



Yunnan Don't Have Ocean

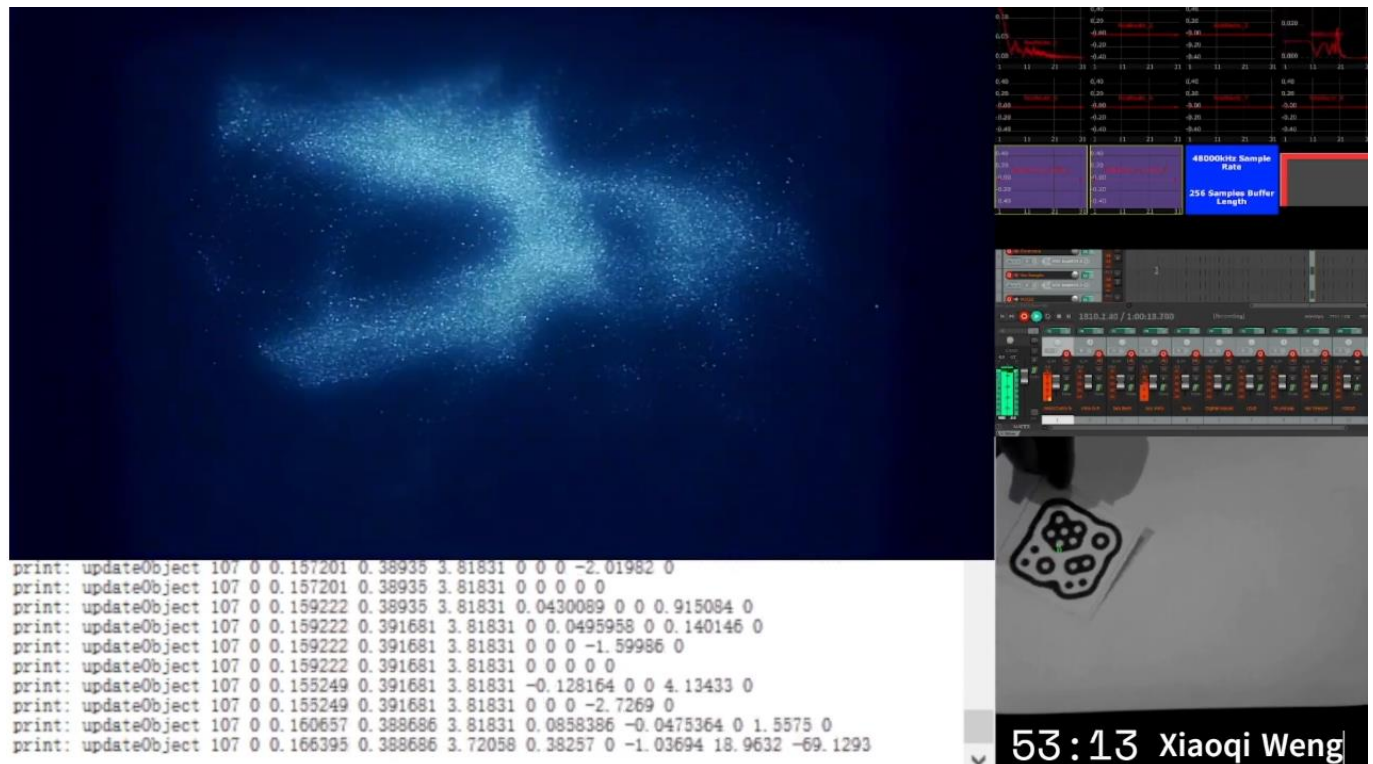
TECHNICAL RIDER

Work name: YunNan Don't Have Ocean (Interactive music with reactivision, pd, reaper, td)

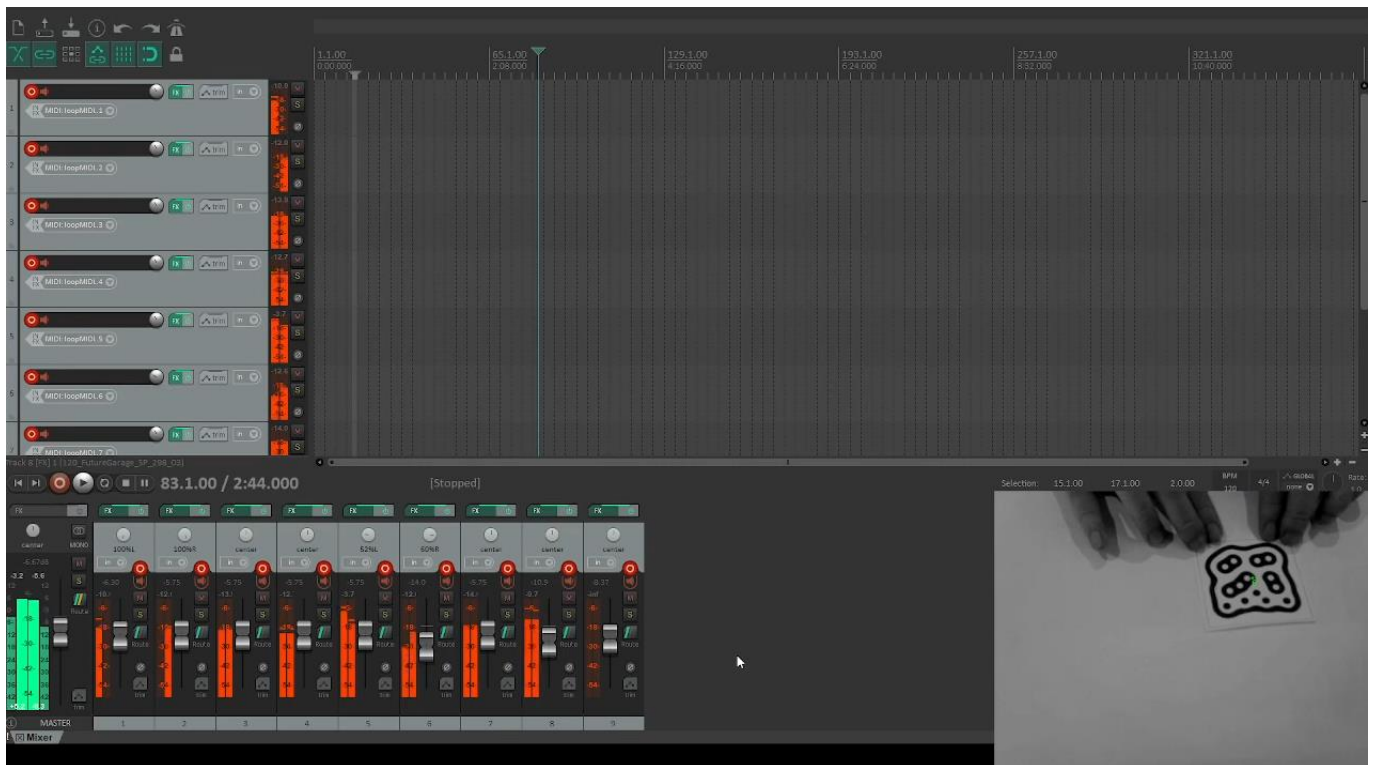
YouTube Link: <https://www.youtube.com/watch?v=zwkSo5SI8Ds>

Old Version (no vj, but sound good than new one):

<https://www.youtube.com/watch?v=xhaAqDXL32g>



The Newest Version



The Old Version

CRITICAL EVALUATION

Work name: YunNan Don't Have Ocean (Interactive music with reactivision, pd, reaper, td)

YouTube Link: <https://www.youtube.com/watch?v=zwkSo5Sl8Ds>

Old Version (no vj, but sound good than new one):

<https://www.youtube.com/watch?v=xhaAqDXL32g>

Introduction

This is a project that was completed using Reactivision, Pure Data, Reaper, and Touch designer. In this project, Reactivision was used as an OSC sender and the data was unpacked by Puredata through TUIOClient and converted into MIDI notes and CC signals. The obtained data was then sent to Reaper for improvisation using a sampler and VST instrument, while audio signals were also sent to Touch designer to generate VJ visuals, achieving the goal of audio-visual interaction. Yi ethnic minority folk tunes recorded in Yunnan, China were used in the project to reinforce the theme.

Contextualisation

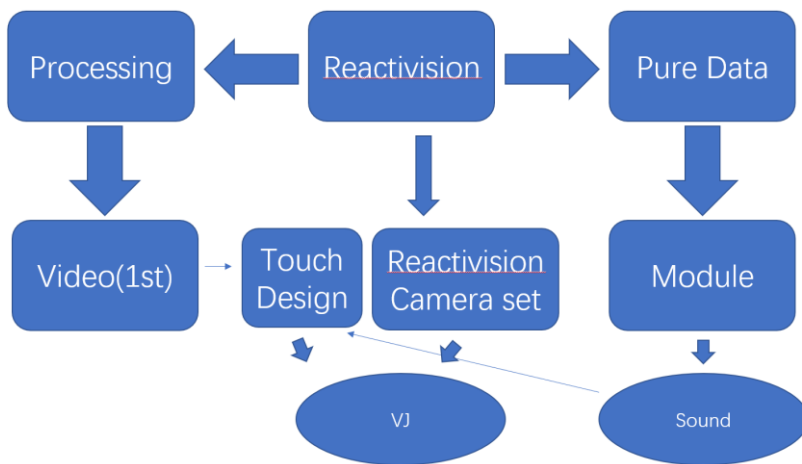
In the early days, I used Reactivision for a project involving robot movement. So when I started this audio-visual interactive project, Reactivision was the first thing that came to mind as a data input source. Reactivision, an open-source computer vision framework, can track multiple inputs, including finger touch recognition, making it an ideal OSC sender. For this project, I took inspiration from Chris Mylrea's work "Paper and Music," which is more musically sophisticated and mature than my own, and I'm not yet proficient enough to use paper as an instrument for performance. In the final step of transmitting Reaper's audio to Touch designer to generate VJ visuals, I used the TD kit "ReaRouteTD_MONO" created by World Wide Basement Vibes. Due to time constraints, it was not possible to map all the audio data from each track to the final video presentation, which is a pity.

In terms of the theme, I chose Yunnan because the city of Kunming is known as the "Spring City" in China, with a pleasant climate that's great for long-term living. However, as someone who was born on the coast, I often enjoy playing or fishing by the sea. But, the locals in Yunnan like to name their lakes "sea," even Yunnan is inland and does not have the ocean. Therefore, I chose this theme.

I think the idea and concept of the work will emerge more during the process of creating the work, and often, it's necessary to make small adjustments to the original workflow to achieve a successful result in all aspects.

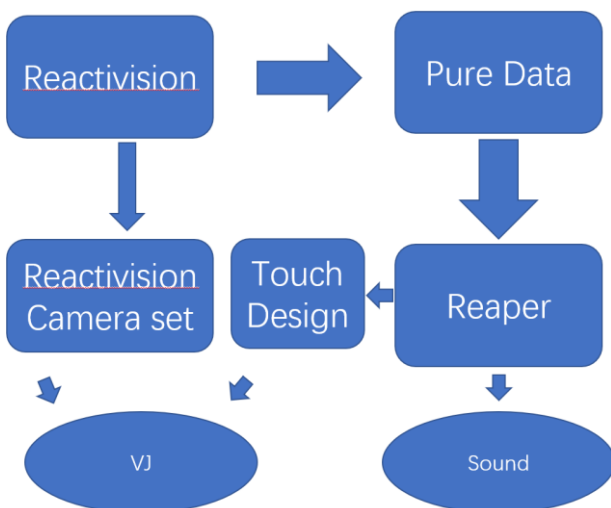
Detailed description of the project

Work Flow



This workflow showcases all the major software used in the project. However, please note that Processing and Module were no longer used in the final project due to compatibility issues. I will explain the reasons for this and possible solutions in the following text.

Work Flow(NEW)



The diagram above shows the workflow used in the final version of the project. I will use this diagram to explain the entire project and any issues that need to be noted.

1.Reactivision

As the main signal sender, the importance of Reactivision cannot be overstated. After downloading the standalone program from the official website, the TUIOClient package needs to be downloaded in order to send information to Puredata. *Hint: If you want to use Puredata to read the package content in TUIOClient.dll, you must use the PD-extended version instead of the latest version.*

1. Camera chooses

As I used a laptop computer to work on this project, when launching Reativision, it detected three cameras.

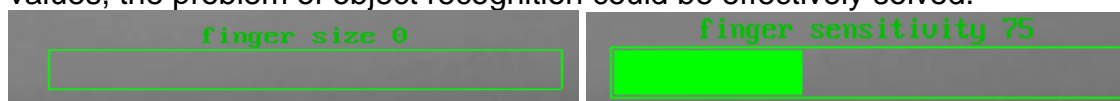
```
reactIVision 1.5.1 (May 18 2016)
3 videoInput cameras found
```

If the desired camera was not selected upon opening, We could change the camera number in the camera.xml configuration file located in the root directory, then restart Reativision until the correct camera was selected.

```
<portvideo>
  <camera id="1">
```

2. The input signal is unstable

When I first started using Reativision, I often encountered problems with object recognition, such as objects not being recognized or fingers being recognized too quickly and causing disruptions to the performance. Through my exploration, I found that by pressing the "o" key in the Reativision interface to open the camera options and adjusting the contrast and other values, the problem of object recognition could be effectively solved.



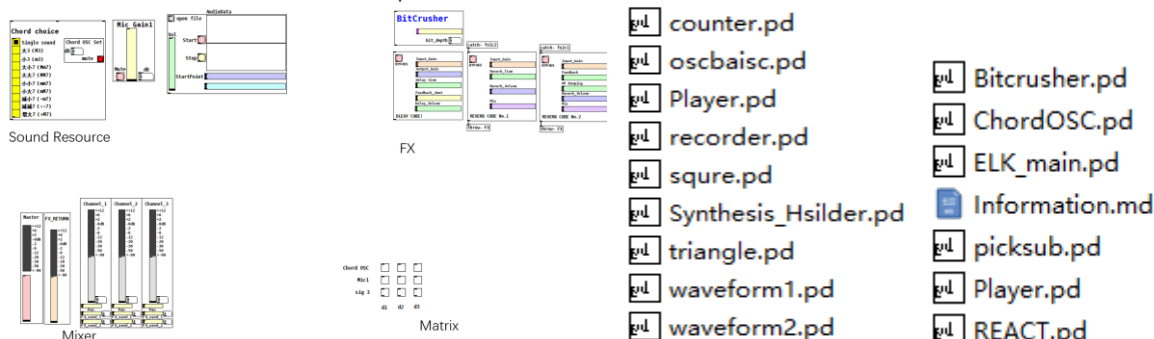
Pressing the "f" key in Reativision to adjust finger size and finger sensitivity can effectively solve the problem of fingers being recognized too quickly.

2. PureData

I mainly used PureData to unpack the data from Reativision and create MIDI signals.

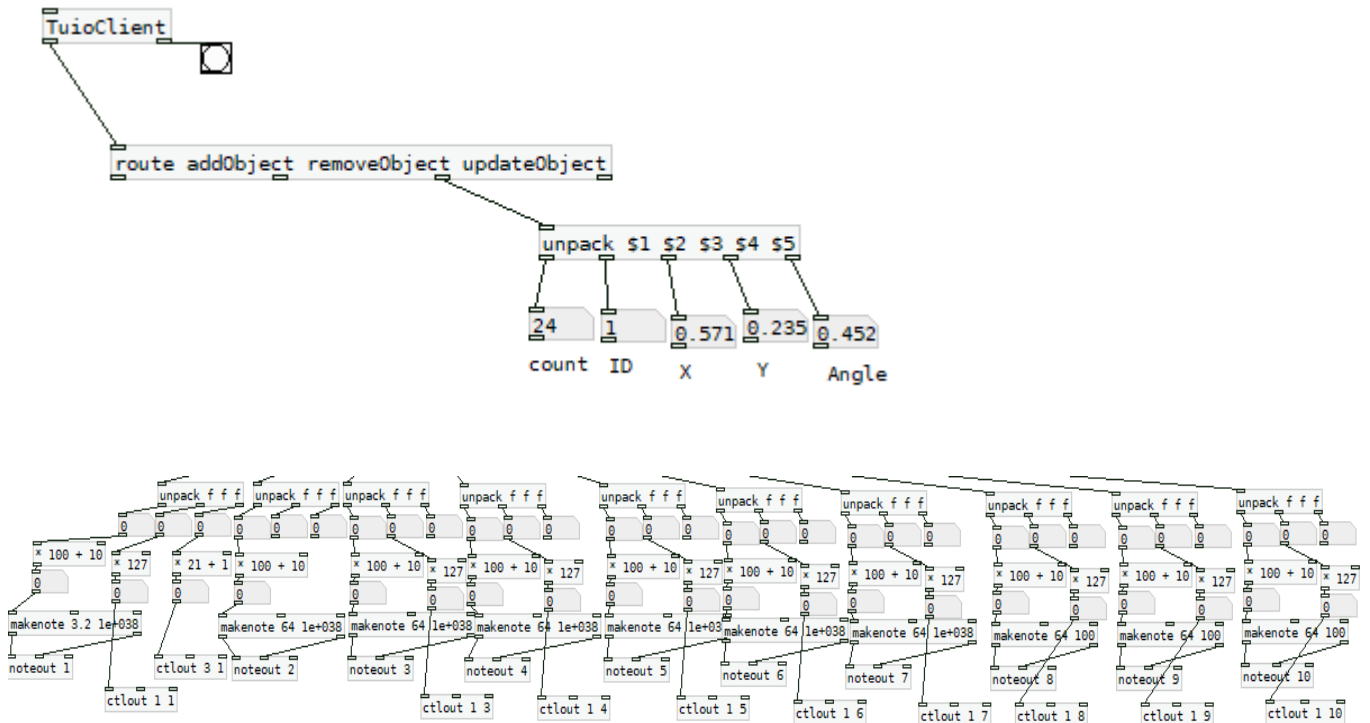
Before that, I also created a separate PD module and planned to use it as a sound source to make music. However, due to time constraints, I did not choose it as the final solution.

The Module I made with pd



The diagram above shows some of the contents of my PD module, which mainly includes building a human-friendly visual interface and a retro routing method in the patch. This allows for the selection of compatible synthesizers, effects, and samplers to achieve free creative expression, meaning I can quickly change the timbre needed for each different performance. It's a shame that this module wasn't used in the final project.

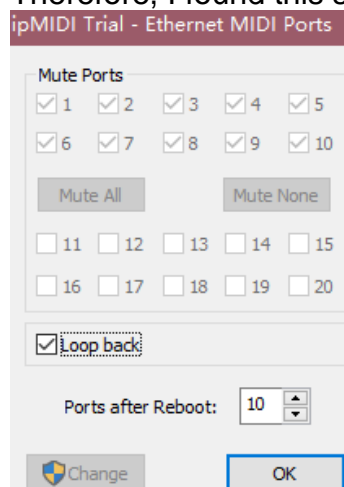
Unpacking Reactivision data in PD is very simple. You only need to download the corresponding extension package from the official website and create a new path in the root directory of the package to use all the functions. After unpacking, we can see that Reactivision provides us with five values that can be used for compilation, which are "count, Object ID, X, Y, Rotation angle." In my final project, I used the compiled "X" value more to control MIDI notes, and "Y" to control CC1, which is the modulation wheel. I only used the rotation angle value in Object ID 0 to control the volume.



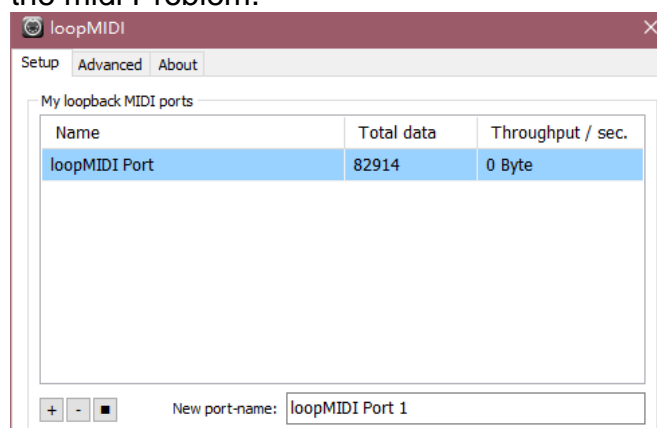
Problems that may be encountered in using PureData

MIDI Output

I encountered a problem when using PD to transmit MIDI signals. Because I was using a Windows computer, I couldn't use the virtual MIDI port IAC that is available on Mac. Therefore, I found this software to route MIDI.



However, since the multi-port feature of this software requires payment to use, I asked Joe for help and he recommended a free and useful tool called "LOOPMIDI" It can fix all the midi Problem.



<https://www.tobias-erichsen.de/software/loopmidi.html> by: Tobias Erichsen

3. Reaper

Yes, I used Reaper to produce sound for my project! Reaper is the most user-friendly DAW I have ever seen for recording and audio programming. It is concise, does not take up a lot of memory, and there is no difference between the free and paid versions, which is why I chose it. If I had used Cubase or Live as the sound source, my computer might have crashed during the performance (because Touchdesigner takes up a lot of memory and requires high GPU usage). Next, I will introduce my channel strip list and routing design in "The Lab".

Channel In Reaper				
Channel	Name	Vst name	Output	Pan
1	Intor/Outro fx L	HS Dyson	1	L100
2	Intro fx R	HS Crackdown	2	R100
3	Sub Bass	Bobjas Bass NF	3&4	C
4	Arp Intro	PV- The Man Machine	3&4	C
5	Sync	Dirty Sync	5&6	L52
6	Digital sound	1 Calculation	5&6	R60
7	LEAD	LEAD Sucker	3&4	C
8	Drumloop	Sample_FutureGarage_SP_298_03	1&2	C
9&10	Voc Sample	Sample_喊坡（喊龙脉）三眼萧	7&8	C

As shown in the figure, these are all the sound source channels that I used in Reaper for this project and their corresponding hardware outputs. I used Tyrell N6 for all the synthesizers, which is a free synthesizer that takes up very little space compared to Serum and Massive. I highly recommend using it.

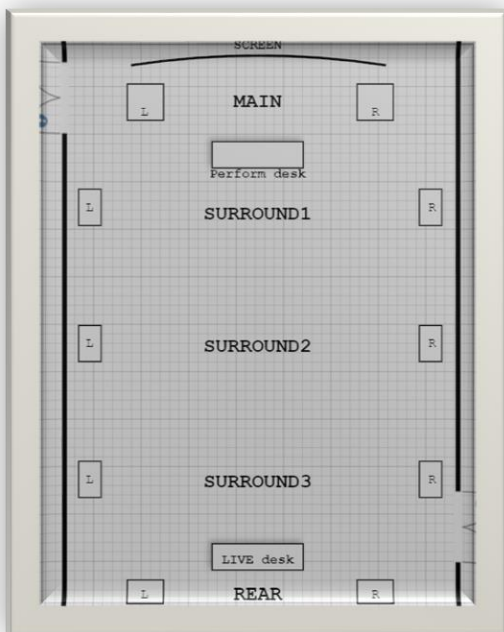


Tyrell N6 Vst

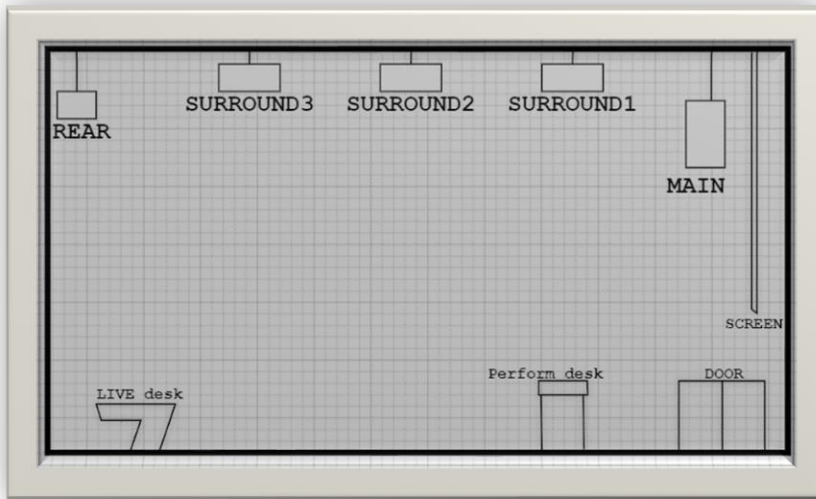
The routing in The Lab is shown in the figure below, along with a bird's-eye view and elevation diagram of The Lab to help determine the position of the speakers.

Channel In Desk			
Channel	Pan	Q to Monitor	Stereo link
1	L100	S2, MAIN	1&2
2	R100	S2, MAIN	1&2
3	L100	S1, MAIN	3&4
4	R100	S1, MAIN	3&4
5	L100	S1,S2,S3	5&6
6	R100	S1,S2,S3	5&6
7	L100	REAR,S3	7&8
8	R100	REAR,S3	7&8

Channel list for The Lab Live Desk



bird's-eye view diagram

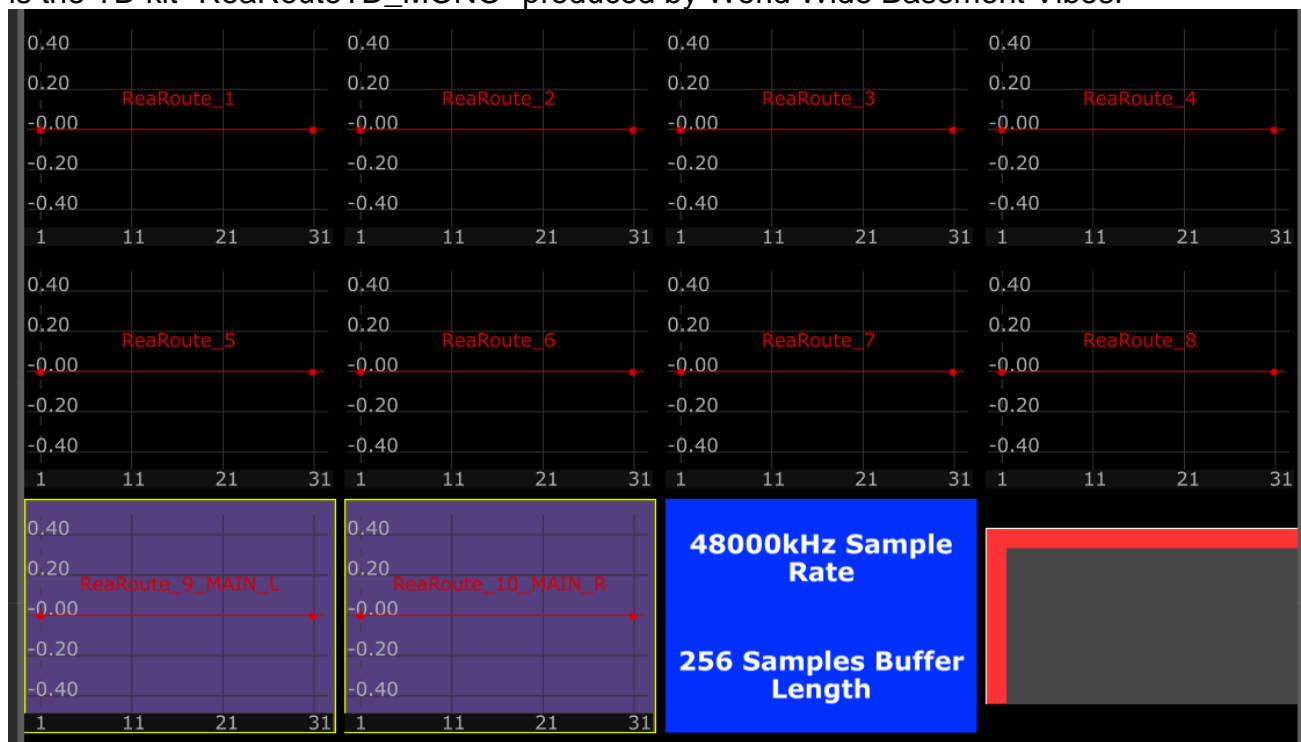


elevation diagram


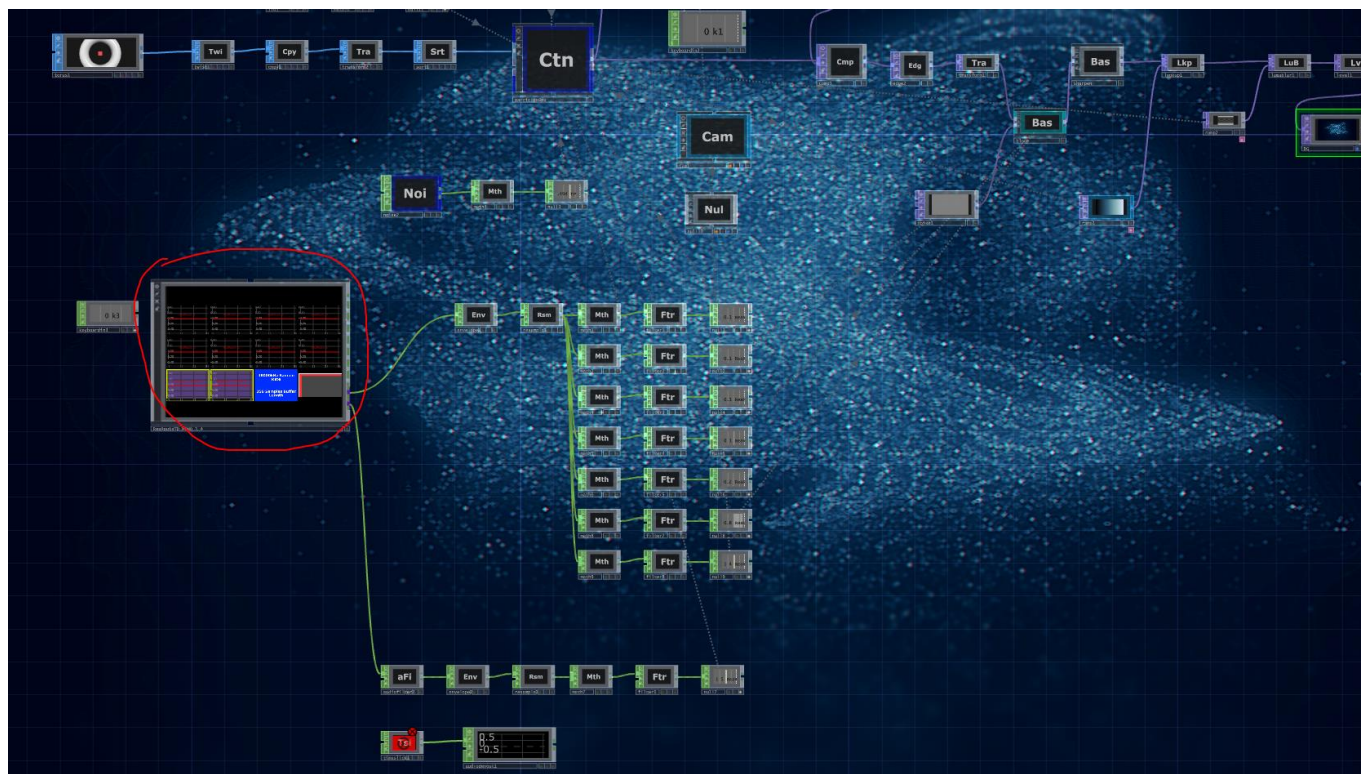
By understanding the placement of the speakers in The Lab, we can precisely control the live sound effects and provide a better auditory experience for the audience. Unlike stereo, placing different tracks on different speakers allows the audience in the venue to hear sound from different locations. For example, for the last two Voc sample channels, if I put them all in one stereo channel, I can only create a sense of distance for the audience through internal software adjustments, which is far less real than the direct sound distance on-site. Sounds behind the audience are usually more impressive than sounds in front of them. This is also why many soundscapes achieve immersion through different Speaker position design.

4. Touch Designer

My original plan was to use Processing for VJ, but due to the problem of the occupied OSC port, I switched to Touch Designer urgently. What I want to emphasize the most in this part is the TD kit "ReaRouteTD_MONO" produced by World Wide Basement Vibes.



The usage of this tool is also very simple. When installing Reaper, install the ReaRoute add-on component at the same time, which can add an additional 16 virtual hardware output channels for Reaper. After connecting the corresponding channels to be output one by one, the audio signal can be transmitted to Touch Designer through this Kit.



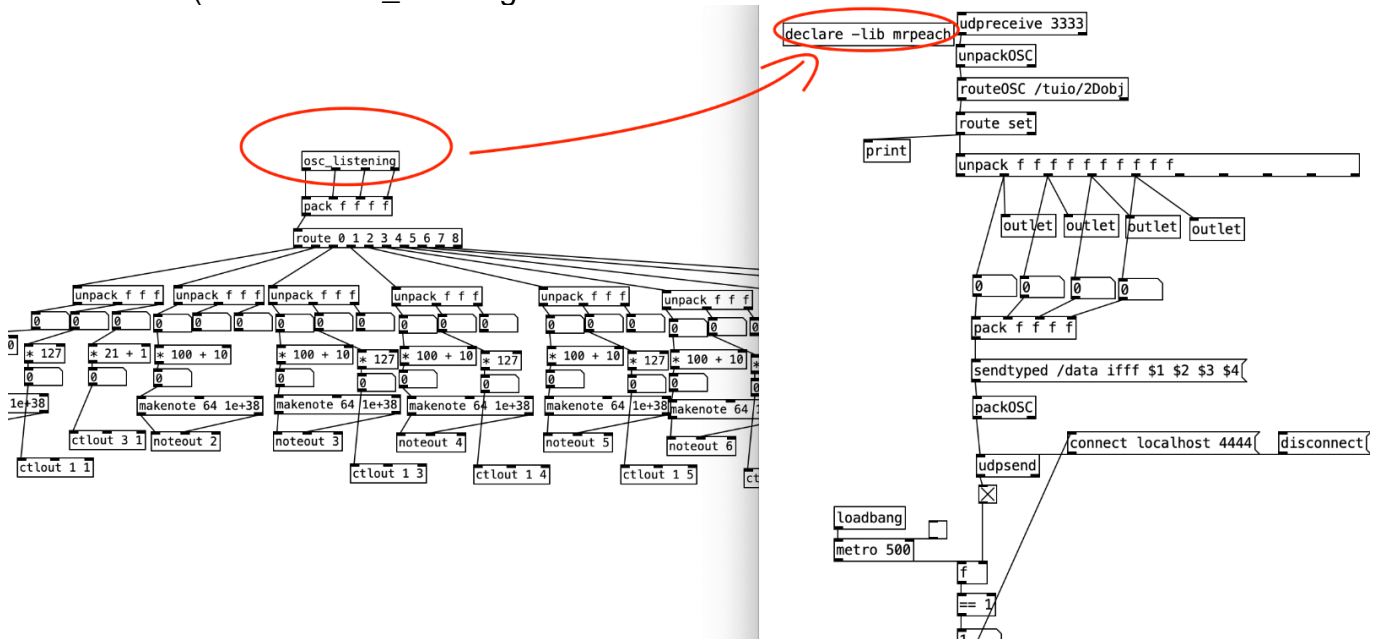
10

Effective ways to solve the Processing osc port occupation issue

This patch does NOT need TUIO. Instead, it uses the [mrpeach] library.

*** Important: ***

The PD files (midiout / osc_listening must be located in the same folder!



Due to the port 3333 being occupied by PD, it is difficult to add a new port in Reactivision. As a Windows user, I cannot use tools like Osculator to replicate and forward the OSC signal. Therefore, it is more reliable to use PD to repack and send the information if you need to control values in multiple programs using Reactivision. The first image shows the specific solution, where you need to import "mrpeach" before using it in the red-highlighted part. The second image shows the updated processing line corresponding to the first image.

```
void setup() {  
  /* start oscP5, listening for incoming messages at port 3333 */  
  oscP5 = new OscP5(this,4444);  
  myRemoteLocation = new NetAddress("127.0.0.1",12000);  
}
```

To consolidate the problems and solutions that appeared in my long debugging process of Processing, I wrote a report on how to correctly map OSC values from Reactivision to Processing. However, now it seems that only the latter part is needed (if using the above solution). For ease of code viewing, I used a markdown file to edit this report document, and I will send it as an attachment. (File name: Processing_Part_Report.md)

Conclusions and further work

This project has truly taught me a lot. As Reactivision is an old software and is not widely used anymore, I had to put in a lot of effort to find corresponding solutions and thought processes. However, I am now very proficient in using it and have even gained a deeper understanding of PD's functions such as OSC sending and receiving. I will continue to develop this project to make it as complete and meaningful as possible.

References

1. *Music with Paper. Reactivision Improvisation with MIDI (Reactable)* Chris Mylrea, 2009 https://youtu.be/5fVWg_rfQ0w

2. *ReaRouteTD_MONO: Reaper + TouchDesigner for Easy Real-time Audio Reactive Vid (Tutorial + .tox file)* WORLD WIDE BASEMENT VIBES,
<https://youtu.be/7tiYVq143gE>
3. Tyrell N6, u-he, <https://u-he.com/products/tyrelln6/>
4. Reactivision, <https://reactivision.sourceforge.net/>
5. Tuio, <http://www.tuio.org/?software>
6. LoopMidi, Tobias Erichsen, <https://www.tobias-erichsen.de/software/loopmidi.html>