Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2009), *23*, 251–261. Printed in the USA.
Copyright © 2009 Cambridge University Press 0890-0604/09 $25.00
doi:10.1017/S0890060409000225

Tangible user interface laboratory: Teaching tangible interaction design in practice

ORIT SHAER,[1] MICHAEL S. HORN,[2] AND ROBERT J.K. JACOB[2]

[1]Department of Computer Science, Wellesley College, Wellesley, Massachusetts, USA
[2]Department of Computer Science, Tufts University, Medford, Massachusetts, USA

(RECEIVED April 23, 2008; ACCEPTED December 30, 2008)

Abstract

Tangible interaction is an emerging field of human–computer interaction that links the digital and the physical worlds by embedding computation in physical artifacts and environments. This paper shares our experience teaching tangible interaction over the past 4 years in an interdisciplinary, project-based laboratory course at Tufts University. Although the course is offered through the Computer Science Department, it reflects the multidisciplinary nature of the field, merging product engineering practices with a design studio approach. With a diverse mix of students, this approach has fostered creativity and hands-on learning. Throughout the course students have created innovative interfaces that not only capture fundamental concepts of tangible interaction but also contribute novel techniques for supporting collaborative design. We discuss examples of student-created interfaces and illustrate the relationship between the methods employed in the course and the artifacts created. We also share our recommendations for implementing such a course in institutions with constraints similar to ours including a limited budget and minimal laboratory space.

Keywords: Design; Education; Human–Computer Interaction; Instruction; Laboratory; Tangible User Interfaces

1. INTRODUCTION

In the last decade, tangible interaction research has gained visibility within the human–computer interaction (HCI) community, showing promise to support activities such as learning, problem solving, and design. As the field matures, a growing worldwide community of researchers is actively exploring new application domains, technologies, and evaluation techniques. However, there is still relatively little discussion of how best to teach tangible interaction design to the next generation of practitioners.

Building a tangible user interface (TUI) is a complex process that encompasses multidisciplinary knowledge including engineering, art, and social sciences. Successful design depends on many factors including physical form, social settings, and aesthetics, in addition to well-designed software and electronics. In this paper we share our experiences teaching an interdisciplinary, project-based laboratory course over a period of 4 years at Tufts University. The course seeks to provide students with the foundations required for exploring the future of tangible interaction. Thus, it draws on a wide range of disciplines, providing students not only with technical skills required for building TUIs, but also with conceptual models and methodologies to support the synthesis of new ideas. The following four goals guided our design of the course: to facilitate hands-on learning, to reinforce the "big ideas" of HCI, to encourage innovation, and to promote interdisciplinary collaboration.

1.1. Facilitate hands-on learning

Education research emphasizes that learning often occurs when people are engaged in designing and building personally meaningful artifacts (Papert, 1980; Resnick & Ocko, 1991). Based on this constructionist approach, we engage students in designing and constructing TUIs of their own creation, while they acquire skills and knowledge in a diverse array of subjects.

1.2. Reinforce the big ideas of HCI

Several ideas are fundamental to the field of HCI, including iterative and user-centered design, interaction styles, and design principles such as simplicity, consistency, visibility, affordance, and feedback. By reinforcing these ideas through design

Reprint requests to: Orit Shaer, Science Center, 106 Central Street, Wellesley College, Wellesley, MA 02481, USA. E-mail: oshaer@wellesley.edu

exercises, critiques, and project work, the course provides students with a conceptual framework for understanding and evaluating TUIs within the broader context of HCI.

1.3. Encourage innovation

Tangible interaction is an emerging field of research, and building novel tangible interfaces requires not only applying existing concepts and practices but also exploring new ideas, domains, and techniques. To accomplish this goal, we encourage TUI Laboratory students to take risks and experiment with new ideas and technologies. We use structured brainstorming exercises throughout the first half of the course and encourage students to explore the boundaries of the tangible interaction paradigm. To focus and motivate students' design efforts, each year we choose a theme centered on a real-world application domain. For example, our most recent theme is interfaces for informal science learning.

1.4. Promote interdisciplinary collaboration

Designing and building TUIs requires cross-disciplinary knowledge. Thus, we encourage the enrollment of students from diverse backgrounds. In addition to a unique set of skills and individual creativity, every student brings field-specific terminologies and work practices to his or her team. In the course, we emphasize the importance of interdisciplinary communication, helping teams develop a cohesive vision while exploring a wide range of solutions across and between disciplines. In pursuit of these goals, the course is offered through the Computer Science department to students from a variety of disciplines.

Below we describe related courses from other institutions that inspired our course. Then, to give a sense of the scope of work accomplished by the students, we describe several resulting projects. Next, we describe the course itself and show how the methods applied in the course affected the resulting projects. Finally, we discuss lessons learned from students' evaluations and our own observations, and share our recommendations for implementing such a course in institutions with similar constraints and culture. We conclude with our vision for the course's future directions.

2. RELATED COURSES

Although inspired by the Physical Computing Studio taught in the Design Machine Group at the University of Washington (Camarata et al., 2003) and the tangible interfaces class taught in the MIT Media Lab, our course was designed and taught under a different set of constraints and academic culture. First, our course was offered within an engineering department that is unfamiliar with the studio culture we sought to cultivate in this course. Several initiatives that foster studio culture in HCI and computing guided our efforts in this regard (Klemmer et al., 2005; Yi-Luen Do & Gross, 2007; Turbak & Berg, 2002). Second, constraints such as the lack of a dedicated teaching lab and a limited budget led us to select implementation technologies that are portable, cross-platform, and open source. For example, although advanced development platforms for tabletop interactions such as Sensetable (Patten et al., 2001) are available to MIT Media Lab students, our students use computer vision techniques to implement tabletop interaction. The lack of a dedicated workspace for students also impacts the scale of students' projects. Rather than large artistic installation projects often seen in other courses (e.g., Camarata et al., 2003), our students tend to produce projects that are portable and smaller in scale, as students need to physically move their projects from our lab to the classroom. Finally, another aspect that distinguishes our class from other offerings, such as the introduction to physical computing course offered in the ITP program at NYU or the physical computing class at the University of Washington (Camarata et al., 2003), is the use of a heterogeneous set of implementation technologies for student projects. Our decision to use diverse technologies aims to allow students to explore a large subset of the tangible interaction design space. To increase knowledge transfer between students, each student works with at least three technologies that they present to the class in teams. These differences and the rationale behind them are the focus of this paper.

3. STUDENT PROJECTS

In this section, we briefly describe five innovative student projects: one from each semester the course was offered. These projects highlight the diversity of class projects, both in terms of implementation techniques and design concepts. The projects also illustrate potential benefits of tangible interaction for collaborative authoring and design activities. Students continued work on three of these five projects beyond the end of their class and published papers in the International Conference on Tangible and Embedded Interaction (TEI).

3.1. The Tangible Video Editor (TVE)

The TVE (Spring 2005) project is a design tool for collaborative editing of digital video (Fig. 1; Zigelbaum et al., 2007). The project was inspired by traditional film editing studios where the task of cutting and splicing film into sequences for playback involved the use of physical tools such as cutting arms and taping stations. The TVE project was partly an attempt to capture beneficial aspects of both traditional and modern tools by blending the physical and the digital. However, unlike traditional tools, the TVE interface is aimed at facilitating *collaborative* editing of video by *amateur* users mainly for educational purposes.

The TVE project was innovative in its use of *active* physical tokens to represent individual video clips. These tokens, called *clip holders*, consist of hand-held computers with backlit, color displays embedded in custom-built cases. By chaining clip holders together with other tokens representing transition effects, users can easily create creative sequences of digital video. The TVE project involved numerous rounds of design and

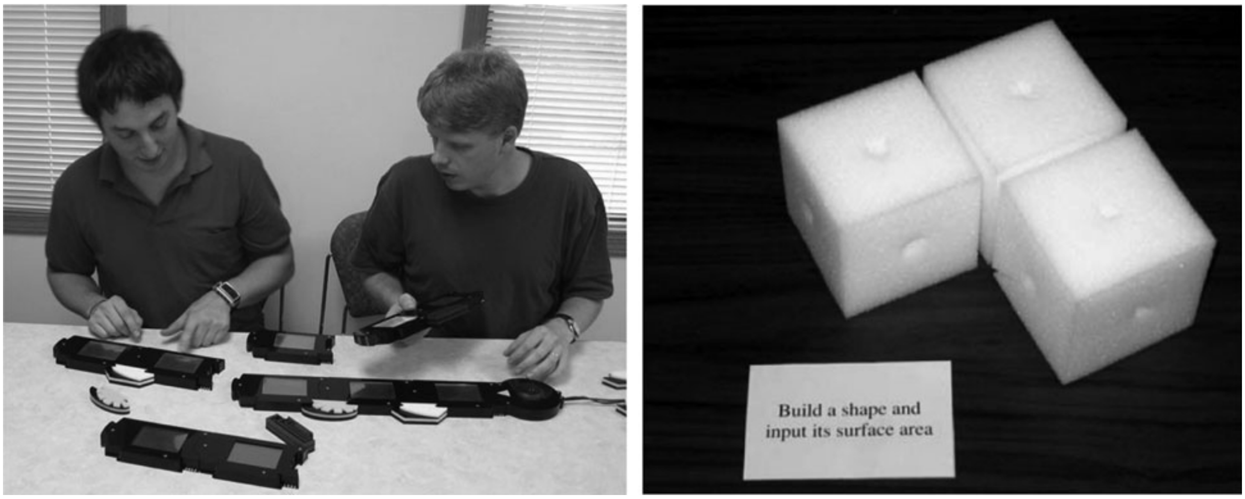


Fig. 1. (Left) The TVE system facilitates collaborative video editing, and (right) the Smart Blocks system allows users to explore concepts of volume and surface area.

prototyping, employing materials such as clay, foam core, and laser cut acrylic. Each successive low-fidelity prototype explored aspects of the physical design of tokens and ways in which they could be combined and constrained. In addition, even though the underlying technology of the project was replaced multiple times throughout the design process (an early prototype used computer vision to track tokens combined with a video projector to augment passive tokens), the core design concepts persisted.

3.2. Smart Blocks

Smart Blocks (Spring 2006; Girouard et al., 2007) is an augmented mathematical manipulative that allows users to collaboratively explore concepts of volume and surface area of three-dimensional objects (Fig. 1). Much like the digital manipulatives proposed by Resnick et al. (1998) the project seeks to combine the benefits of physicality with the advantages of digital information in creating an educational manipulative. The Smart Blocks system consists of 3-in. cubes, 3-in. dowel connectors, and a set of question cards. Children can answer questions prompted by the system or experiment with different constructions and see the resulting surface area and volume immediately. To implement the system, the team used a multitag radio frequency identifier (RFID) reader, embedding RFID tags in each block and each dowel.

3.3. Marble Track Audio Manipulator (MTAM)

The MTAM (Spring 2007) project augments a popular children's toy to allow collaborative musical composition through constructive play (Fig. 2; Bean et al., 2008). Children start by building towers and marble tracks. In the augmented system, marbles represent audio samples and colored tracks represent sound effects that can be applied to the samples. When a marble is dropped in the track at the top of the tower,

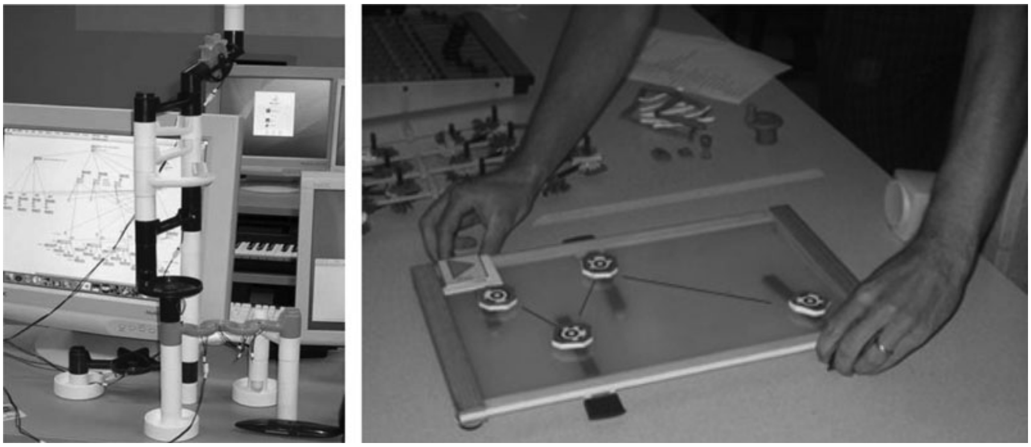


Fig. 2. (Left) The Marble Track Audio Manipulator and (right) WaveTouch system are playful educational tools for exploring the properties of audio and waveforms.

the sampled sound that marble represents starts to play. As the marble rolls through tracks, various sound effects and filters are applied to the original sample. For example, a wavy track adds a delay, a funnel adds reverb, and a water wheel adds distortion. The system was implemented with gold plated marbles that close electrical switches as they roll through the tracks. The switches, in turn, trigger keyboard input into a Max/MSP audio generation program.

3.4. WaveTouch

The WaveTouch (Summer 2007) project allows for collaborative and hands-on exploration of the properties of electromagnetic waves (Fig. 2). Users slide physical tokens on an interaction surface to adjust the frequency, amplitude, and form of a wave. The system responds in real time to user actions by projecting a waveform onto the interaction surface and by generating the corresponding audio signal. WaveTouch was developed for collaborative student exploration in classrooms and small learning situations to combine advantages of physical demonstrations and onscreen graphical simulations. It was implemented using the TopCodes computer vision library (http://hci.cs.tufts.edu/topcodes) and an overhead LCD top projector.

3.5. Traffic Flow Simulator

The Traffic Flow Simulator (Spring 2008) project was designed as a tool for children to learn about emergent properties of traffic flow through roadways. Children explore this phenomenon by connecting wooden sections representing various types of roadway and intersections. Traffic is introduced into the system using special source tokens, and flow is adjusted using a valvelike control. The system uses computer vision to track roadway sections on the interaction surface and an LCD projector to augment the roadway with images of cars, traffic lights, and accidents. Much like Zuckerman's FlowBlocks interface (Zuckerman et al., 2005), the Traffic Flow Simulator encourages collaborative and playful experimentation with dynamic systems to help children develop an intuitive grasp of these concepts.

4. TUI LABORATORY

4.1. Course structure

The TUI Laboratory is offered through the Computer Science Department to students from a variety of disciplines (Fig. 3). On average, the course enrolls 12 students, including advanced undergraduates and graduate students. Over the past 4 years, the course has drawn students from computer science, engineering, studio arts, art history, child development, human factors, education, and international relations. The course typically lasts one semester (14 weeks) meeting once per week for a 3-h session. We also successfully taught the course during a 7-week summer session meeting twice per week for a 3-h session (Summer 2007) and during a 14-week semester in which the course met twice weekly for 1.5 h (Spring 2008).

Similar to other HCI and computing courses in different institutions such as the University of Washington (Camarata et al., 2003), Stanford University (Klemmer et al., 2005), and Wellesley College (Turbak & Berg, 2002), the course borrows from the studio art tradition, where creativity and hands-on learning are strongly promoted. The course emphasizes collaborative design and peer teaching as means to transfer knowledge between students from diverse backgrounds. We divide the course into two phases. In the first phase, students typically spend 6 weeks acquiring background knowledge and conceptual foundations of tangible interaction. They also begin to work with the tools and technologies necessary for designing and building TUIs. In the second phase, students work in interdisciplinary teams to design and build functional TUI prototypes. Teams are typically assigned by the instructors and include students with complementary skills. Tables 1 and 2 summarize the topics and assignments we cover.

A practical challenge we face in offering the course is the lack of a dedicated laboratory space where students can design, build, and test their interfaces. Our solution involves dividing the course sessions between a conference room and our relatively small HCI research lab. The conference room

Fig. 3. Students in the TUI laboratory course.

Table 1. *TUI laboratory course topics covered in a 14-week course with 3-h sessions*

Topic	Sessions	Assignment Due
Introduction to TUIs	1	A1
TUIML	0.5	
Reality-based interaction framework	0.5	A2
Introduction to technologies and labs	2	
Theme overview & brainstorming methods	1	
Site visit (e.g., science museum visit)	1	A3
Technology demos	0.5	A4
Conceptual design lab	2	
Low-fidelity prototyping	2	A5
Proof-of-concept prototyping	1.5	A6
Functional prototyping	2	A7

allows for informal discussion and teamwork, while the minimal lab space enables students to store their artifacts and materials as well as to experiment with a variety of technologies. In addition, we encourage students to use their personal laptops for course projects.

4.2. Methods and techniques

In this section, we discuss methods and techniques that we apply throughout the course to facilitate hands-on learning, promote interdisciplinary collaboration, encourage innovation, and reinforce fundamental HCI concepts.

4.2.1. Conceptual foundations

As is common in evolving fields, researchers have not yet come to shared definition of tangible interaction. Thus, we begin each semester by discussing a variety of well-regarded TUIs in an attempt to bring the class to a consensus understanding of the nature of tangible interaction. Students are responsible for reading about and presenting an example interface to the class in one of the first sessions. Stressing traditional HCI concepts, this presentation includes a description of the design concept, use scenarios, interaction techniques, and implementation details. The interfaces we discuss include URP (Underkoffler & Ishii, 1999), Designers' Outpost (Klemmer et al., 2001), Topobo (Raffle et al., 2004), Tern (Horn & Jacob, 2007), SenseBoard (Jacob et al., 2002), Navigational Blocks (Camarata et al., 2002), and Illuminating Clay (Piper et al., 2002). We introduce an interdisciplinary view of tangible interaction that includes perspectives from the arts and design disciplines, but our course focuses on the data-centered view (Hornecker & Buur, 2006) of tangible interaction. This view is mainly concerned with physical representation and manipulation of digital data.

Table 2. *Student assignments for the TUI laboratory course*

Assignment	Description	
A1	Students present a description of a TUI of their choice	Individual
A2	Students hand in a TUIML specification for TUI presented in A1	Individual
A3	Students critique an existing computer-mediated system	Individual
A4	Students present a simple interface that combines the use of two types of tangible interface technologies	Team
A5	Students present the conceptual design of their TUI including low fidelity prototypes	Team
A6	Students present a proof-of-concept prototype of their TUI	Team
A7	Students present functional TUI prototypes and final papers	Team

After this preliminary period of discussion, we introduce students to the reality-based interaction framework (Jacob et al., 2008), which serves to place tangible interaction in the broader context of emerging interaction styles. This framework emphasizes existing real-world knowledge and skills that users leverage when they interact with a computer system. The framework identifies four themes of *reality* as being especially important: naïve physics, body awareness and skills, social awareness and skills, and environmental awareness and skills. By virtue of its embodied nature, tangible interaction can take advantage of user skills in all these areas. We refer to these themes in brainstorming sessions throughout the course in order to generate ideas for novel tangible interaction techniques.

4.2.2. Brainstorming techniques

Following the establishment of conceptual foundations, we introduce students to multiple brainstorming techniques including random inputs, idea maps, unusual combinations, and story telling, with the goal of fostering innovation. Early in the semester, an instructor moderates a brainstorming session that is followed by individual and student teams brainstorming activities. For each cohort, we select a theme to serve as a seed for generating project ideas. These themes focus on areas that are relatively new application domains for tangible interaction and that emphasize collaborative authoring and design work. For example, in the past we have chosen themes such as digital libraries, end-user programming systems, and informal science learning. Each topic is introduced to students prior to the brainstorming activities. For example, in 2008 students visited the Boston Museum of Science and met with domain experts such as exhibit developers and production staff to learn about informal science learning. However, although we encourage students to focus their design effort around the theme, final projects are not limited to these topics. Rather, we provide students with an opportunity for creative expression and for working on a project that they are passionate about.

4.2.3. Modeling and specification techniques

As groups are formed in the second phase of the course, we expect each student to contribute a unique set of skills to their team. Although this diversity is critical for innovation, it can also be a source of difficulty and frustration for newly formed groups. In the first two cohorts of the TUI laboratory, all teams experienced communication and workload division problems. As one student explained "the division of labor problem cropped up partially because we were unable to come up with a clear specification of our system at first." Another student wrote, "We didn't document well how different parts of the system will come together so this created a challenge when we put things together—communication is definitely the key." To address these challenges of communication, specification, and documentation, as well as to provide students from different disciplines with a common ground for exploring tangible interaction, we introduced TUI Modeling Language (TUIML; Shaer & Jacob, 2006), a visual modeling language for TUIs, starting the second cohort of our TUI Laboratory.

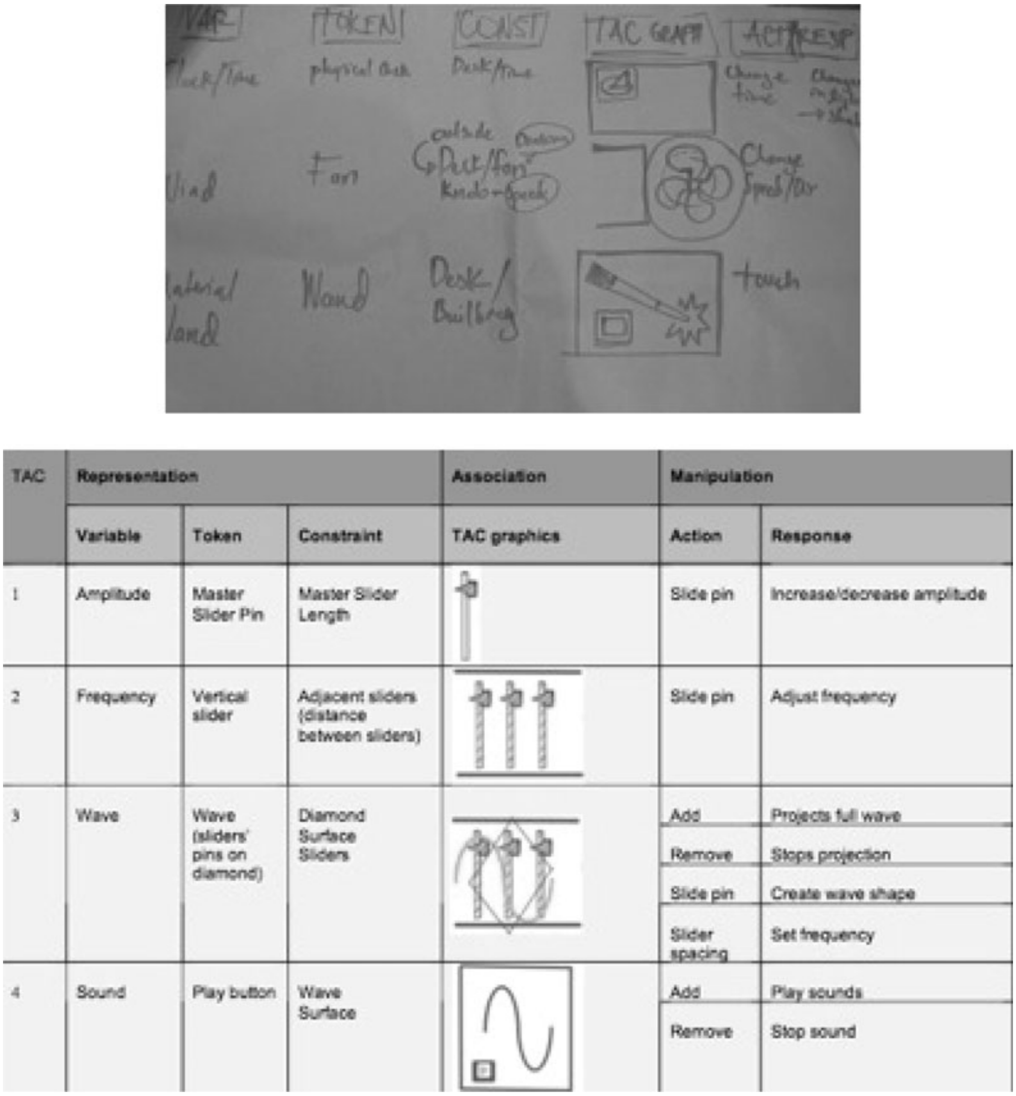
TUIML consists of an interaction model and diagramming techniques for describing the *structure* and *behavior* of TUIs in a high-level technology-independent manner. It is implemented as a visual language that combines iconic and diagrammatic approaches, drawing upon current practices of user interface and software design. To describe the *structure* and *functionality* of a TUI, TUIML uses a set of constructs (Shaer et al., 2004) that extends across a broad spectrum of TUIs. This set consists of *tokens, constraints*, and *tokens and constraints* (TACs) elements. A *token* is a graspable physical object that represents digital information or a computational function in an application. A *constraint* is a physical object that limits the behavior of a token with which it is associated. Finally, a *TAC* is a relationship between a token and one or more constraints. Such relationship expresses to users the kinds of interactions that an interface can and cannot support. TAC objects are similar to widgets in graphical user interfaces because they encapsulate both the set of meaningful manipulations users can perform upon a physical object (i.e., behavior) and the physical relations between tokens and constraints (i.e., state). TUIML provides an iconic representation for token and constraints and captures the structure of a TUI as a set of TAC relationships using a diagramming technique called the *TAC Palette*. To describe the underlying *behavior* of TUIs, TUIML offers a two-tier interaction model that consists of a *dialogue tier* and a *task tier*. The *dialogue tier* consists of a set of high-level states and transitions, whereas the *interaction tier* provides a detailed view of each user interaction that represents a thread of functionality. TUIML is most effective at specifying data-centered tangible interaction styles such as Interactive Surfaces, Constructive Assemblies and, Token + Constraint systems (Ullmer, 2002; Hornecker & Buur, 2006). TUIML can also be used to specify aspects of other design perspectives of tangible interaction such as the space-centered view and the expressive-movement centered view (Hornecker & Buur, 2006). As our course focuses students' design efforts around the data-centered view of tangible interaction, TUIML is capable of directly describing most student prototypes.

In the first part of the course we dedicate one lecture to TUIML. During this lecture students collaboratively construct TUIML diagrams to describe the seminal URP system (Underkoffler & Ishii, 1999). Figure 4 shows a TAC palette for URP that was constructed by students. Students identified three TAC relationships in URP and used iconic graphical representations to depict these relationships. For each TAC relationship the students defined computationally meaningful user interactions. Later, each student is required to use TUIML to describe an existing TUI system, study its properties, and present their analysis to the class.

In the second part of the course, students use TUIML to model their conceptual design, compare alternative designs, and communicate their designs to the rest of the class. In general, introducing the TUIML language helped students to communicate within and outside their teams as well as to thoroughly study their system designs prior to implementing them. As one student who worked on the Marble Track Audio Manipulator project described, "It helped us think about the many physical constraints that we could use to guide users and it helped us ensure that all possible actions on objects were accounted for within our implementation . . ." Another student who worked on the WaveTouch project explained, "TUIML gave us a clear abstraction of how we wanted the users to interact with the system. This allowed us to focus on designing those pieces well for our prototype, rather than spending a lot of time on parts that weren't as crucial to the actual user interaction." Figure 4 shows a TUIML TAC palette created by students for the WaveTouch system. The WaveTouch system uses token and constraints to guide users in exploring the concepts of frequency and amplitude. Tokens are not placed arbitrarily upon a surface. Rather they are placed on a slider. By sliding a token vertically, users change the amplitude of a waveform, and by sliding horizontally, users change the frequency of a waveform. These TAC relationships are captured in the TAC palette in Figure 4.

4.2.4. *Low-fidelity prototyping*

Prototypes are widely recognized as core means for exploring and expressing designs of interactive systems (Houde & Hill, 1997). We encourage students to create many design artifacts including sketches, low-fidelity prototypes, digital mock-ups, and interactive simulations. These design artifacts reinforce the concept of iterative design, while helping students to investigate open design questions regarding the function of a system, its look and feel, and alternative implementation techniques. The artifacts also help to facilitate the communication of ideas within the interdisciplinary teams. Throughout the course students create a series of artifacts, each representing a different aspect or view of the system. Students learn to create an effective storyboard to convey an interaction scenario, to use simple prototyping materials (such as prototyping clay, cardboard, and blue foam) to express ideas, and to build simple functional proof-of-concept prototypes. Figure 5 shows low-fidelity prototypes created to explore design considerations in two student projects. Both prototypes were built to investigate a design concept of surface-based interaction. By simulating the task and observing users interacting with these prototypes, the teams made significant modifications to their designs.



TAC	Representation			Association	Manipulation	
	Variable	Token	Constraint	TAC graphics	Action	Response
1	Amplitude	Master Slider Pin	Master Slider Length		Slide pin	Increase/decrease amplitude
2	Frequency	Vertical slider	Adjacent sliders (distance between sliders)		Slide pin	Adjust frequency
3	Wave	Wave (sliders' pins on diamond)	Diamond Surface Sliders		Add	Projects full wave
					Remove	Stops projection
					Slide pin	Create wave shape
					Slider spacing	Set frequency
4	Sound	Play button	Wave Surface		Add	Play sounds
					Remove	Stop sound

Fig. 4. A TUIML TAC palette of (top) URP and (bottom) the WaveTouch system.

4.2.5. Project technology

To translate low-fidelity prototypes into functional prototypes we introduce students to a selection of technologies that are portable, cross-platform, and open source. By necessity, we choose technology that fits within the budgetary and spatial constraints of our department, allowing students to use their own computers for development work.

Computer vision. Overall, computer vision has been the most popular technology choice for student projects. This is likely because many conceptual designs students create employ tabletop interaction techniques. One practical way to implement this style of interaction is to use computer vision to track the position of multiple objects on a two-dimensional surface while augmenting the surface with digital information using an LCD projector (e.g., Underkoffler & Ishii, 1999). For software, we have found that fiducial-tracking toolkits tend to offer the easiest solution. These toolkits track objects tagged with various types of barcode-like symbols (fiducials). There are several good options available including TopCodes (http://hci.cs.tufts.edu/topcodes) and ARToolkit (http://www.hitl.washington.edu/artoolkit). Both systems are open source, well documented, and have been used in many student projects. Among the challenges involving the use of computer vision is that using fiducial-tracking toolkits requires some programming experience. Thus, this technology is less accessible to students who come from disciplines other than Computer Science, and, within our project teams, the task of implementing the computer vision component of a project is almost always the sole responsibility of students with programming backgrounds. From an educational point of view this limits knowledge transfer within a team. Furthermore, students almost always find the physical installation and calibration of a camera and an LCD projector above or below an interaction surface challenging and time consuming. To address these issues, we recommend installing a dedicated computer vision interaction surface if space is available.

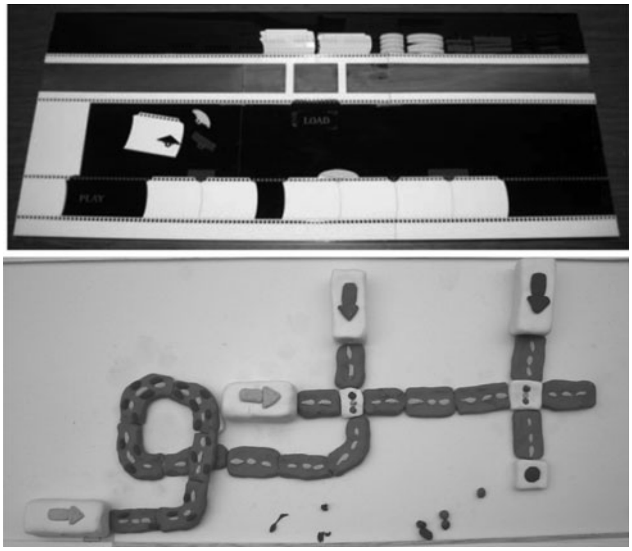


Fig. 5. Low-fidelity prototypes of (top) the TVE system and (bottom) the traffic flow simulator (from laser cut acrylic and modeling clay).

Microprocessors and project boards. Building a project using a microprocessor emphasizes the embedded aspect of tangible interaction. Students in our class have used two different types of embedded technology: the Handy Board and the LEGO® MINDSTORMS® RCX. Both are cross-platform, easy to use, and provide excellent solutions for educational projects with limited input/output requirements. Two relatively new technology platforms are also now available, which we will incorporate into future classes. The first is the LEGO MINDSTORMS NXT kit, which has more sophisticated capabilities than its predecessor, including servo motor drivers and a variety of sensors such as a sonar range finder and a sound sensor. The second is the Arduino Diecimila (http://www.arduino.cc), an inexpensive, open-source prototyping platform that can be programmed directly through a USB cable using a PC or a Macintosh.

RFID. Introducing RFID technology to students is a provoking way to start discussion on both the power and potential ethical considerations of emerging HCI techniques. Student projects have made clever use of two different RFID systems in the course. The first is the OBID® i-scan RFID reader from FEIG Electronics. This device is capable of identifying multiple RFID tags at once, and can write a small amount of data to individual tags. The second is the Phidgets RFID reader (www.phidgets.com). This device is read-only and can only recognize one tag at a time; however, it is inexpensive, cross-platform, and easy to use.

Input–output (I/O) boards. I/O boards can be connected to a desktop or laptop computer through a USB cable and allow both digital and analog I/O. The Phidgets product line includes a number of useful I/O boards that are inexpensive and easily programmed on multiple platforms using the Phidgets API (www.phidgets.com). Keyboard emulators enable switch input to be easily read as keyboard events within any closed-source program. For example, the I-PAC arcade controller from Ultimarc is an input-only USB keyboard emulator that requires almost no programming experience, no driver installation, and can be used with a variety of operating systems. A minor shortcoming of keyboard emulation is that it inherits keyboard functionality such as key repeat events injected at the operating system level that is inappropriate for non keyboard use.

Sensing technology. A wide variety of sensors are available to allow embedded systems to react to users and changes in an environment. Igoe and O'sullivan (2004) provide an excellent summary of devices that can be used with HandyBoards, Arduinos, or other microprocessors. There are also a variety of Phidgets products that provide the same capabilities for desktop or laptop computers.

4.2.6. Formal critique sessions

Art and architecture studio classes have a tradition of design critiques (crits) that serve as a form of evaluation of current ideas, exploration of future directions, and reflection. The conversation during a crit unveils the design rationale used during the design process. In our TUI Laboratory, we implement four formal crit sessions throughout the second part of the course as students develop their own interfaces. Instructors, students, guest domain experts, and expert interaction designers comprise crit panels. Our challenge as instructors is to create an atmosphere in which good ideas can surface while criticism can be openly discussed. Thus, we start each crit session by clearly defining goals and questions. Students then present their design and clarify their intentions, leading into an open discussion with the panel.

4.2.7. Reflection

Throughout the course we strongly emphasize reflection as means to promote long-term learning and sharing of knowledge. The structure of the course provides opportunities for students to face challenges, make mistakes, and reflect upon them. We implement several reflection techniques: blogging, retrospectives, and reflective questionnaires. All students are required to maintain a blog throughout the course, which serves as a design journal. Blogging enables students to reflect on their design process and decisions, communicate with the instructors and their peers, and share lessons learned with future course cohorts. Class retrospective sessions encourage students to share their experiences with other class members. In these sessions, held during the final stages of the project phase, each student shares her goals for the previous week, describes what went well, what went poorly, and why. These retrospective sessions allow students to learn from their peers and explore specific problems from multidisciplinary points of view. Finally, reflective questionnaires are given to students at the end of the course to allow for open reflection on their learning experience. Through their responses we learn about students' personal journeys throughout the course, about the challenges they faced, and about the insights they gathered.

5. COURSE RESULTS

In many ways, the TUI laboratory has exceeded our expectations. Students have created a wide variety of tangible interfaces that are outstanding for their creativity and innovation. Fourteen student projects were completed over the five semesters that the course was taught. Of these, 4 were extended into summer research projects involving a total of more than 10 students, and 3 resulted in publications in the Conferences on Tangible and Embedded Interaction (2007, 2008). In addition, 3 students continued on to pursue TUI research in doctoral studies both at Tufts and elsewhere, and 1 studio arts student developed an exhibit based on tangible interaction.

To determine the effectiveness of the course and of students' learning process, we rely on formal midterm and end of term course evaluations completed anonymously by students, and on informal evaluations such as reflective questionnaires, one on one conversations, and open discussions. In general, although some students had difficulty adjusting to the design studio approach, evaluations provided evidence that students found the course interesting, enjoyable, and motivating. Students immersed themselves in their projects with passion and enthusiasm. As one student described, "the project was something I was so excited about I wanted to do every part of it myself. I made a conscious effort to try not to step on anyone's toes. The only improvement I can think of would be meeting outside of class more . . ." Another student wrote, "this was my first "Studio" style class, and it was great. Lots of energy and enthusiasm. I would have liked to have had more classes between the start and finish." Student feedback also helped us to refine the course structure and the topics it covers, as described in the following section.

5.1. Course evolution

Naturally, the course has evolved throughout the five semesters we have taught it. In this section we provide a brief description of the materials introduced and the challenges facing students in each cohort, as well as the adaptations we made in the course to mitigate these challenges.

5.1.1. Spring 2005

In our pilot semester we placed a stronger emphasis on the technological aspects of building tangibles than on the design aspects. We dedicated 4 weeks to introducing a variety of implementation technologies and to the completion of a team project aimed to demonstrate the use of implementation technologies. The technologies introduced included microcontrollers, RFID, and computer vision. Each team was assigned two technologies and successfully completed the task of demonstrating the use of these technologies in the context of tangible interaction. In the end, however, all student teams chose to implement their final project using computer vision. This choice of technology was influenced by a majority of computer science students who felt more confident using software-based implementation techniques than working with hardware. Furthermore, the choice of technology influenced the design space of student projects; all projects implemented an interactive surface style interface. Finally, to our surprise, despite the new technologies and implementation techniques learned, by the end of course, most students reported that they found the conceptual design part of their project as the most difficult. As one student described: "The biggest difficulties were conceptual. After deciding on the high-level concept, we struggled to refine the interaction techniques . . ." In addition, almost all students reported team dynamic challenges, specifically communication and work division problems.

5.1.2. Spring 2006

To mitigate the conceptual design challenges, we redesigned the course to devote more time to conceptual design and low-fidelity prototyping. We also introduced TUIML and incorporated it in two course assignments: the specification of an existing TUI and the specification and documentation of the final team project. Furthermore, we reassigned student teams for the final project. This way, each student worked with two different teams: one for the technology project and one for the final project of designing and building a TUI. These measures were reflected in the results of the end of course questionnaire. Students still found the conceptual design part of their project the most challenging; however, fewer students reported problems within their team. In addition, the students' final projects were considerably more diverse, both in terms of tangible interaction design and the implementation technologies used.

5.1.3. Spring 2007

Following students' feedback we revised TUIML and introduced a new version of the language that allowed students to use iconic representations to describe the structure of TUIs (see Fig. 4). We also presented the reality-based interaction framework and used it to discuss existing TUIs and to brainstorm. Furthermore, we introduced new implementation technologies and emphasized the use of tangible output in addition to tangible input. The course structure of Spring 2007 semester is described in Table 1. It reflects a shift of focus in the course from tangible implementation technologies (Spring 2005) to the design and implementation of novel tangible interactions. Student projects in this semester exhibited not only diversity but also considerable creativity and innovation. Two of the three projects implemented tangible output techniques, and another project made use of continuous interactions and improvised interactions. Students' feedback highlighted challenges that student faced in attempting to produce custom-made artifacts for their projects. Of interest, students still pointed to the conceptual design as the most difficult phase of their design process. This led us to further investigate the challenges inherent to tangible interaction design.

5.1.4. Summer 2007

We introduced to the curriculum rapid prototyping methods including silicon molding and the use of a laser cutter. This resulted in additional prototyping rounds and the production of well-designed custom-made artifacts for student projects. Team dynamics in this cohort were exceptional and led to the development of innovative and creative projects.

5.1.5. Spring 2008

We established a partnership with the Boston Museum of Science. Through this collaboration students were exposed to real-world design problems introduced by the museum staff and through visits to the museum. Museum staff attended four course sessions, brainstormed with the students, and critiqued finished projects. Although student projects were not ready for exhibition in the Museum by the end of the semester, we hope that students will choose to extend their projects with the possibility of exhibiting and evaluating them in the museum. Nevertheless, this collaboration motivated the students and resulted in high-quality projects that took into account the requirements unique to informal science learning in a museum environment.

6. CONCLUSION AND FUTURE WORK

Tangible interaction has shown promise to improve computer-mediated support for domains such as learning, problem solving, and design. However, designing tangible interaction is a complex process that encompasses multidisciplinary knowledge including engineering, art, and the social sciences. In this paper we shared our experiences teaching tangible interaction design in an interdisciplinary, project-based laboratory course over a period of 4 years at Tufts University. We designed our course to facilitate hands-on learning, reinforce the *big ideas* of HCI, foster innovation, and promote interdisciplinary collaboration. In doing so, we considered the particular constraints of our institution such as the lack of dedicated laboratory space and limited budget. We also confronted the need to cultivate a studio culture within a Computer Science department. We shared lessons learned over 4 years and provided recommendations for implementing tangible interaction courses in institutions with similar constraints.

We have two primary future goals for the course. First, we plan to teach this course in the Computer Science department at Wellesley College. This will require us to adapt the course for an undergraduate women's liberal arts college environment. We believe that the interdisciplinary nature of the course will attract students from a variety of disciplines, and thus will help introduce young women to computing through a nontraditional path. It will be interesting to see in what ways teaching this course in a single-gender college will differ from its current version in a coeducational environment. Second, we plan to strengthen our collaboration with the Boston Museum of Science, while emphasizing the topic of tangible interaction support for informal science learning. As tangible interaction offers potential benefits for learning such as support for colocated collaboration and tangible thinking, we believe that there is a strong basis for enhancing the collaboration between the course and the Boston Museum of Science. Students may assist exhibition designers develop new ideas and solutions, whereas the museum forum can provide students with inspiration for their projects as well as with potential showcase and opportunities to evaluate their projects with users in real-world environments.

ACKNOWLEDGMENTS

We thank Ken Camarata, Ellen Yi-Luen Do, and Mark Gross for inspiring us through their Physical Computing Studio and assisting us with the design of our first TUI laboratory course; the Department of Computer Science at Tufts University for supporting this nontraditional course; and Hiroshi Ishii and the Tangible Media Group at the MIT Media Lab for close collaboration and exchange of ideas and speakers. We are grateful to our guests Angela Chang and Jamie Zigelbaum from the MIT Media Lab; Daniel Dennett from the Tufts' Philosophy Department; Paul Lehrman from the Tufts' Music Department; and Erin Solovey, Audrey Girouard, and Leanne Hirshfield from the Tufts HCI research lab for their feedback and advice to the students. We also thank the Boston Museum of Science for fruitful collaboration, specifically Taleen Agulian, Lucy Kirshner, Dan Noren, Bill Rogers, Deb Souvinee, and Peter Wong for volunteering their time to meet with students and explaining the process of designing interactions to support informal science learning in museum settings. Finally, we thank the students of the TUI laboratory for sharing with us their enthusiasm, creativity, and insights. Support for our research in the area of tangible interaction was provided by NSF Grant EHR-0227879 on wireless networks and NSF Grant IIS-0414389 on reality-based interaction.

REFERENCES

Bean, A., Siddiqi, S., Chowdhury, A., Whited, B., Shaer, O., & Jacob, R.J.K. (2008). Marble track audio manipulator: a tangible user interface for music composition. *Proc. TEI '08 Int. Conf. Tangible and Embedded Interaction*.

Camarata, K., Do, E.Y.-L., Gross, M.D., & Johnson, B.R. (2002). Navigational blocks: tangible navigation of digital information. *Proc. CHI '02 Extended Abstracts on Human Factors in Computing Systems*. New York: ACM Press.

Camarata, K., Gross, M.D., & Do, E.Y-L. (2003). A physical computing studio: exploring computational artifacts and environments. *International Journal of Architectural Computing 1(2)*, 169–190.

Girouard, A., Solovey, E.T., Hirshfield, L., Ecott, S., Shaer, O., & Jacob, R.J.K. (2007). Smart Blocks: a tangible mathematical manipulative. *Proc. TEI '07 Int. Conf. Tangible and Embedded Interaction*.

Horn, M.S., & Jacob, R.J.K. (2007). Designing tangible programming languages for classroom use. *Proc. TEI '07 Int. Conf. Tangible and Embedded Interaction*.

Hornecker, E., & Buur, J. (2006). Getting a grip on tangible interaction: a framework on physical space and social interaction. *Proc. CHI '06 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Houde, S., & Hill, C. (1997). What do prototypes prototype? In *Handbook of Human–Computer Interaction* (Helander, M., Landauer, T., & Prabhu, P., Eds.), 2nd ed. New York: Elsevier Science.

Igoe, T., & O'Sullivan, D. (2004). *Physical Computing: Sensing and Controlling the Physical World With Computers*. Boston: Thompson Course Technology PTR.

Jacob, R.J.K., Girouard, A., Hirshfield, L.M., Horn, M.S., Shaer, O., Solovey, E.T., & Zigelbaum, J. (2008). Reality-based interaction: a framework for post-WIMP interfaces. *Proc. CHI '08 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Jacob, R.J.K., Ishii, H., Pangaro, G., & Patten, J. (2002). A tangible interface for organizing information using a grid. *Proc. CHI '02 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Klemmer, S.R., Ju, W., & Verplank, B. (2005). Teaching embodied interaction design practice. *Proc. ACM Conf. Designing for User Experience, DUX '05*.

Klemmer, S.R., Newman, M.W., Farrell, R., Bilezikjian, M., & Landay, J.A. (2001). The designers' outpost: a tangible interface for collaborative web site design. *Proc. UIST '01 Symp. User Interface Software and Technology*. New York: ACM Press.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Piper, B., Ratti, C., & Ishii, H. (2002). Illuminating clay: a 3-D tangible interface for landscape analysis. *Proc. CHI '02 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Patten, J., Ishii, H., Hines, J., & Pangaro, G. (2001). SenseTable: a wireless object tracking platform for tangible user interfaces. *Proc. CHI '01 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Raffle, H., Parkes, A., & Ishii, H. (2004). Topobo: a constructive assembly system with kinetic memory. *Proc. CHI '04 Conf. Human Factors in Computing Systems*. New York: ACM Press.

Resnick, M., Martin, F., Berg, R., Borovoy, R., Colella, V., Kramer, K., & Silverman, B. (1998). Digital manipulatives: new toys to think with. *Proc. CHI '98 Human Factors in Computing Systems*. New York: ACM Press.

Resnick, M., & Ocko, S. (1991). LEGO/Logo: learning through and about design. In *Constructionism* (Harel, I., & Papert, S., Eds.). Norwood, NJ: Ablex.

Shaer, O., & Jacob, R.J.K. (2006). A visual language for programming reality-based interaction. *Proc. IEEE Symp. Visual Languages and Human-Centric Computing*.

Shaer, O., Leland, N., Calvillo, E.H., & Jacob, R.J.K. (2004). The TAC paradigm: specifying tangible user interfaces. *Personal and Ubiquitous Computing 8(5)*, 359–369.

Turbak, F., & Berg, R. (2002). Robotic design studio: exploring the big ideas of engineering in a liberal arts environment. *Journal of Science Education and Technology 11(3)*, 237–253.

Ullmer, B.A. (2002). *Tangible interfaces for manipulating aggregates of digital information*. PhD Thesis. Massachusetts Institute of Technology.

Underkoffler, J., & Ishii, H. (1999). URP: a luminous-tangible workbench for urban planning and design. *Proc. CHI '99 Human Factors in Computing Systems*. New York: ACM Press.

Yi-Luen Do, E., & Gross, M.D. (2007). Environments for creativity: a lab for making things. *Proc. ACM Creativity and Cognition*.

Zigelbaum, J., Horn, M., Shaer, O., & Jacob, R.J.K. (2007). The Tangible Video Editor: collaborative video editing with active tokens. *Proc. TEI '07 Int. Conf. Tangible and Embedded Interaction*.

Zuckerman, O., Arida, S., & Resnick, M. (2005). Extending tangible interfaces for education: digital montessori-inspired manipulatives. *Proc. CHI '05 Human Factors in Computing Systems*. New York: ACM Press.

Orit Shaer is an Assistant Professor of computer science at Wellesley College. Orit received her PhD and MS in computer science from Tufts University. She was a Visiting Researcher in the Design Machine Group at the University of Washington and in the University College London Interaction Center. Dr. Shaer previously served as a System Engineer at Orbotech Ltd., where she led an interdisciplinary development team. Her research in HCI focuses on developing interaction techniques and software tools for next generation user interfaces.

Michael Horn is a PhD candidate in computer science at Tufts University. He also serves as a Faculty Associate in the Problem Solving With Discrete Mathematics Workshops for grade 5–9 teachers. Michael received an undergraduate degree in computer science from Brown University. As a graduate student he spent 2 years as an NSF GK-12 Fellow working in a local K–8 school. His research interests include exploring innovative ways to integrate appropriate and useful technology into K–12 classrooms. In particular, he is interested in educational programming languages and the opportunities created by TUI technology.

Robert Jacob is a Professor of computer science at Tufts University. He was also a Visiting Professor at the MIT Media Laboratory in the Tangible Media Group. Before coming to Tufts, he was in the Human–Computer Interaction Lab at the Naval Research Laboratory. He received his PhD from Johns Hopkins University. He was Papers Co-Chair of CHI 2001, Co-chair of UIST 2007, and Vice-President of SIGCHI. He was elected to the ACM CHI Academy in 2007. Dr. Jacob's research interests are new interaction media and techniques and user interface software.