

## **MAT 240B "Digital Audio Programming: Spatial Sound"**

MF 12-2, 4 Units

Elings 2003 (MAT Conference Room)

Instructor: Dr. Matthew Wright [matt@create.ucsb.edu](mailto:matt@create.ucsb.edu)

Office Hours TBA

TA: Ryan McGee <[ryan@mat.ucsb.edu](mailto:ryan@mat.ucsb.edu)>

Lab/TA Office Hours Mondays 2-3:30 and Thursdays by Appt.

### **Description**

Advanced theory and practice of digital audio research focusing on real-time sound spatialization, including VBAP, DBAP, Ambisonics, Wave Field Synthesis, Head-Related Transfer Functions, sound propagation delay, doppler shift, and psychoacoustics of spatial sound perception. Students will learn current C/C++ audio paradigms with an emphasis on state-of-the-art environments used at UCSB, particularly in the AlloSphere Research Group. Fundamental concepts from 240A including I/O, oscillators, spectral processing, delay lines, and OSC control will be reintroduced in the context of C/C++. For students with limited C++ background, this will also be an opportunity to learn object oriented programming, pointers, and the mechanics of installing libraries and building C/C++ projects.

### **Online Resources**

Class mailing list: [240@mat.ucsb.edu](mailto:240@mat.ucsb.edu)

Please sign up: <http://lists.mat.ucsb.edu/mailman/listinfo/240>

Web site (currently out of date): <http://www.mat.ucsb.edu/240>

Class wiki TBA

### **Philosophy**

Digital audio programming, like most interesting topics, has deep aspects and superficial aspects. The goal of the course is to teach the deep aspects: principles, concepts, techniques, and theory. But to do a hands-on project and actually implement something you must also master many superficial aspects: specific programming languages, syntax, alternate names of things, where you need to put your files, etc. I learned digital audio programming 20 years ago with a substantially different set of tools than what I use today, and so inevitably much of my hard-won superficial knowledge is now obsolete. But the deep knowledge is still relevant, useful, and essential to my work. If you understand the deep aspects then you can always easily learn new superficial aspects just by looking them up. But without the deep knowledge you might not even be able to understand the documentation you find.

I see a large part of my role as instructor of this course to be a sort of guide that points out what is deep and what is superficial. I will make every effort to teach the former and I will encourage you to learn the latter on your own, from each other, and from the TA. If you try to learn something superficial and it's really hard then there's probably something deep we need to talk about; please don't be shy with questions.

## Software

With that said, this quarter we happen to have a remarkable historic opportunity to participate in some state of the art research. There is a fortunate confluence of events right now. The AlloSphere Research Group, with lessons learned from the first few years of developing AlloSphere content, has created *AlloCore*, a C++ library at the center of the shared functionality of the software that creates various real-time interactive, 3D audiovisual immersive virtual worlds. AlloCore has reached an initial level of usability and we are adding a lot to it as we go along. At the same time, Ryan McGee has just completed the very powerful Sound Element Spatializer software, based on the older CSL libraries. Bringing these together will result in a very flexible, powerful system that we and many others in the future will be able to use to spatialize content in the AlloSphere and elsewhere.

Many of the details and rough edges of AlloCore involve shallow knowledge that may be especially challenging and/or frustrating; that's often what it's like doing research on "the bleeding edge."

If you do not have your own computer (or for any other reason) you may do all programming on the computers in CREATE studios Xenakis (Music 2215) and Theremin (Music 2203) and possibly others.

## Readings, Homework, and Labs

We will assign required readings each week, generally 1-2 short academic articles. Each will be due for a particular class and for each you will email Matt and Ryan the following **reading summary assignment** before 10pm on the night before the class:

1. One to four sentence summary, in your own words, of the article.
2. Something you like about the article (interesting, useful...) and why.
3. Something you either don't understand about the article (confusing, incomplete, knowledge you lack...) or don't like about the article (something you think could have been done better) and why.

We will also recommend additional optional readings if you need additional background information or depth.

Most weeks will also have two types of homework:

*Trivial Homework* helps ensure that you understand the important basic concepts and techniques and that you have at least minimal proficiency with the software we will be using. These are for pedagogical value only and should therefore take a short amount of time and/or teach you something useful.

*Nontrivial Homework* uses the same concepts and techniques to build something useful, novel, and/or interesting. We will suggest a possible project but you are encouraged to adapt it or completely substitute an equivalent that is more in line with your final project and/or overall goals.

## **Tentative Schedule**

### **Week 1 3-Jan, 7-Jan**

Monday (MW) - Intro to spatial hearing, perception, cognition. Scope of spatial sound and of this course. AlloSphere field trip.

**Extra Session 4pm Thursday 6-Jan (2809 Elings):** Fundamentals of C++ (arrays, pointers, classes, functions, control flow, header files, compiling and linking). This session is optional.

Friday (RM) – Review xcode setup and portaudio installation, explain demos, explain multichannel in the context of audio callbacks.

Reading #1 due 10pm Thursday 6-Jan - Blauert (excerpt), Eric Newman's C++ Notes (optional)

Trivial HW #1 due 10pm Thursday 6-Jan: Join the 240 email list. Figure out what computer(s) you'll use for programming assignments. Install PortAudio on your computer, build, and run the examples.

Nontrivial HW #2 due 10pm Thursday 13-Jan: Modify PortAudio Demos and Implement Simple Panning in C++

### **Week 2 10-Jan, 14-Jan**

Monday (RM) - Overview of panning algorithms: VBAP, DBAP, Ambisonics, WFS, HRTF/Binaural. Other elements of spatial sound programming: distance cue (attenuation, air absorption, delay, reverberation), intensity vs. loudness, localization of early echoes.

Friday (MW) Usage of AlloSphere's WFS array: Physical layout, wiring, CoreAudio aggregate devices, launching demo software.

Reading # 2 - Baalman - Spatial Comp Tech., Chowning- Simulation of Moving Sound Sources

Nontrivial HW #3 - 1-2 page essay on sound spatialization techniques. Pick 3 panning algorithms (VBAP, HOA, etc) and describe the strengths and weaknesses of each. Give at least one example in which each algorithm is used (a software package, installation, hardware, etc). Besides panning, what else do you need for effective spatialization?

### **Week 3 (No class 17-Jan: MLK Holiday) 21-Jan**

Friday (MW, RM) - OSC intro, liblo installation and usage, introduction and usage of SES, discussion of possible trajectory controllers (hardware and software)

Reading #3 – Roads – “Articulating Space”, Schmeder- “Best Practices for OSC”

Trivial HW #4– We will give you three simple programs that send 2D OSC messages to SES, one in each of these environments: Max, Processing, C++. Pick one and modify it to send 3D trajectories. Add user interface components to control the position of the sound source. (We will provide more specific instructions.)

### **Week 4 24-Jan 28-Jan**

Monday (RM) - Composing with Spatialization, Complex Trajectory Control (flocking algorithms, boid, sharing space with visual systems, mobile devices, surface computers). Decorrelated upmixing strategies.

Friday (RM, MW) – Particle Systems as Trajectory Controllers in Processing, Sending OSC with liblo, Brainstorm final projects

Nontrivial HW #5 – In the environment of your choice, create some form of complex trajectory control that supports multiple moving sound sources. Use SpatDIF OSC messages to be compatible with SES. Projects will be presented in the AlloSphere and Pluriphonic labs next week.

**Wednesday Noon CONCERT** 26 Jan 2011, 12 PM, Music Bowl, free.  
UCSB AfroBrazilian Ensemble (dir. Matt Wright).

### **Week 5 31-Jan, 4-Feb**

Monday (RM) – Spatialization models in the CREATE Signal Library (CSL). Programming concepts behind SES.

Friday (MW, RM) - Introduction to AlloCore. Current AlloSphere spatial tools (non-SES): ambisonics, source delay lines, data structures for dynamic objects in 3D space, viewer/camera/avatar models. Version control with Git.

Nontrivial HW #6: Final Project Proposal (1-2 pages)

### **Week 6 7-Feb, 11-Feb**

Individual meetings with students (30 minute sections during class time) to discuss final projects.

### **Week 7 14-Feb, 18-Feb**

Projects (class time for workshops, progress reports, and additional lecture topics by request)

### **Week 8 (No class 21-Feb: President's Day Holiday) 25-Feb**

Projects (class time for workshops, progress reports, and additional lecture topics by request)

### **Week 9 28-Feb, 4-Mar**

Projects (class time for workshops, progress reports, and additional lecture topics by request)

**AlloSphere Event 28 Feb: TED pre-conference event at the AlloSphere. Details TBA.**

### **Week 10 7-Mar 11-Mar**

Dead week: review/help sessions, no new material. Possible early final project presentations.

**CONCERT** Thursday 10 March 2011, evening, Muddy Waters Coffeehouse (508 East Haley Street, Santa Barbara). Santa Barbara New Music Series featuring the CREATE Ensemble (dir. Matt Wright).

### **Final Exam Time(s): Weds March 16 12-3pm and Thurs March 17 4-7pm**

Final Presentations. (No final exam.)

(The final exam time is a function of the weekly class meeting time. Since this class meets for two hours there are two possible final exam times. We will use one or both of these time slots, TBD later in the quarter.)

### **Final project**

Written portion: Add something substantial to the wiki documentation about our system.

Programming portion (any combination of below):

- Do a substantial part of the port
  - coding
  - design (e.g., translate CSL's spatial sound object model to AlloCore)
  - compatibility debugging
  - documentation
- Write a really useful OSC trajectory control app
- Write a really useful multichannel audio source generator (decorrelated upmixing)
- Write a really useful tool (source management interface, etc..)
- Touchtable as trajectory interface for spatial sound?

Ideally your final project will relate to a larger goal: research, composition, education, etc. Team projects encouraged as long as each member has a worthy, well-defined individual contribution and the whole is not much less than the sum of its parts.

## **Grading**

25% Participation (attendance, reading, contributing to class)

25% Homeworks

50% Final project