

BJÖRN HARTMANN TEACHING STATEMENT

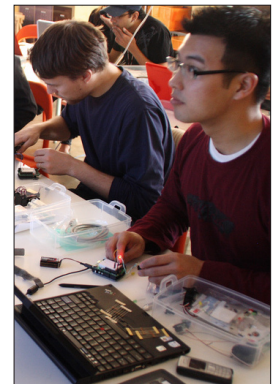
I first became fascinated by Computer Science research when my cohort of undergraduate Digital Media Design students was invited to accompany researchers to the SIGGRAPH conference in 1999. My most memorable class experiences were studio work and class critiques in the Fine Arts department, which taught me the value of shared work spaces and frank, constructive feedback. And I still take pride in Art History papers that I wrote, rewrote, then rewrote again in college. From my background, I draw three core pedagogical beliefs: to *foster undergraduate research* through apprenticeship; *encourage experiential learning* through studio-based, project-centric courses; and *emphasize writing* as a core academic skill.

As a teaching assistant for seven different courses, I have covered areas from graphic design to programming languages. In 2006, my service was recognized with Stanford's *Centennial Teaching Assistant* award. For four summers, I have mentored undergraduate students in full-time research internships. In addition, I have supervised independent study projects, taught electronics workshops and delivered guest lectures in HCI and Fine Arts. This diverse set of opportunities has shown to me that teaching well is hard work, but also immensely rewarding.

Three teaching appointments demonstrate the breadth of technical, practical, and theoretical teaching perspectives I have learned from. At Stanford, I was most strongly influenced by the *Human-Computer Interaction Design Studio* co-taught by Scott Klemmer and Bill Verplank. Students are led through a series of short team projects to become intimately familiar with iterative, user-centered design of interactive systems. As part of the course, I took students to the San Francisco farmers' market to practice need finding and contextual inquiry in situ; I planned weekly tutorials with experts to quickly build technical skills; and evaluated work through review of idea log books and prototype critiques. Based on the tutorials, I also developed online workshop materials for physical computing, which I subsequently taught to HCI students and at Stanford's Institute of Design.

For Terry Winograd's course on *Phenomenological Foundations of Cognition, Language, and Computation*, I learned how to prepare and discuss challenging primary texts from Artificial Intelligence, Cognitive Science, and Philosophy. I reviewed and graded proposals, drafts and final term papers. I am looking forward to teach a similar course based on broad reading and careful development of written arguments. More broadly, I am interested in conveying the history of human-computer interaction and the associated debates to understand possible alternatives to our technological status quo.

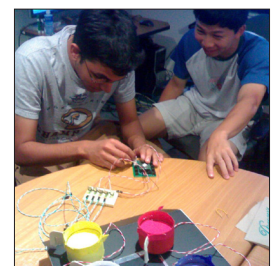
As an undergraduate at the University of Pennsylvania, I assisted for Steven Lane's course on *Advanced Topics in Computer Animation and Robotics* which introduced students to forward and inverse kinematics for character animation and control systems for dynamic simulations and flock behaviors. This technical course gave me experience in explaining complex and conceptually challenging



Workshop on physical computing at Stanford's Institute of Design.



Students visit the San Francisco farmers' market as a need finding exercise for the HCI Design Studio class.



Undergraduates Nundu Janakiram and Nan Gao assemble electronics for their HCI Design Studio project, a tangible color mixing board.

subject matter to students through problem sessions, one-on-one tutoring and preparation of explanatory material to accompany lectures.

MY MOST MEANINGFUL MENTORING EXPERIENCES were as an advisor for Stanford's CURIS program in which undergraduate students spend the summer working in close collaboration with PhD students. These internships have contributed significantly to four out of five of my major research projects. Undergraduate students were involved in all aspects of research – from brainstorming to literature searches, from implementation to evaluation. CURIS students have appeared as coauthors on our most visible papers at CHI and UIST, the premiere publishing venues for systems work in human-computer interaction.

In 2005, Michael Bernstein, a Symbolic Systems major, wrote the core of the user interface for the d.tools visual authoring environment with me through daily pair programming. Michael subsequently decided to pursue a PhD in Computer Science at MIT CSAIL. Two years later, I worked with Loren Yu on *Juxtapose*, a novel editing environment that enables interaction designers to experiment with multiple alternative designs in parallel. Loren constructed a design space of interaction prototyping tools, helped prototype early ideas in Java and Processing, and contributed significantly to Juxtapose's code editor. Loren's research was recognized with an Outstanding Undergraduate Honorable Mention by the Computing Research Organization.

Beyond campus, I have engaged the larger community of interaction designers and technology hobbyists through demonstrations at the Maker Faire, tutorials at Silicon Valley Code Camp, and lectures at professional conferences and workshops such as MX (Managing Experience Design) and Sketching in Hardware. I look forward to continuing these bridging activities between academia, industry, and hobbyists.

PROPOSED COURSES

I look forward to teaching the following undergraduate and graduate courses:

Human Computer Interaction

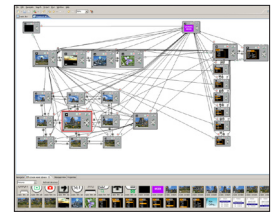
A practice-oriented course that focuses on experiential understanding of user-centered design. Involves real-world need finding, use of sketching and lo-fidelity prototyping, studio critique to guide iteration, as well as formal and informal evaluation techniques.

Programming Languages

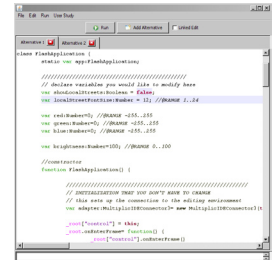
A course introducing students to concepts in programming languages, covering history, functional programming, object-oriented programming, and language implementation strategies.

Ubiquitous Computing Laboratory

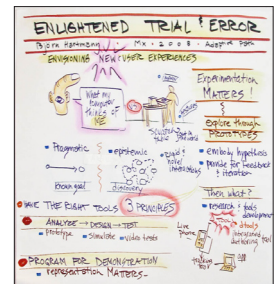
Focus on rapid prototyping and iterative design of functional interactive artifacts. Electronics for sensing and actuation; vision-based systems and algorithms; co-location of input & output (e.g., multi touch displays); rapid fabrication methods (laser cutting, 3D printing, aluminum framing).



Michael Bernstein contributed GUI interactions for our research project d.tools.



Loren Yu contributed code editing and differencing techniques for our research project Juxtapose.



Graphic recording of a survey talk I presented to design managers at the MX conference.

HCI Research Survey

An intensive introduction to current research literature. Students read four to six papers per week and submit written responses. Students also produce an in-depth literature review of a chosen area. To merge theory with practice, student teams rebuild systems described in seminal papers, and propose ways to expand on this existing work.

Software Architectures for Interaction

An advanced graduate course on toolkits and architectures for constructing interactive user interfaces. Event-based architectures, declarative UI languages, split client-server architectures, mobile development frameworks. In pairs, students carry out research that extends the state of the art.

Theories of Presence and Practice for HCI

Advanced reading seminar — students read and prepare responses to primary literature on phenomenology, distributed cognition, tacit knowledge, and communities of practice. A term paper requires students to critically respond to contemporary writing in light of the course readings.