Sound & Sight and *Phase Music*: Two Transdisciplinary Courses

by

Julian Nguyen

A Research Paper Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Musical Arts

Approved November 2023 by the
Graduate Supervisory Committee:

Jonathan Swartz, Co-Chair
Seth Thorn, Co-Chair
Fernanda Navarro

ARIZONA STATE UNIVERSITY

December 2023

ABSTRACT

From the earliest operatic spectacles to the towering Coachella-esque stages that dominate today's music industry, there are no shortage of successful examples of artists combining music and visual art. The advancement of technology has created greater potential for these combinations today. Music curriculums that wish to produce well-rounded graduates capable of realizing this potential need to adapt to teach how to incorporate technology in performances. This paper presents two new courses that integrate technology with performance: Sound & Sight: A Practical Approach to Audio-Visual Performances; and Phase Music: An Introduction to Design and Fabrication. In Sound & Sight, students will learn how to "storyboard" pieces of music, realize that vision through object-oriented programming in Processing, and synchronize audio and visual elements in live performance settings using Ableton Live and Max. In Phase Music, students will be introduced to Phase Music, learn how to use Ableton Live to perform one of Steve Reich's phase pieces or compose and perform their own piece of phase music, and design and build a custom Musical Instrument Digital Interface (MIDI) controller using Arduino, Adobe Illustrator, and Max. The document includes complete fifteen-week lesson plans for each course, which detail learning objectives, assignments, use of class time, original video coding tutorials, and lecture notes.

To my parents, family, and friends, for your endlessly generous and enthusiastic support.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

INTRODUCTION

This research paper is a distillation of the approaches and techniques I have developed through creating audio-visual performances with my violin, code, and electronics. Taking the form of two college level classes, it serves to pass on the knowledge I have inherited from others participating in the long tradition of integrating technology into music performance, while inquiring into a theoretical understanding of what this activity means and what possibilities it offers when forming relationships with our tools and each other. To that end, I briefly examine the evolution of how practitioners of this integration have understood and described their own work over the last few decades. Keeping in mind how one might answer a student who asks what the point of building a new musical controller is, this introduction traces this history by looking at Robert Rowe's classic systems of classification for understanding works of computer music; at the growth and evolution of the popular NIME conference during the last two decades; and finally, at Tara Rodgers' alternatives to the violent metaphors coded into computer music during its earliest days.

In *Interactive Music Systems: Machine Listening and Composing,* a text exemplary of a formative period in computer music, Rowe lays out three axes upon which interactive musical systems can be classified: Score- vs. performer-driven programs, where the system either relies on predetermined event patterns or doesn't; response methods that are transformative, generative, or sequenced; and the instrument and player paradigms, where instrumentalists interact with computers through gestural controllers or the computer acts as a player participating in chamber music with its human counterpart.

This last axis is especially salient as it identifies a compelling motivation for creating new controllers - satisfying the "desire to include human musicianship in computer music compositions"[1] that has become critical to artists and researchers operating in the field of Human-Computer Interaction.

As new input and sensor technologies emerge, they are adapted for use in musical contexts and the body of research exploring novel ways of composing, performing, and experiencing music has seen considerable growth, as evidenced by the history of the New Interfaces for Musical Expression conference, first held in 2002. The number of papers published at NIME each year has consistently increased, and as of 2020, the proceedings archive contained 1,867 papers representing 2,550 unique authors who have collectively received 20,658 citations.[2] The sustained interest and contribution of such a large number of HCI researchers speaks to their recognition that musical instruments are ideal paradigms of gestural "control." The methods and principles they establish and refine act as a sort of sediment of computer-musical experience, creating a foundation for future generations to build upon.

In *Principles for Designing Computer Music Controllers*, published at the very first NIME conference, Perry Cook articulates that "the music we create and enable with our new instruments can be… greatly influenced by our initial design decisions and techniques."[3] As such, it is imperative to reflect critically on the origins of the tools of the

---

[1] Robert Rowe, *Interactive Music Systems: Machine Listening and Composing*, (Cambridge: MIT Press, 1993), 5.
[2] Stefano Fasciani and Jackson Goode, "20 NIMEs: Twenty Years of New Interfaces for Musical Expression," *NIME 2021*.
[3] Perry Cook, "Principles for Designing Computer Music Controllers," *arXiv.org* (2020): 3.

trade when endeavoring to educate the future generation of technologically literate

musicians. In her book *Pink Noises: Women on Electronic Music and Sound*, Tara

Rodgers explains:

> "The tools for making electronic music are not innocent: true sound 'mediums,'
> they are an interface to ghosts of technoscientific projects of the past. In the United
> States, links between audio and military technologies were well established by the
> 1920s… These associations persist today in the terminology of electronic music:
> DJ's 'battle'; a producer 'triggers' a sample with a 'controller,' 'executes' a
> programming 'command,' types 'bang' to send a signal, and tries to prevent a
> 'crash.' The very act of making electronic music thus unfolds with reference to
> high-tech combat, shot through with violent symbols of confrontation and
> domination."[4]

Students who are made aware of and offered alternatives to these underlying

references are better equipped to make informed decisions on how to navigate the

continued use and development of the implements they add to their creative toolboxes.

Rodgers presents one such alternative path, trailblazed by Theremin virtuoso Clara

Rockmore, who she credits with "open[ing] an 'elsewhere' within electronic music

discourses: a space for mutual encounters between humans and technologies, between

familiarity and otherness, that motivates a sense of wonder and a sense of possibility

instead of rhetorics of combat and domination."[5] As educators and artists, it is our

responsibility to keep these "elsewheres" open in classrooms and concert venues, because

operating within them inspires the expansion and exploration of alternate paths and

dimensions in other areas, especially given how technology now permeates nearly all

aspects of modern-day life.

---

[4] Tara Rodgers, *Pink Noises: Women on Electronic Music and Sound,* (Durham: Duke
University Press, 2010): 6-7.
[5] Ibid., 9.

Just as the corporeal boundaries between members of a string quartet blur as they create a unified musical output, the boundaries between the concert hall and the outside world blur when artists who model Rockmore's approach of engaging technology with nuance and care prime audiences to return to their communities reinvigorated to realize the possibilities of enhanced togetherness, healing, and empathy that approach enables. In an effort to travel along Rodgers' more affirmative and imaginative path into technologically augmented artistry, I humbly present *Sound & Sight* and *Phase Music* as invitations for readers to create new relationships, communities, and friendships that evoke wonder and awe through the addition of technology to their practices.

SOUND & SIGHT: A PRACTICAL APPROACH TO AUDIO-VISUAL

PERFORMANCES

Course Description

Sound & Sight is a transdisciplinary course that integrates topics and approaches from music, visual art, and computer science. Throughout the semester, students learn to code responsive visuals and synchronize them with live and pre-recorded audio using Processing, Ableton Live, and Max. The course culminates with public student performances of the works developed in class. Outside of class, students engage with a combination of original video coding tutorials, readings, and projects. Class time is used for lectures, discussion, and supervised labs where the instructor answers questions, provides feedback, and helps students complete the weekly assignments.

Week 1: Introduction to Principles of Coding and Audio Analysis for Visualizing Music

Before class, students will be expected to complete coding tutorials *3 shapes 4 colors pt. 1-3,*[6] and readings from *FORM+CODE*[7] and *Learning Processing*.[8] These tutorials and readings introduce students to fundamental coding concepts including abstraction, algorithms, and incremental development, and the specific coding language, Processing, that will be used to create visuals. Students will learn how to draw shapes and

---

[6] See appendix, nos. 1-3.
[7] Casey Reas, Chandler McWilliams, and Jeroen Barendse, *Form+Code in Design, Art, and Architecture*, (New York: Princeton Architectural Press, 2010), 8-41 and 119-143.
[8] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images*, *Animation, and Interaction,* 2nd ed. (San Diego: Elsevier Science, 2015), 3-65.

change their colors using code, as well as how to use mouse input to modulate parameters of their Processing sketches. During the first class, the instructor will provide an opportunity for the students to briefly get to know one another to foster a collaborative environment, set work ethic expectations for the semester, and share examples of the type of works students can expect to produce by the end of the semester. A lecture on musical analysis to storyboard a visual accompaniment, grounded in a case study of Jessie Montgomery's Rhapsody No. 1, will be delivered in class. This lecture will cover analytical strategies that do not require prior music theory training. Remaining class time will be used as supervised lab, where students will work on the assignment: an analysis of a chosen piece for their final project, and an "*n*-shapes *n*-colors" Processing sketch that reinforces the skills learned from this week's coding tutorials. Students will be provided with a template on which to base their musical analysis.

<u>Assignment</u>

- Write five processing sketches that contain at least one shape, two colors, and one responsive element.

- Submit a first draft of a formal analysis of the final project piece, and a storyboard for the visual based on that analysis.

- Watch coding tutorials *repetition & animation pt. 1-2.*[9]

- Read *FORM+CODE* chapters Repeat and Transform;[10] and *Learning Processing* chapters 5-7.[11]

---

[9] See appendix, nos. 4-5.
[10] Casey Reas, Chandler McWilliams, and Jeroen Barendse, *Form+Code in Design, Art, and Architecture*, (New York: Princeton Architectural Press, 2010), 42-91.
[11] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction,* 2nd ed. (San Diego: Elsevier Science, 2015), 67-137.

Week 2: If Statements and For Loops

Before class, students will be expected to complete coding tutorials *repetition &*
animation pt. 1-2, as well as readings from *FORM+CODE* and *Learning Processing*.
These materials will introduce the concepts of repetition and animation, and how to
implement them in Processing sketches using if statements and for loops. Class time will
be devoted to supervised lab, where students will share the previous week's assignment
with their peers, continue refining their musical analysis/visual storyboard, and work on
this week's assignment: Processing sketches utilizing repetition and animation.

Assignment

- Write three processing sketches. One must include repetition using a for loop.
  One must include animation controlled with if statements. One must include both
  repetition and animation. All three must include at least one responsive aspect.

- Watch coding tutorials *orbital motion / polygon algorithm* and *coin toss.*[12]

- Read *Learning Processing* chapter 9 and 13.[13]

Week 3: Arrays, Generativity, and Implementing Math in Processing

Before class, students will be expected to complete coding tutorials *orbital motion*
/ polygon algorithm and *coin toss,* as well as readings from *Learning Processing.* These
materials introduce students to using math in Processing, specifically the sin() and cos()
functions, as well as arrays and generativity using the random() function. Class time will
be devoted to supervised lab, where students will share the previous week's assignment

---

[12] See appendix, nos. 6-7.
[13] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images*,
Animation, and Interaction, 2nd ed. (San Diego: Elsevier Science, 2015), 163-185 and 233-263.

with their peers and work on this week's assignment: Processing sketches that use arrays, math, and generativity.

Assignment

- Write three Processing sketches. One must include the sin() and/or cos() functions. One must include the random() function. One must include an array. All sketches must include at least one responsive aspect.
- Watch coding tutorials *oop pt. 1-2*.[14]
- Read *FORM+CODE* chapter Parameterize[15] and *Learning Processing* chapter 8.[16]

Week 4: Object-Oriented Programming

Before class, students will be expected to complete coding tutorials *oop – ripples, and oop – shrinking shapes,* and readings from *FORM+CODE* and *Learning Processing*. These materials introduce the concepts of object-oriented programming and parameterization. Class time will be devoted to supervised lab, where students will share the previous week's assignment with their peers and work on this week's assignment: Processing sketches that implement object-oriented programming.

Assignment

- Write two Processing sketches. Each sketch must contain at least one class and make use of at least one instance of an object described by that class. Both sketches must include at least one responsive element.

---

[14] See appendix, nos. 8-9.
[15] Casey Reas, Chandler McWilliams, and Jeroen Barendse, *Form+Code in Design, Art, and Architecture*, (New York: Princeton Architectural Press, 2010), 92-117.
[16] Shiffman, 139-159.

- Read the "What is Max?" webpage.[17]

- Watch coding tutorial *udpsend~ & oscP5*.[18]

- Watch the following videos on ableton.com's "Learn Live" page: *Live's user interface, Session View, Arrangement View, Tracks, Live's Browser, Detail View, Live's Mixer, Setting up an audio interface, Setting up MIDI*, and *Configuring your audio ins and outs*.[19]

- Read *Learning Processing:* chapter 12.[20]

Week 5: Ableton Live, Max, Max for Live, Networking with OSC, and Synchronization Strategies

Before class, students will be expected to watch coding tutorial *updsend & oscP5,* watch ten short videos on the Ableton Live website, and complete readings from *Learning Processing* and the Cycling '74 website. The readings introduce students to using libraries in Processing and the Max programming language. The videos and tutorial introduce students to Ableton Live, computer networking, and teach how to communicate between Processing and Max using the Open Sound Control (OSC) protocol. During class, part one of a lecture on synchronization strategies will introduce the Max programming language, the Digital Audio Workstation (DAW) Ableton Live, and present two approaches to sending OSC messages to Processing from Max: using a MIDI controller to control a Max for Live device for human input, and programming

---

[17] "What is Max?" Accessed November 12, 2023, https://cycling74.com/products/max
[18] See appendix, no. 18.
[19] "Learn Live," Accessed October 1, 2023, https://www.ableton.com/en/live/learn-live/
[20] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction,* 2nd ed. (San Diego: Elsevier Science, 2015), 225-230.

automation envelopes in Ableton Live to sync the Processing sketch with Live's master clock. Remaining class time will be used as supervised lab, where students will share the previous week's assignment with their peers and work on this week's assignment: a Max patch that communicates with Processing via OSC.

<u>Assignment</u>

- Write a Max patch that controls at least five parameters in a Processing sketch using the udpsend~ object and the oscP5 library. The Processing sketch may be one of the previous week's assignments updated to receive OSC messages.

- Create an Ableton Live Set that demonstrates basic competency with audio routing, audio effects, and automation envelopes.

- Watch coding tutorials *images pt. 1-2*.[21]

- Read *Learning Processing:* chapter 15.[22]

Week 6: Image Manipulation and antescofo~

Before class, students are expected to complete coding tutorials *images pt. 1-2* and a reading from *Learning Processing*, which introduce them to importing and displaying images, manipulating image transparency and alpha masking techniques, and using PGraphics buffers in Processing. During class, part two of the synchronization lecture will introduce students to antescofo~, a Max object that enables automated synchronization with live performers. Remaining class time will be used as supervised lab, where students will work on this week's assignment: Processing sketches that include

---

[21] See appendix, nos. 10-11.

[22] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction,* 2nd ed. (San Diego: Elsevier Science, 2015), 301-327.

at least one image and/or PGraphics, synchronizing their final project piece audio with visuals, and revising their audio analysis/visual storyboard.

<u>Assignment</u>

- Write three Processing sketches that include at least one image and/or PGraphics. Each sketch must include at least one responsive element.
- Use one or more of the three strategies learned in class to synchronize the final project piece with one or more of the visuals created thus far.
- Revise audio analysis/visual storyboard.
- Synthesize everything learned in class so far to complete a first draft of the final performance piece.

Week 7: Final Project First Drafts

Before class, students are expected to complete a first draft of their final performance project. Class time will be devoted to a "studio class" where each student or student group may perform their piece and receive constructive feedback from the instructor and their peers. Performances will be recorded so students may review them. Remaining class time will be used as supervised lab, where students will work on implementing feedback they receive on their final project.

<u>Assignment</u>

- Watch the recording from studio class and compile a list of feedback from self, peers, and instructor.
- Begin to implement desired feedback on final project.

- Watch coding tutorials *images pt. 3-5.*[23]

Week 8: Advanced Image Manipulation and Drawing in 3D

Before class, students will be expected to complete coding tutorials *images pt. 3-5: pseudo slit scan, funhouse mirror, and kaleidoscope.* These tutorials introduce students to advanced image manipulation at the pixel level using the pixels[] array and drawing in 3D using the P3D renderer in Processing. Class time will be used as supervised lab, where students will work on this week's assignment: Processing sketches that use the pixels[] array and the P3D renderer.

Assignment

- Write two Processing sketches. One must use the pixels[] array. One must use the P3D renderer. All sketches must include at one least responsive aspect.

- Watch coding tutorials *illusion of motion pt. 1-3.*[24]

- Read *Learning Processing* chapter 14.[25]

Week 9: Animation in 3D

Before class, students are expected to complete coding tutorials *illusion of motion pt. 1-3* and a reading from *Learning Processing,* which introduce students to intermediate 3D drawing techniques, including animation in 3D and camera manipulation. Class time will be used as supervised lab, where students will share the previous week's assignment

---

[23] See appendix, nos. 12-14.
[24] See appendix, nos. 15-17.
[25] Daniel Shiffman, *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction,* 2nd ed. (San Diego: Elsevier Science, 2015), 265-297.

with their peers and work on this week's assignment: one Processing sketch that creates the illusion of motion.

<u>Assignment</u>

- Write one Processing sketch that creates the illusion of motion by moving things around a stationary camera. The sketch must include at one least responsive aspect.

- Find a visual, share it with the class, and try to recreate it in Processing.

Week 10: Control Structures in Max and Reverse-Engineering Visuals

Before class, students will find a visual, share it with the class, and try to replicate in Processing. This assignment is akin to learning a piece of music "by ear," and will exercise students' creative reasoning skills by challenging them to navigate from blank canvas to finished product through an iterative process of analysis, trial and error, and reflection. During class, two lectures will be given. The first covers creating Low Frequency Oscillators (LFOs) and sampling amplitude in Max, with the intent of using these sources to modulate parameters in a Processing sketch. The second presents a case study on reverse-engineering a visual, providing students with an analytical framework within which to recreate visuals. A discussion examining students' approaches will follow the case study, creating the possibility for novel analytical frameworks to be generated. The remaining class time will be supervised lab, where students will share their assignments from last week, and work on this week's assignment: a written analysis of the visual they chose to recreate, a Processing sketch that attempts to recreate or is

inspired by the visual they found before class, and a Max patch that uses one or more of the control structures covered in class.

Assignment

- Submit a written analysis of the visual found before class.

- Finish recreating the visual in Processing.

- Write a Max patch that uses an LFO and/or samples amplitude to control one or more parameters in a Processing sketch. The Processing sketch may be one of the previous week's assignments.

- Watch and complete a coding tutorial on Processing, Max, or Ableton Live.

- Create a five- to ten-minute presentation on the tutorial, what you learned, and the work generated from it.

Week 11: Autodidacticism

Before class, students will be expected to find and complete an outside coding tutorial for Processing, Max, or Ableton Live. During class, students will give a five- to ten-minute presentation on the tutorial and their Processing sketch, Max Patch, or Ableton Live Set based on and/or inspired by it. The remaining class time will used as supervised lab, where students will work on their final projects.

Assignment

- Work on Final Project.

Week 12: Lab

Class time will be used as supervised lab, where students will work on their final projects. The goal is for the project to be more or less complete by this week so that students have a full two weeks to rehearse their pieces before the final performance.

Assignment

- Submit final draft of audio analysis/visual storyboard.

- Finish final project.

- Run through and practice final project piece every day.

Week 13: Studio Class

Outside of class, students will be encouraged to run through and practice their pieces every day leading up to the final performance. Class time will be used for "studio class" where each student or student group will perform their piece and receive constructive feedback from the instructor and their peers. Performances will be recorded so students may review them. Remaining class time will be used as supervised lab, where students will implement feedback they receive on their final project.

Assignment

- Watch recording of in-class performance and compile list of feedback from self, peers, and instructor.

- Implement desired feedback.

- Run through and practice final project piece every day.

Week 14: Dress Rehearsal

Outside of class, students will be encouraged to run through and practice their pieces every day leading up to the final performance. Class time will primarily be used for dress rehearsals in the performance space where the final performance will take place. Dress rehearsal run through will be recorded for students to review. Remaining class time will be used as supervised lab, where students will implement feedback they receive on their final project.

Assignment

- Watch recording of dress rehearsal and compile list of feedback from self, peers, and instructor.

- Implement desired feedback.

- Run through and practice final project piece every day.

Week 15: Final Performance

Outside of class, students will be encouraged to run through and practice their pieces every day leading up to the final performance. In lieu of a class meeting, there will be a final performance featuring all the students and the work they developed over the semester.

Assignment

- Give final performance.

Week 1

<u>Coding Tutorials</u>

- *3 shapes 4 colors pt. 1-3*

<u>Readings</u>

- *FORM+CODE* chapters Introduction, What is Code?, Form and Computers, and Visualize

- *Learning Processing* chapters 1-4

<u>Lecture Notes</u>: *Visualizing Music*

Formal Analysis

Today, we'll begin the process of conceptualizing visuals for the piece(s) of music you've chosen for your final performances. Whenever approaching a piece for the first time, I find it useful to employ some formal analysis. There are tons of ways to do this, but we're going to start with a very basic approach. While this may be limiting for more experimental/less conventional pieces of music, it will still provide a basic set of tools and ways of thinking about music that can be expanded upon. If you have been exposed to classical music theory topics or classes in the past, you are likely familiar with a number of analytical strategies and "classical" forms - Binary, Ternary, Rondo, Sonata, etc. Those of you who have had exposure to popular music are likely familiar with the terms intro, verse, chorus, bridge, and outro. If you're into EDM you probably also know all about build-ups and drops. What all of these analytical strategies have in common is that they define segments of time (containers) in which musical events (contents) take

place. Defining and analyzing these segments of time provides us a framework within which to craft a visual. Each section of a piece of music can correspond to a scene in our visual, and the musical activity within those sections can influence what we choose to populate that scene with.

If you're working with a piece that has a written score, a lot of this information can be found just by looking at the music. Is there a double bar separating sections? How about a tempo change marking? Or maybe a change in character or expression is prescribed by the composer. If there is not a written score, or you don't have access to one, you'll have to use your ear, which is still recommended regardless.

How do we figure out or decide the size of sections? One way is to identify repetitive elements in a piece, and then look at where those repetitions change and/or remain the same. Most of these elements can be boiled down to fit into these categories:

- Rhythm
- Melody
- Harmony

By analyzing through this lens, patterns in the music will become apparent. Similar patterns can be grouped into one segment, and contrasting ones can be grouped into others.

In addition to these elements, we can examine other characteristics of a piece such as perceived intensity. Does the piece have sections with clear dynamic contrasts? (loud vs. soft) Does it have noticeable textural changes? (more or fewer instruments playing at the same time)

Another common contrasting element between sections in a piece is the tempo. Many times, a piece of music will start out with a slow introduction and then transition into a faster more energetic tempo. Other times, pieces might sandwich a relatively calmer and slower section in between faster more energetic sections.

One final clue to delineating sections we'll consider is rhetorical reinforcement:

- Swoosh/crash (texture)

- Drum fills (rhythm)

- Turnarounds (harmony)

Storyboarding

Once you've mapped out your piece of music, start populating your scenes. Use your imagination and picture yourself or the performers up on stage to try and think of something that would look good and enhance the music/performance. Watch performances of artists you like and make observations about the visuals they use. Seek inspiration from other media online, in print, etc. Use paper and some kind of writing utensil to sketch out a rough idea of what you have in your head. While engaging in this brainstorming process, consider these parameters:

- Background/Midground/Foreground

- Busy/Calm; order/disorder; density

- Scale: consider the perspective of your audience. Don't necessarily limit your creativity based on an imagined restriction but do take into consideration how things will translate from your computer screen onto the projector/LED wall in the performance space. Audience may be far from the screen or viewing from an

angle, and resolution of the screen/LED wall may affect legibility of visual forms you create.

- Color Palettes

Consider the types of transitions you want between scenes. We will cover these strategies in class:

- Hard cut

- Crossfade

- Persistent virtual world where things fade in/out disappear, move away, etc.

Relationship Between Visual Forms and Musical Elements

- Amplitude

- Pitch

- Note/Musical Score Event

Later in the semester, we'll learn how to get this data from live performers, but you can still begin thinking about how you want to map or relate these elements to parameters of your visuals. Of course, one of the most interesting aspects of music is the transformation of musical material over time, so it is beneficial to identify how the evolution of musical elements can influence the evolution of these parameters as well. Here are some parameters to get you started:

- Characteristics of visual forms

    o Shape

    o Size

    o Color

    o Location

    o Speed

    o Persistence

    When these characteristics change over time, we can think of our visual forms as having "behaviors." How might a shape behave in the case of a higher amplitude vs. a lower one? Perhaps it moves slower or faster, or changes paths. Perhaps it glows brighter or darker, or changes colors entirely. Maybe it grows and shrinks or extends and retracts a limb.

<u>Assignment</u>: *Audio Analysis*

Answer the following:

Name of piece?

Artist?

Provide relevant program note information or additional context.

How will this piece be realized in performance?

    1. Live musicians?

    2. Computer playback/DJ style performance?

    3. Some combination of the two?

What is the form of the music, and how would you characterize the different sections?

    1. Is there a story this piece is trying to tell?

    2. Is there an overall affect/mood/vibe to the piece/sections?

How will this influence the visual narrative?

    1. How many scenes will there be?

    2. What will happen in the scenes?

    3. What kind of transitions do you envision between scenes?

4. Any color palettes come to mind?

5. How will the descriptors you chose to characterize the different sections of the piece apply to the visual? (Color, form, activity, transformation?)

Are there any musical aspects of the piece you want to relate to visual elements in these scenes?

1. Pitch Content

2. Rhythmic Content

3. Score position

4. Amplitude

5. Spatialization

6. Timbre

7. Interactivity

Week 2

<u>Coding Tutorials</u>

- *repetition & animation pt. 1-2*

<u>Readings</u>

- *FORM+CODE* chapters Repeat and Transform

- *Learning Processing* chapters 5-7

Week 3

<u>Coding Tutorials</u>

- *orbital motion / polygon algorithm*

- *coin toss*

<u>Readings</u>

- *Learning Processing* chapters 9 and 13

Week 4

<u>Coding Tutorials</u>

- *oop pt. 1-2.*

<u>Readings</u>

- *FORM+CODE* chapter Parameterization

- *Learning Processing:* chapters 7-8

Week 5

<u>Coding Tutorials</u>

- *udpsend~ & oscP5*

<u>Videos</u>

- *Live's user interface*

- *Session View*

- *Arrangement View*

- *Tracks*

- *Live's Browser*

- *Detail View*

- *Live's Mixer*

- *Setting up an audio interface*

- *Setting up MIDI*

- *Configuring your audio ins and outs*

<u>Readings</u>

- *Learning Processing:* chapter 12

<u>Lecture Notes</u>: *Introduction to Max*

- What is Max?

- What is a Patch?

  o Patcher

  o Inspector

  o Objects

  o Inlets/Outlets

  o Help Files/Reference

  o Keyboard Shortcuts

- Objects

  o bang

  o toggle

  o message

  o trigger

  o comment

  o number

- slider

- live.dial

- live.button

- live.toggle

- metro

- random

- int

- float

- +, -, /, *

- >, <, <=, >=, ==

- send

- receive

Lecture Notes: *Synchronization Strategies Part 1*

Today we'll examine two strategies for synchronizing the visuals you're creating with the music being performed. While each approach has its own advantages and disadvantages, you are not limited to using only one method in isolation but are free to balance your aesthetic/pragmatic needs and combine elements of each approach to fit your specific situation.

The first approach we'll learn is having a human performer trigger events in real time with the musical performers. In this way, the person controlling the visuals becomes another member of the ensemble, following and/or anticipating musical events with the musicians.

Open Processing and create a new sketch with a setup and draw functions, then add the keyPressed function. With this added to our sketch, Processing will now be monitoring for keyboard input. We can use if statements to see if specific keys have been pressed, and then execute whatever action we desire when the condition in that if statement is true.

So, let's imagine for example that we have a piece that is in ABA form, and you want the background of your visual to be red during the A sections and blue during the B sections. Inside of your keyPressed function, create two new if statements checking to see if the a or b keys have been pressed. In order to check for letter keys, make sure to put them between single quotation marks. Let's make a global color variable called bgColor and assign it a value of 255. Inside of our if statements checking for a or b, set bgColor to red and blue respectively. Run your sketch and play around with pressing the a and b keys. As you can see, you now have control over when the background changes from red to blue. With this, you can extend your code to trigger transitions between scenes, add or remove objects from your scene, increase or decrease parameter values in your sketch, etc.

Adding a performer is a great way to sync up your visuals with the music, but of course has its limitations. One person can only control so many actions/parameter values at a time, and the method of input also affects their ability to manipulate those controls. As a bridge to the next method of synchronization, let's look at how we might use a MIDI controller to control parameters in a Max patch that can send messages or values to our processing sketches.

In Ableton, we can plug in MIDI over USB devices which give us better physical control over our programs. Controllers with buttons, knobs, and sliders can be easily mapped to live.buttons and live.dials using the MIDI map function in Ableton. Additionally, using the midiin and midiparse objects, Max can easily grab the raw data from the controller and control bangs, sliders, etc.

In Live, create a new Max for live Audio effect and add a live.dial. Save the patch and close the editor. You'll see that the live.dial has appeared in the device, and when you activate the MIDI mapping mode in Live, it is highlighted blue, allowing you to select the dial, and then move the knob or slider on your controller to link them together. You could map a button to the dial, but it would only ever output a minimum or maximum value, generally 0 and 127 (on/off). From here, you can change the behavior of the live.dial to output whatever range of values you need using the inspector, and then send that data to your processing sketch using the udpsend object. If you want to send messages, you can use a live.button. This object is MIDI mappable and outputs a bang, which you can use to trigger a message to be sent using the udpsend object.

Our next synchronization strategy is going to be using the computer as a conductor. With this method, the performers will follow the click coming from Live, and you can use MIDI Clips and Automation in Live to control parameters of visuals. To do this, we're going to use the midiin and midiparse objects. The midiin object routes incoming MIDI information to our patch, and the midiparse object parses that information into useful formats. By hovering our mouse over each outlet, we can see what MIDI information comes from each. Let's start with the leftmost outlet, which reports the incoming note-on and note-off information in a list format. As you can see,

the first item in the list is the MIDI note number, and the second is the velocity. If we use a route object, we can separate each note.

The route object will take an incoming list and output the remainder of the information if the first item in the list matches the argument. So, if we create a route object with the argument 72, it will output the velocity of any C4 MIDI note that comes in. If you plug that route object into a bang, you can see you're getting two bangs - one for the note on and one for the note off. What if you only want to trigger a bang when you play the note, and don't care about how long you hold the note? In this case, we can use a change object, to output a bang when a value transitions from 0 to non-zero. With this bang triggered by MIDI note, you can now send messages, turn on/off metro objects, trigger random objects, etc. all synced to the piece the computer is conducting.

Next, let's take a look at how we might use the automation features in Live to control a live.dial. Using that same live.dial we created earlier, we can simply go into arrangement view and toggle on automation mode by clicking the icon in the top right. Click on the live.dial in the Max device and you'll see that the automation lane pops up in the channel. Alternatively, you can navigate to this lane by using the drop-down menus. From here, you can simply draw in the value changes you wish to happen over time, synced up with the correct place on the timeline.

You could also take parts of this approach and use it in the last one:

- Create a midi clip that has a collection of actions.
- Use the scene function in Live to trigger multiple MIDI clips simultaneously.

Week 6

- *images pt. 1-2*

Readings

- *Learning Processing:* chapter 15

Lecture Notes: *Synchronization Strategies Part 2*

- What is a score follower?

- What is antescofo~?

    o Setting up audio routing

    o Adding the antescofo~ object to your Max patch

    o Play/stop vs. mute/unmute

- Writing the antescofo~ score

    o BPM

    o NOTE

    o CHORD()

    o TRILL

    o MULTI

    o Max messages

- Audio Processing for antescofo~

    o Gate

    o EQ

Week 8

<u>Coding Tutorials</u>

- *images pt. 3-5*

Week 9

<u>Coding Tutorials</u>

- *illusion of motion pt. 1-3*

<u>Readings</u>

- *Learning Processing:* chapter 14

Week 10

<u>Lecture Notes</u>: *Max Intermediate Topics*

- urn, counter, delay, pipe
- Signal vs. data
    - snapshot~
- phasor~
    - Creating Saw, Square, Sine, and Triangle LFO's
- Sampling amplitude
    - ezdac~
    - Getting Audio from Live
        - plugin~
    - live.gain~
    - slide

- o zmap

- Formatting complex OSC messages

  - o Sprintf

Week 11

Outside Tutorial Resources

- Processing Examples: https://processing.org/examples

- Daniel Shiffman Coding Train: https://thecodingtrain.com

- thedotisblack channel: https://www.youtube.com/@thedotisblack

- Cycling '74 Tutorials: https://cycling74.com/tutorials

- Mari Kimura Max Starter: https://www.youtube.com/watch?v=JlqbweNGukE

- Ableton Live Tutorials: https://www.ableton.com/en/live/learn-live/

PHASE MUSIC: AN INTRODUCTION TO DESIGN AND FABRICATION

Course Description

Phase Music is a transdisciplinary course that integrates topics and approaches from music performance, theory, and history, computer science, and engineering. Students learn about Phase Music as it relates to Steve Reich, as well as its historical precursors; compose and/or perform a piece of phase music using Ableton Live; and learn to design and build a MIDI controller to aid in the performance of said piece. Using Arduino, Adobe Illustrator, and Max to build the MIDI controller will expose students to coding, circuitry, soldering, and laser cutting practices. Class time will be a combination of lectures, discussion, and supervised labs where students will receive feedback and guidance on their weekly assignments. The course will culminate with public performances of the works developed in class. As the nature of phase music lends itself to lengthier works, it is possible that some flexibility will be needed to accommodate everyone in the class getting a chance to perform their works during a dress rehearsal/studio class and a final performance. Weeks 12, 13, and 14 can be modified as needed.

Week 1: Phase Music and Deep Listening

Before class, students will be expected to read *Music as a Gradual Process* by Steve Reich[26] and *Quantum Listening: From Practice to Theory (To Practice Practice)*

---

[26] Steve Reich, "Music as a Gradual Process," accessed September 28, 2023, http://www.bussigel.com/systemsforplay/wp-content/uploads/2014/02/Reich_Gradual-Process.pdf.

by Pauline Oliveros,[27] which introduce the topics of Phase Music and Deep Listening. During class, the instructor will set work ethic expectations for the semester and provide an opportunity for the students to get to know one another, to foster a collaborative environment. Students will be briefed on the final project: composing and/or performing a piece of Phase Music using as custom-built MIDI controller. A lecture will more closely inspect the topic of Phase Music, delving into historical precursors, examining a case study of Piano Phase by Steve Reich, and contextualizing the Reich and Oliveros readings. It will also expose students to artists who have created and/or use custom controllers in their performances Following the lecture, students will have the opportunity to listen to and discuss one or more recordings of phase pieces. Remaining class time will be used as supervised lab, where students will work on this week's assignment: writing a response to one or more of the pieces of Phase Music presented and discussed in class and selecting or beginning to write a piece for the final performance. Additionally, the instructor will ensure that all students have been able to successfully install the necessary applications for completing the course: Ableton Live, Max, Arduino, and Adobe Illustrator.

Assignment

- Write a 250-word minimum response to one or more of the pieces of Phase Music presented and discussed in class, reflecting on emotional, intellectual, and physical reactions to the piece(s).

---

[27] Pauline Oliveros, "Quantum Listening: From Practice to Theory (To Practice Practice)," accessed September 28, 2023, https://s3.amazonaws.com/arena-attachments/736945/19af465bc3fcf3c8d5249713cd586b28.pdf.

- Decide what piece you will perform for the final project.

- Watch the following videos on Ableton's "Learn Live" webpage: *Live's user interface, Session View*, *Arrangement View*, *Tracks*, *Live's Browser*, *Detail View*, *Live's Mixer*, *Setting up an audio interface*, *Setting up MIDI*, and *Configuring your audio ins and outs.*[28]

- Read *Psychoacoustic Phenomena in Musical Composition: Some Features of a Perceptual Geography* by Maryanne Amacher.[29]

- Listen to at least one of the tracks from Amacher's *The Making of the Third Ear*.

Week 2: Ableton Live and Otoacoustic Emissions

Before class, students will be expected to watch ten short introductory videos on Ableton Live and read *Psychoacoustic Phenomena in Musical Composition: Some Features of a Perceptual Geography* by Maryanne Amacher, which introduces the concept of otoacoustic emissions. During class, this information will be reinforced and expanded upon during a lecture demonstration on how to compose and/or perform phase pieces using the Session View and zero-feedback delays in Ableton Live, and a discussion on the Amacher reading and how it pertains to composing phase pieces. Remaining class time will be used as supervised lab, where students will work on this week's assignment: setting up a Live set for their final project and demonstrating their proficiency in live looping.

---

[28] "Learn Live," accessed October 1, 2023. https://www.ableton.com/en/live/learn-live/.
[29] Maryanne Amacher, "Psychoacoustic Phenomena in Musical Composition: Some Features of a Perceptual Geography," accessed September 28, 2023, https://www.sonami.net/Articles/Amacher-OAE.pdf.

- Set up a Live set with the appropriate number of tracks, correct audio routing, and delays for the piece chosen for the final project.

- Demonstrate ability to capture, quantize, and loop audio using the session record button in Live.

Week 3: Circuitry

Class will begin with a lecture introducing students to the fundamentals of electronics and circuitry. The lecture will include basic information, teach how to read schematics, and go over common electrical components. Remaining class time will be used as supervised lab, where students will work on this week's assignment: breadboarding simple circuits.

Assignment

- Demonstrate understanding of fundamentals of electronics and circuitry by breadboarding the following circuits: light switch, visual doorbell, dimmer circuit controlling the brightness of a Light Emitting Diode (LED) using both a potentiometer and photoresistor, and a 555 Integrated Circuit (IC) oscillator.

Week 4: Arduino and Control Surface

Class will begin with a lecture introducing students to microcontrollers, specifically the Arduino platform. Students will learn about Arduino's Integrated Developer Environment (IDE), the form of a typical Arduino program, and the syntax rules that must be adhered to when coding for an Arduino. After the basics have been

introduced, students will learn about the Control Surface library for Arduino and how they will use it to create a custom MIDI controller. Remaining class time will be used as supervised lab, where students will complete this week's assignment: breadboarding some of the previous week's analog assignments using Arduino, and breadboarding a MIDI controller.

<u>Assignment</u>

- Breadboard a light switch, visual doorbell, and dimmer circuit using Arduino.

- Breadboard a MIDI controller with at least one potentiometer, rotary encoder, or photoresistor, one momentary push button, and one switch.

Week 5: Max and live.object

Class will begin with a lecture introducing students to the Max programming language. Students will learn about Max's programming environment and how to create and connect objects, messages, comments, bangs, and toggles. Two pieces of example code will demonstrate common objects and how they can interact with each other. Following the basic material, the lecture will introduce students to Max for Live, and teach how to use live.object to control parameters in Live using a Max for Live device. Remaining class time will be used as supervised lab, where students will complete this week's assignment: designing, breadboarding, and testing a prototype of their MIDI controller.

<u>Assignment</u>

- Write out the number and types of inputs the MIDI controller will include and list the parameters in the final project Live set that each one will control.

- Breadboard the MIDI controller and if necessary, write a Max patch that acts as an interface between the controller and the Live set using live.object~.

- Demonstrate that the MIDI controller prototype successfully modulates necessary parameters in the final performance Live set.

Week 6: Designing Laser Cut Enclosures and Adobe Illustrator

Class will begin with a lecture introducing students to designing laser cut enclosures. Students will be introduced to basic principles of design, learn how to balance aesthetic and ergonomic considerations with pragmatic limitations, and learn about common materials and methods of construction. Students will be given class time to sketch their designs on paper. After initial designs have been completed, a second lecture will introduce students to Adobe Illustrator, the vector art program they will use to plot their designs for the laser cutter. This lecture will introduce students to the program's basic features and outline the tools and suggested workflow for translating drawings into digital form. Remaining class time will be used as supervised lab, where students will work on this week's assignment: a first draft of enclosure plans.

Assignment

- Submit a first draft of enclosure plans in both paper and digital formats. This will include:
  - a written description of the enclosure and its fastener system,
  - a list of all components that will be mounted on or in the enclosure,
  - sketches of the enclosure in perspective, front, side, back and inside views,

o   and an Adobe Illustrator file that contains all panels and parts with their appropriate joinery, holes, and etching.

Week 7: Creating Test Pieces

Class will begin with a lecture on creating test pieces for the laser cutters that students will use to create their enclosures. Students will learn about tolerance in manufacturing, and how this will affect their designs. After designing a test piece that encompasses all the joints, holes, and etching contained in their designs, students will use a laser cutter to create their test piece and document what is working and what parts of the enclosure design need to be iterated upon. Remaining class time will be used to update designs, or in cases where no additional changes are needed, begin cutting enclosures.

Assignment

- Create a test piece in Adobe Illustrator that includes all joints, holes, and etching contained in enclosure design, then cut it out using the laser cutter.
- Test the fit of joinery and holes for electronic components.
- Document any changes that need to be made to the original design and update the Illustrator file that contains the enclosure plans.

Week 8: Soldering and Circuit Building

Class will begin with a lecture demonstration teaching students how to solder circuits onto generic perforated circuit boards. This will include best practices and safety precautions for soldering, as well as how to cut and strip wire, mount components, and

use "helping hands" tools. Remaining class time will be used as supervised lab, where students will complete this week's assignment.

<u>Assignment</u>

- Design a layout for and solder together a 555 IC "theremin" on generic perforated circuit board.

Week 9: Planning and Starting a Successful Build

Class will begin with a brief lecture on setting yourself up for a successful build. Students will learn the importance of creating a bill of materials and compiling all drawings, schematics, Illustrator files, enclosure materials, and electronic components necessary to create their MIDI controllers before beginning the build. Remaining class time will be used as supervised lab, where students will work on this week's assignment.

<u>Assignment</u>

- Create a bill of materials and compile all drawings, schematics, Illustrator files, enclosure materials, electronic components, fasteners, and tools for the build.
- Write out the order in which build tasks will be completed.
- Begin cutting and assembling enclosures.

Week 10: Soldering Internal Components

Class time will be fully devoted to supervised lab, where students will finish cutting and assembling their enclosures and begin soldering together the internal components of their MIDI controllers.

Assignment

- Finish enclosures.

- Begin assembling internal components.

Week 11: Completing Assembly and Testing of MIDI Controllers

       Class time will be fully devoted to supervised lab, where students will complete the assembly of their MIDI controllers and begin testing them to ensure they are working with the Live set they created for the final performance. Students who have a complete and working controller will begin rehearsing their final performances. The goal is for the final project to be more or less complete by this week so that students have two weeks to rehearse and perform their pieces at home and in studio classes/dress rehearsals before the final performance.

Assignment

- Complete assembly and testing of MIDI controllers, ensuring that everything works as intended with the final project Live set.

- Document any issues or bugs that arise and begin resolving them.

- Run through and practice final project piece every day.

Week 12: Flex Week – Lab, Studio Class, or Dress Rehearsal

       Outside of class, students will be encouraged to run through and practice their pieces every day leading up to the final performance. Supervised lab class time will tie up loose ends on MIDI controllers, and address any issues discovered during testing. Studio class time will provide each student or student group an opportunity to perform their

40

piece and receive constructive feedback from the instructor and their peers. Dress rehearsal class time will provide an opportunity for students to perform their piece in the performance space. Studio class and dress rehearsal performances will be recorded for students to review.

<u>Assignment</u>

- Run through and practice final project piece every day.
- If applicable, watch recording of studio class/dress rehearsal and compile list of feedback from self, peers, and instructor.
- Implement desired feedback.

Week 13: Flex Week – Studio Class or Dress Rehearsal

Outside of class, students are encouraged to run through and practice their pieces every day leading up to the final performance. See Week 12 for studio class and dress rehearsal descriptions.

<u>Assignment</u>

- Run through and practice final project piece every day.
- If applicable, watch recording of studio class/dress rehearsal and compile list of feedback from self, peers, and instructor.
- Implement desired feedback.

Week 14: Flex Week – Dress Rehearsal or Final Performance

Outside of class, students are encouraged to run through and practice their pieces every day leading up to the final performance. See Week 12 for dress rehearsal description. See Week 15 for final performance description.

Assignment

- Run through and practice final project piece every day.

- If applicable, watch recording of studio class/dress rehearsal and compile list of feedback from self, peers, and instructor.

- Implement desired feedback.

Week 15: Final Performance

Outside of class, students are encouraged to run through and practice their pieces every day leading up to the final performance. In lieu of a class meeting, there will be a final performance featuring all the students and the work they developed over the semester.

Assignment

- Give final performance.

PHASE MUSIC: SUPPLEMENTAL MATERIALS

Week 1

<u>Readings</u>

- *Music as a Gradual Process* by Steve Reich

- *Quantum Listening: From Practice to Theory (To Practice Practice)* by Pauline Oliveros

<u>Examples of Artists Using Custom Controllers</u>

- Pamela Z: https://www.youtube.com/watch?v=ebxvVJwGWek

- Laeticia Sonami: https://www.youtube.com/watch?v=C8GqbS2w_Lg

- Mari Kimura: https://www.youtube.com/watch?v=hnEJNhPRhYg

- Imogen Heap: https://www.youtube.com/watch?v=3QtklTXbKUQ

- Michael Waisviz: https://www.youtube.com/watch?v=SIfumZa2TKY

Week 2

<u>Readings</u>

- *Psychoacoustic Phenomena in Musical Composition: Some Features of a Perceptual Geography* by Maryanne Amacher

<u>Listening</u>

- *The Making of the Third Eye*, Maryanne Amacher

<u>Videos</u>

- *Live's user interface*

- *Session View*

- *Arrangement View*

- *Tracks*

- *Live's Browser*

- *Detail View*

- *Live's Mixer*

- *Setting up an audio interface*

- *Setting up MIDI*

- *Configuring your audio ins and outs*

Week 3

<u>Lecture Notes</u>: *Introduction to Circuits*

- What is Circuitry?

  - Flow of electricity

  - Positive/negative charges

  - Completing circuits so electricity can flow

- Reading Schematics

- Common components

  - Switches

  - Buttons

    - Momentary and latching

  - Resistors

  - Capacitors

  - LED's

- Potentiometers

- Photoresistors

- 555 IC

- How to prototype and test circuits using a breadboard

Week 4

Lecture Notes: *Introduction to Arduino and Control Surface*

- What is a microcontroller?

  - "Mini-computer"

  - Digital and analog input/output

- How to code in Arduino

  - IDE

  - Syntax

  - If statements

- Recreating analog circuits from Intro to Circuitry lecture

- Control Surface Library

  - Importing Libraries

  - MIDI over USB

- Designing a MIDI Controller with Arduino

  - What components will you include?

  - Do you need/want visual feedback?

Week 6

<u>Lecture Notes:</u> *Designing Laser Cut Enclosures*

- What is a laser cutter and how does it work?

- The "Box"

    o Front (top) Panel

        ▪ Holes for components

        ▪ Labeling

        ▪ Visual designs

    o Sides

        ▪ Holes for input/output

    o Back (bottom)

    o Dimensions

        ▪ Need enough space to fit internal components

- Typical designs for joining pieces together

    o Gear-like tooth design

    o Through-hole

        ▪ with pins

        ▪ with fasteners

    o Glue & machine screws

    o Right angles are the easiest, but what if you want an angled front panel?

    o Freeform designs by stacking layers

    o How to plot designs efficiently on the laser cutter to minimize wasted

    material

- o Simpler designs have less room for error, but you might want rounded corners…

    - ▪ Aesthetic

    - ▪ Ergonomic

    - ▪ UX

- How to mount front panel so that it is removable?

    - o Internal components can be serviced/added to as needed.

- Draw your box on paper first.

    - o Write a list of panels

    - o List holes needed for mounting components

    - o Describe construction and fastener system(s)

Lecture Notes: *Introduction to Adobe Illustrator*

- What is Vector drawing and how does it differ from raster images?

    - o What are the advantages of this and why are we using it?

- How to create a new Illustrator File

    - o Menus

    - o Tools

    - o Windows

- Grid/Snap

    - o Setting up the grid for designing enclosures

- Fill/Stroke settings

- Creating Paths

- Primitive Shapes Tool

- Dimensions tool

- Text

- Importing/converting Images

Week 7

Lecture Notes: *Creating Test Pieces*

- What is tolerance in manufacturing and how will it affect my build?

- Laser cutter precision

- Quality of materials/components

  o MDF is cheap and readily available but can be warped over larger surface
    areas compared to other cuts of wood

  o Even in the same factory in the same batch, components may have slightly
    different measurements

  o Taking measurements of through-hole components with calipers

- Designing test pieces to verify measurements/laser cutter settings

  o Joinery

  o Holes for electronic components

  o Etching

BIBLIOGRAPHY

Amacher, Maryanne. "Psychoacoustic Phenomena in Musical Composition: Some Features of a Perceptual Geography." Accessed September 28, 2023. https://www.sonami.net/Articles/Amacher-OAE.pdf

Cook, Perry R. "Principles for Designing Computer Music Controllers." *arXiv.org* (2020). DOI: 10.5281/zenodo.1176358

Fasciani, Stefano and Jackson Goode. "20 NIMEs: Twenty Years of New Interfaces for Musical Expression." *NIME 2021*. https://doi.org/10.21428/92fbeb44.b368bcd5

Filimowicz, Michael, and Veronika Tzankova, eds. *Teaching Computational Creativity*. Cambridge: Cambridge University Press, 2017.

"Learn Live." Accessed October 1, 2023. https://www.ableton.com/en/live/learn-live/

"Max 8 Documentation." Accessed October 1, 2023. https://docs.cycling74.com/max8

Oliveros, Pauline. "Quantum Listening: From Practice to Theory (To Practice Practice)." Accessed September 28, 2023. https://s3.amazonaws.com/arena-attachments/736945/19af465bc3fcf3c8d5249713cd586b28.pdf

"Processing Reference." Accessed October 1, 2023. https://processing.org/reference/

Reas, Casey, Chandler McWilliams, and Jeroen Barendse. *Form+Code in Design, Art, and Architecture*. New York: Princeton Architectural Press, 2010.

Reich, Steve. "Music as a Gradual Process." Accessed September 28, 2023. http://www.bussigel.com/systemsforplay/wp-content/uploads/2014/02/Reich_Gradual-Process.pdf

Rodgers, Tara. *Pink Noises: Women on Electronic Music and Sound*. Durham: Duke University Press, 2010.

Rowe, Robert. *Interactive Music Systems: Machine Listening and Composing*. Cambridge: MIT Press, 1993.

Shiffman, Daniel. *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction.* 2nd ed. San Diego: Elsevier Science, 2015.

APPENDIX A

LINKS TO CODING TUTORIALS

1. *3 shapes 4 colors pt. 1*: https://youtu.be/JIlbdna9kbM

2. *3 shapes 4 colors pt. 2*: https://youtu.be/YN5cNqfFkpQ

3. *3 shapes 4 colors pt. 3*: https://youtu.be/uurQ2mKO--c

4. *repetition & animation pt. 1*: https://youtu.be/b2Ko8d68Kps

5. *repetition & animation pt. 2*: https://youtu.be/agpT1BIBZXY

6. *orbital motion / polygon algorithm*: https://youtu.be/x2h_bfFVNA0

7. *coin toss*: https://youtu.be/eo2cYqZGhVM

8. *oop pt. 1 – ripples*: https://youtu.be/bM5rbU9XqOM

9. *oop pt. 2 – shrinking shapes*: https://youtu.be/Ob3DocpQkqw

10. *images pt. 1 – basics*: https://youtu.be/C_tDtJOhEfs

11. *images pt. 2 – transparency & masking*: https://youtu.be/fRe0R3cjDEs

12. *images pt. 3 – pseudo slit scan*: https://youtu.be/59_DCLypYqc

13. *images pt. 4 – funhouse mirror*: https://youtu.be/TLJpf00gNeg

14. *images pt. 5 – kaleidoscope*: https://youtu.be/IdI9faZkgIw

15. *illusion of motion pt. 1 – TRON floor*: https://youtu.be/NxwwNyH3Y8I

16. *illusion of motion pt. 2 – endless tunnel*: https://youtu.be/EVyaRRCZyLM

17. *illusion of motion pt. 3 – star field*: https://youtu.be/rAtZtTxT9tQ

18. *udpsend~ & oscP5*: https://youtu.be/kRKxnEYFqow