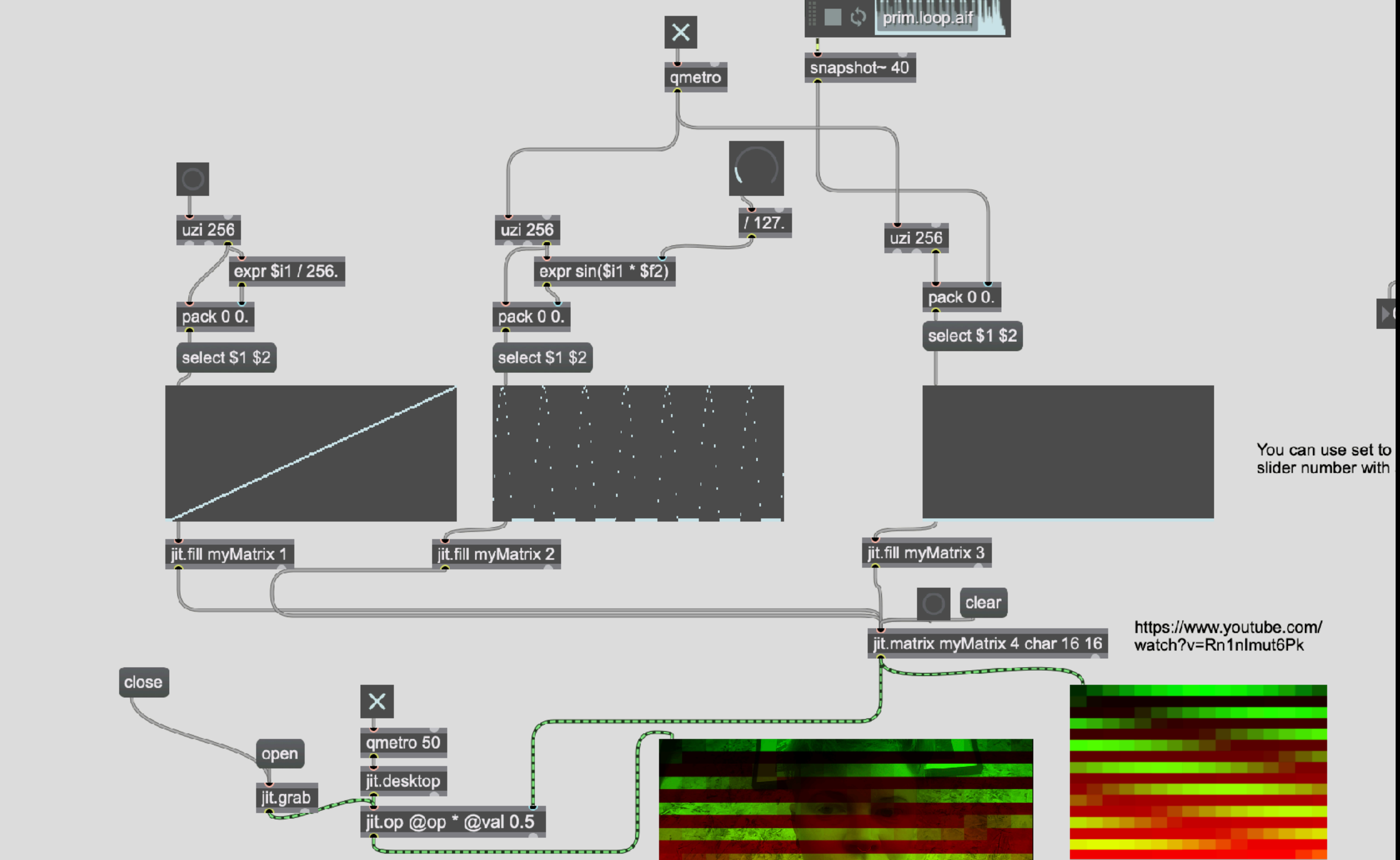


Simple Matrix

You can assign these through a multi slider too. Create a multi slider object and in the inspector, make the sliders 16 and the range 0 255






prim.loop.aif
qmetro
snapshot~ 40
uzi 256
/ 127.
uzi 256
uzi 256
expr $i1 / 256.
expr sin($i1 * $f2)
pack 0 0.
pack 0 0.
pack 0 0.
select $1 $2
select $1 $2
select $1 $2
jit.fill myMatrix 1
jit.fill myMatrix 2
jit.fill myMatrix 3
You can use set to
slider number with
clear
jit.matrix myMatrix 4 char 16 16
https://www.youtube.com/
watch?v=Rn1nImut6Pk
close
open
qmetro 50
jit.desktop
jit.grab
jit.op @op * @val 0.5

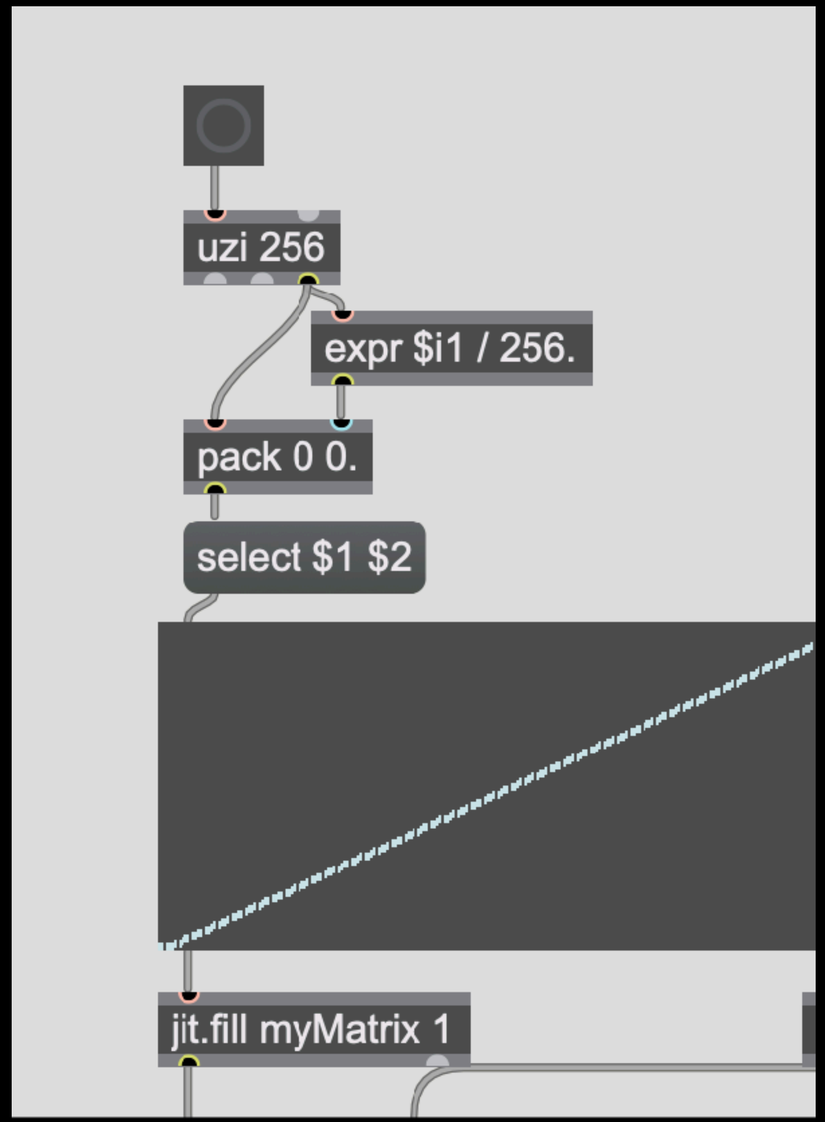
Matrix Patch

You can use the uzi object to output many bangs. We're going to output 256 for the size of our matrix. We will then pack them into a list. The list will contain each the number of the bang and that number divided by 256, which makes a ramp!

So each list lookss like
1 0.0093
2 0.0078
3 0.0117
4 0.0156
Etc all the way up to 256!

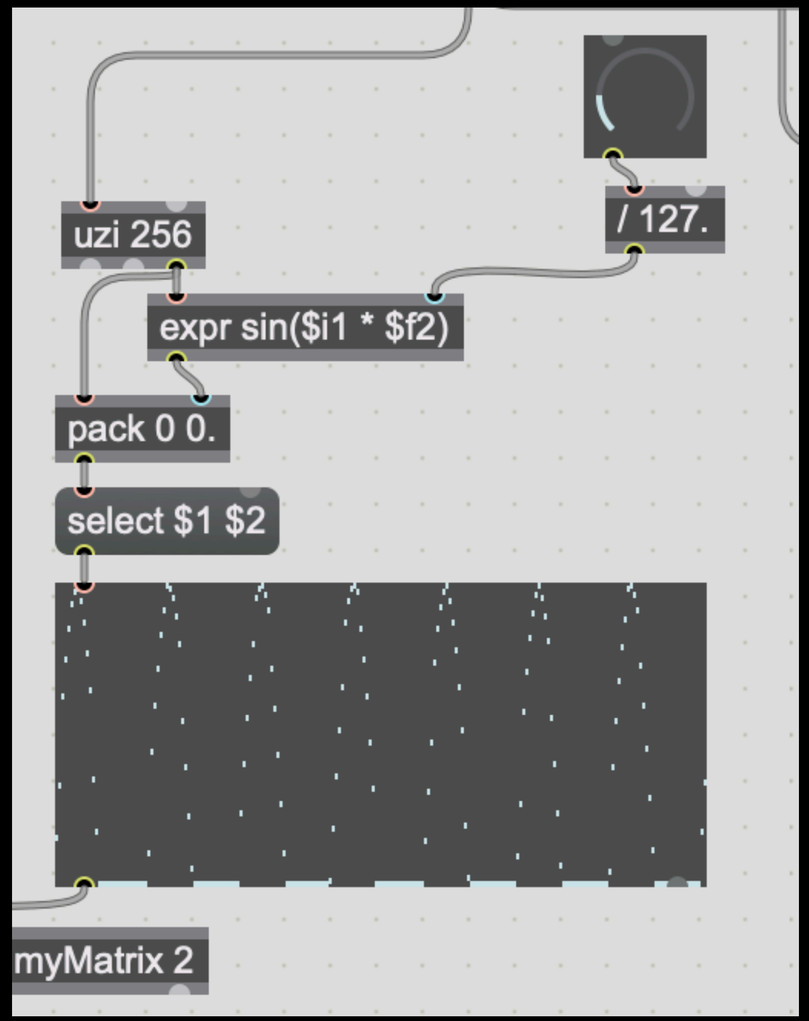
Select grabs each slider (first item) and assigns in second value value.

That value gets sent to the matrix.



Matrix Patch

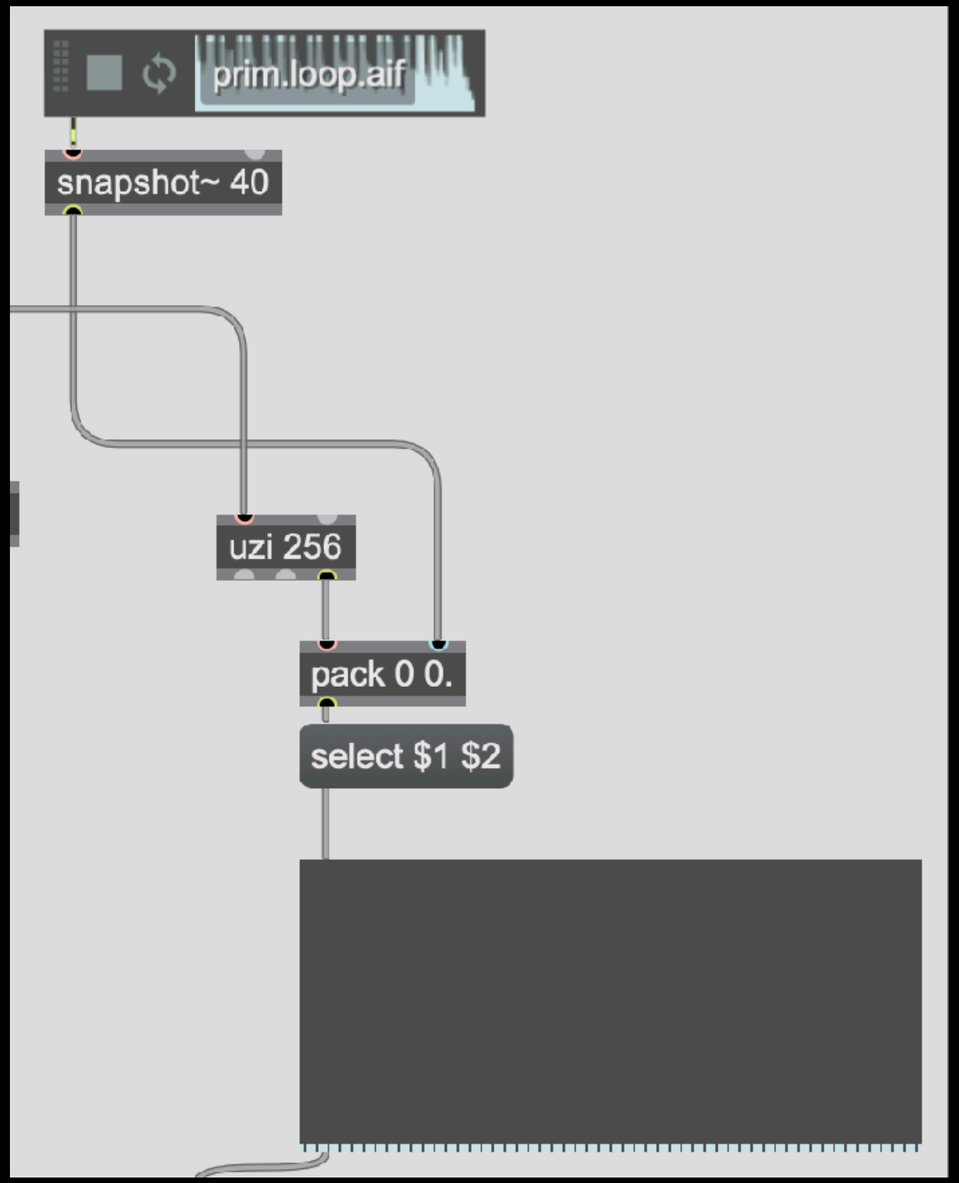
In the second example on the green plane we use a value from a dial to multiple a sin wave. Here we get arch, play with the dial!



Matrix Patch

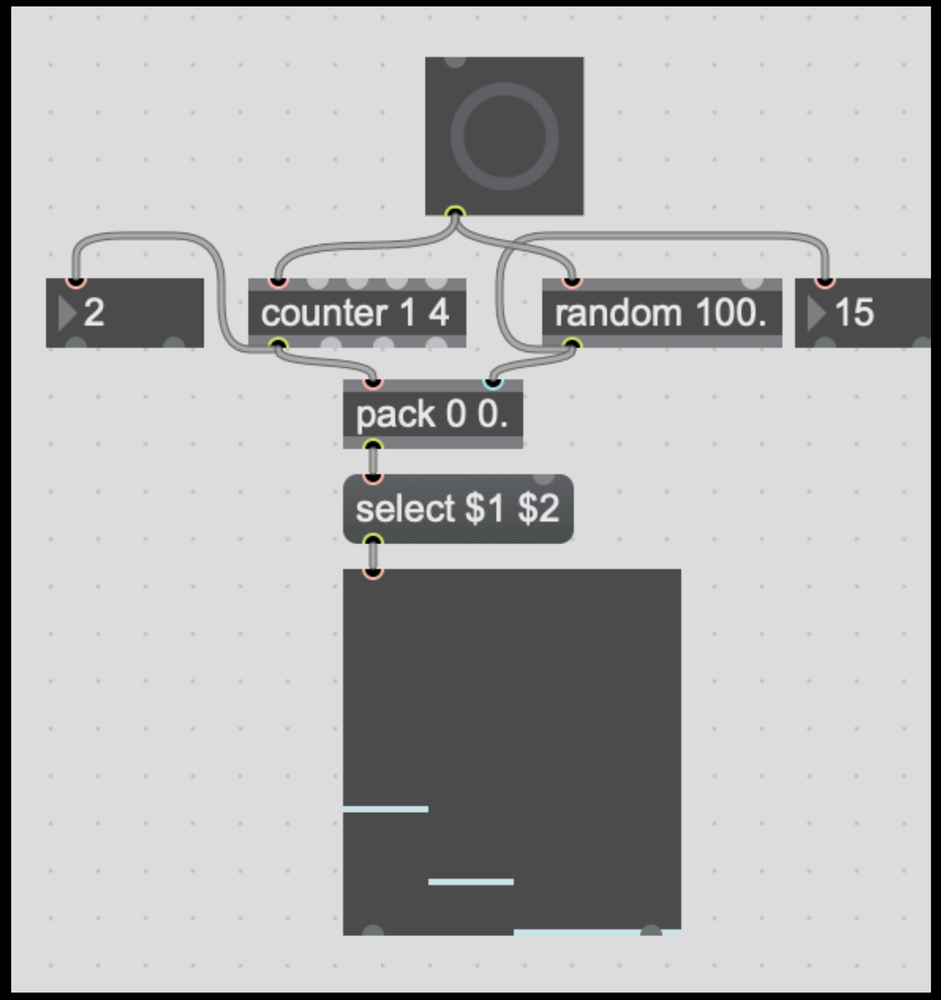
In the third example on the green plane we use the signal output from an audio playlist object. Snapshot is used to convert the signal into a number.

Try hooking up the snapshot output to the sin wave!



Matrix Patch

To understand pack a bit better, we can use count to set each slider with a random value. $1 is the slider, $2 is the value. Slider is coming from count, value is coming from random. They are being packaged and sent to the slider.



Explore More There are tools to draw in 2d:
LCD Object https://music.arts.uci.edu/dobrian/maxcookbook/basics-drawing-lcd-object and https://www.youtube.com/watch?v=QH6eAg2_2vU&list=PLWNLM73RF6jv_QwNymCJJOwNc1RJ9wz9K&index=2

Depth and layers in max https://www.youtube.com/watch?v=7aBEXs-2tEs

More Open GL in Max https://www.youtube.com/watch?v=HORv7Ljd2gQ&list=PLRc5WfOZXC4ITij-h2bRQC4x6mDDkehKw (this whole series is great!)

More Jitter https://docs.cycling74.com/max8/vignettes/video_and_graphics_tutorials_topic

Shaders https://www.youtube.com/watch?v=EO5PSzC8vfM

Sign up for a critique slot

Documentation is due midnight of critique day (after class)

https://docs.google.com/spreadsheets/d/1tYaLwQvdHKLPFb9jIBjtOkV2yuIljyn--IB8jXvJ52s/edit?gid=0#gid=0

Site Specific Installation

Site specific means that your work is linked to the place it is installed. This can mean architecture or form, social or cultural meaning, feelings tied to a space, things the space is used for.

A good site specific installation doesn’t work as well in another location. If you want help figuring it out, let me know.

You can set up in a public place or private space. Studios are fine as long as it is responding in some way directly to the space.

Site Specific Installation
25% Exploration & Functionality:

The goal of this grade is to help you push your boundaries but work within your capabilities. Having a more complex project is part of the grade, but it is more important that your work functions. Be intentional and spend time trying to understand your problems instead of abandoning things that don’t work.

A good way to excel in this area is to explore a new technique we did not touch on in class OR expand meaningfully on something we touched on in class.

Site Specific Installation

25% Execution & presentation quality:

This is measured by how well presented your work is. This is aside from the creativity, but how well it is physically or digitally assembled. How would it look in a gallery?

A good way to excel in this area is to make sure your work is polished and clean. This can vary widely depending on the work. Clean up your cables, good presentation, well made objects or assets etc.

Site Specific Installation

25% Creativity & Concept:

Every project should have a concept. Concepts don’t have to be elaborate or in depth, they can be as simple as evoking a feeling, exploring a moment, explaining a subject or cause, sharing something you like, making the viewer feel something briefly. Keep your concepts simple as you start. This grade is measured by how effective your concept is on the viewer so don’t try anything complex when you start. Some really simple ideas: colours, nostalgia for a particular time, surprise, de-contextualize a space, etc. As you get better at concepts, you can explore larger ideas. Always have a concept, no matter how simple.

A good way to excel in this area is to have a clear concept that is evident to the viewers without explanation.

Site Specific Installation

25% Documentation:

Project is documented as outlined below. Remember: if you don’t document your project, nobody else will know it happened!

A saved file (File > Save as Project) .maxpat
A screen capture of your max patch
A clear video and/or audio recording of your project working (can be a video or a link to a private video on YouTube, Vimeo or Google)
A 50-100 word explanation of your project inside your maxpatch

Be prepared to talk about your work and process during critique as part of the documentation grade.

Critique procedure

- Everyone is expected to attend all critiques
- You may leave 1 critique session before yours to set up
- Everyone is expected to engage with all work and discussions

Critique checklist before you leave

- Location decided and put on the class signup list (I may change the order to accommodate walking distances)
- You have a list of equipment needed and where you'll get it
- You have a work plan, divided among each group member in clear tasks and scheduled meetings.
- You've discussed your idea with lee

Plan your code

- If you don’t have a clear vision of how your code will work, it will be harder to write. Start by thinking of your project from a purely artistic/experience perspective and then work backwards to how you can do that with your code.
- Write “pseudo code”, which means a written description of how your code will work.

 Ex: The user presses X and then a number counts from 1 to 10 and then changes the size of a circle

- Don’t forget that its okay to simplify. Your audience doesn’t know your wildest dreams of your installation, they only know what you show them. Maybe this is version 1, and you can do more later. Its always better to do less properly than more poorly!

Building your code

- Break down your code into small pieces.

Ex: Rotate a video, draw a sphere, determine face location.

- Get each piece to work in isolation before putting it in your main patch
- Don’t use chat GPT (seriously), use Google, Max Forums, ask lee or James or your peers!

* Chat GPT is good when you know when it is wrong. Sometimes its wrong in ways difficult to understand if you’re learning. Once you have a better grasp on using software its more useful.

Working collaboratively

- Work separately and meet to bring your work together
- Test any live video capture work in your final setting
- Divide tasks clearly
- Comment your code!
- Use GitHub for sharing files, if it works for you.

Critique Guide

Be prepared to talk about your work in detail. This includes both technical and artistic goals.

- Begin with letting people experience your work.
- We will open the floor to comments and discussions.
- Explain your artistic intentions. What did you intend to make people feel or think? How did you achieve that? Do you think it was successful? What would you improve next time? Discuss your process.
- After discussion, explain your technical details. How did you do this? What was easy or difficult? What did you learn? What was something unexpected? What would you do differently?

Some things to explore:

- Drawing in openGL
- Mixing video
- Explore max patches and Open CV
- Audio Reactive video
- Integrating tacking into installation
- Projection and colours in interactive spaces
- Using video to change or manipulate spaces
- Jitter / Makey Makey / Midi / Live music / Soundscapes
- Placement of cameras in space

Handing in projects

Each project should include a .zip file that contains:

You can also upload to GitHub in a repository and provide the link

- A saved file (File > Save as Project) .maxpatch
- A screen capture of your max patch
- A video or audio recording of your project working (can be a video or a link to a private video on YouTube, Vimeo or Google)
- A 50-100 word explanation of your project inside your maxpatch

Be sure to name files properly (no untitled-1.zip)

All files are expected to be cleaned up and arranged in a reasonable, legible way. Videos should be clear, well light and show your project working.

Homework:

Work on your mid term

Next week (after break):
Mid term critiques - Be ready at the start of class