NOVEL INTERFACES FOR INTERACTIVE ENVIRONMENTS

Robert W. Lindeman, Worcester Polytechnic Institute

INTRODUCTORY ESSAY

Novel Interfaces for Interactive Environments is a second- or third-year undergraduate course that arms students with the knowledge and experience to think beyond traditional interaction paradigms when designing gaming experiences. In recent years, important technical advancements have converged, allowing gamers to interact with game worlds in new and exciting ways. Computer systems, including mobile platforms, are small, cheap, and powerful. Reliable and pervasive networking infrastructures at many levels (e.g., Ethernet, cellular, WiFi, Bluetooth) mean anyone can play with anyone else, at any time, anywhere. New interface hardware, starting with the Nintendo Wii and continuing with Microsoft Kinect, PlayStation Move, Oculus Rift, and multi-touch surfaces has been strongly embraced by large segments of the population, some of whom have not traditionally been served by mainstream games. These same developments, however, challenge game designers to utilize these technologies effectively by reconsidering their reliance on tried-and-true game interface devices, such as the keyboard and mouse or game controllers.

In 2009, WPI implemented an experimental undergraduate course, called IMGD 3100: Novel Interfaces for Interactive Environments, to help students explore the use of these emerging technologies as part of their four-year B.S. degree program in Interactive Media & Game Development. The course has three main thrust areas: 1) study of the strengths and limitations of the different human sensory systems (e.g., vision, hearing, and touch); 2) hands-on work building prototype interface devices using off-the-shelf microprocessors (e.g., Arduino), simple sensors and actuators, along with programming interfaces for accessing them from a game; and 3) exploration of how interface device characteristics influence game design. Since 2009, the course has been offered twice more, and the structure has been tweaked to better emphasize the theory and practice of designing for novel interaction.

COURSE STRUCTURE

The overall theme is to experience the design trade-offs that must be made to deliver effective user experiences using novel interfaces. Each class meeting begins with a live demonstration of a game that uses an "alternative" interface, such as a brain-machine device, light gun, or tactile vest, followed by a class discussion. Since the students get to actually try each of the demos, this required us to obtain several rather-obscure game interfaces, though many of them we already had acquired over time.

Following the demo, a short lecture is given covering the technical and human aspects of the demoed interface in detail, such as which senses it stimulates and how (e.g., touch using vibro-tactile motors). Additional, related interface topics are also covered, such as exoskeleton devices for touch. This requires the lecturer to have a solid foundation in both the human and technical sides of human-computer interaction, which fits well into common preparation of virtual reality researchers.

Each session concludes with an in-class, hands-on exercise using devices such as the Arduino, an Android phone, or the like. In one session mid-way through the term, for example, each student was given an old (mechanical) computer mouse to dissect. The idea was to demystify these revolutionary devices, driving home the idea that this stuff is really not that hard.

In addition to the significant work done in class, there were a series of out-of-class assignments. One of these was individual, and the other three were done in teams. Examples of the assignments can be found in the syllabus section below. The final group project had small (2-3 people) teams of students propose a design for a game experience where the primary inputs and outputs were not the keyboard, mouse, or monitor. Please see the syllabus below for more information on the final project.

COURSE EVOLUTION

The course has evolved over time, most recently (September 2013) experimenting with flipping some of the lectures to allow more in-class exploration and discussion. Evaluation of the course through student questionnaires, supplemented by face-to-face interviews with students conducted by a third party, has shown the strength of the balanced approach of theory (e.g., study of the human senses) and practice (e.g., building with Arduino). Because much of the course uses off-the-shelf technologies that are inexpensive yet expressive, the class model lends itself well to recreation by other game-design programs.

Some interesting observations were made during the delivery of this course. First, it was interesting to note how hesitant students were when it came to exploring hardware. Most (about 85%) of the students in the course had a technical focus but felt much more comfortable incorporating creative uses of software into their designs as opposed to hardware innovation. Similarly, students typically employed visual and, to a lesser extent, auditory elements to support the story they were telling with their games; there was almost no consideration for "displaying" elements to the other senses, such as touch or smell.

More recently, Maker Culture has started to impress itself on popular culture, including on college campuses. This has been supported by the availability of low-cost electronics (e.g., the

Arduino family) and robotics kits like the Vex products, as well as the emergence of low-cost 3D printers, laser cutters, and the like. Maker Spaces, as the labs where these technologies are commonly housed are called, also bring together people with varying interests, including artists, engineers, and computer scientists. The Novel Interfaces course described here fits very well into this recent growth in DIY culture.

The course material relied heavily on the use of an Arduino Starter Kit, along with a book that covered an introduction to electronics and circuit design. The kits cost about US $50, and each student was required to purchase his/her own kit. The books for the course (see syllabus below) were made freely available on the University network, so the cost was easy to justify. In addition, we also had several Android smartphones available for use by the students.

We experimented during the last offering of the course with flipping some of the lectures so that the material was recorded ahead of time for students to view prior to coming to class. Each video was accompanied by a set of questions based on the video material that the students had to answer and email to the lecturer. The questions were more than simple verbatim retrieval of the material but rather focused on what the student thought about the topic; we were more interested in their opinions and assessment of the material than in checking that they had actually watched the video, though this is certainly a worry in flipped classes. Before class, we would read the answers and then in class we could focus on the material that they had the most trouble with. At the end of the course, we administered a short survey about the flipped portions of the lectures, and what the students thought about them. In general, the video lectures were well received, even though the style was just a talking head as an inset to the PowerPoint slides. The students liked the low-production quality of the presentation and said they thought flipping worked very well for material that was not so complicated; however, they preferred the in-class lectures for materials that required more-careful explanation or that had a higher chance of raising questions.

SYLLABUS

READINGS

There are three main books for this course, and other readings will be provided over the course of the term.

- **PI**: *Programming Interactivity*, 2nd Edition, Joshua Noble, O'Reilly Media, Inc., Print ISBN-13: 978-0-596-15414-1
- **GSA**: *Getting Started with Arduino*, 1st Edition, Massimo Banzi, O'Reilly Media, Inc., Print ISBN-13: 978-0-596-15551-3
- **ACB**: *Arduino Cookbook*, 2nd Edition, Michael Margolis, O'Reilly Media, Inc., Print ISBN-13: 978-1-4493-1387-6

COURSE OBJECTIVES

This course is designed to make students think beyond "traditional" computer and video game interfaces, such as the keyboard, mouse, and game pad.

By the end of the course, you should feel comfortable assembling low-level components (and *dis*assembling larger ones) to create novel interface devices. You should also know how to make them communicate with application software, and how the design of interactive applications, such as games, is influenced by the selected devices.

Beyond games, the knowledge gained from this course will allow designers of teleoperated robotics systems as well as traditional user interface designers to better understand the possibilities of incorporating simple sensor and actuator systems into their designs.

This is a perfect time to study these topics because of several current developments:

- sensors and actuators are becoming cheap and plentiful;
- computer systems, including mobile platforms, are becoming smaller, cheaper, and more powerful;
- networking infrastructures at many levels (e.g., Ethernet, cellular, WiFi, Bluetooth) are now widely available;
- the Nintendo Wii, Microsoft Kinect, PlayStation Move, Leap Motion, SmartPhones, eReaders, all "game changing" technologies, have been embraced by large segments of the population who have not traditionally been served by their respective communities;
- the use, availability, and acceptance of robotic systems is steadily increasing; and
- people love gadgets more than they ever have in the past!

There are several objectives for this course:

- Understand the strengths and limitations of the different human sensory systems (e.g., sight & sound);
- Understand how device characteristics influence system design;
- Learn how to program an embedded processor to gather input from, and deliver output to a user;
- Learn just enough about electrical engineering to construct simple circuits for input and output;
- Explore emerging mobile platforms (e.g., Android);
- Do some cool stuff!

GRADING

50% Regular Projects
50% Final Project

PRESUPPOSED BACKGROUND

In this course, students will be expected to understand systems-level computer science concepts, such as systems programming. In addition, students will evaluate various interface options for video games, which will draw upon material from a course on The Game Development Process.

PROJECTS

The projects for this course consist of several programming assignments, designed to supplement the material covered in the lecture with practical experience. The projects for this course are demanding and will require a lot of time. On the flip side, most people enjoy these types of projects, so it should be okay.

Most of the projects will use the Arduino platform, which allows you to quickly (and relatively painlessly) set up a framework for building physical Input/Output (I/O) devices. The Arduino has a very active community of, er, hardware hackers who not only build interesting I/O devices with the Arduino, but also provide good support for others wanting to do the same. If you want to get the most out of this course, you will embrace and contribute to this community.

DISCUSSION BOARDS

You are encouraged to post your questions on the class discussion board, and to look for answers there. We will be using this heavily during the course for clarifications, corrections, etc. In addition, the Arduino and Android communities have forums with relatively quick turnaround. These are indispensable, so please take advantage of them as well.

LATE POLICY

Projects are due at the specified date and time. Late projects will be penalized 10% for each 24-hour period after the due date/time. Whether a project is 3 hours or 20 hours late, it will be graded down by 10%. You will be given adequate time to complete each project if you start when it is assigned. Projects will be turned in electronically, and the date/time received will be used to determine any late penalty. PLEASE do not miss class in order to finish up a project.

ATTENDANCE

Attendance is required. If you have an unavoidable need to be absent from the lecture, you do not need special permission, but you are responsible for the work covered even if you are not in class. Also, please bring your Arduino kits with you to all class meetings.

GENERAL

Questions and discussion are highly encouraged throughout the lecture hours. The best way to reach the instructor is by using e-mail.

The Computer Lab can be used for this course, and the Arduino boards you will use can be connected to the lab computers for you to do your projects there. Android development can also be done on the lab machines. You are also free to work on your own computer.

CLASS CONDUCT

This course is intended for serious students. Participants will be expected to adhere to all rules of professional behavior.

Individual projects are expected to be done individually. As such, students are encouraged to discuss their work with each other but are also expected to do the work by themselves.

Any breach of professional ethics as evidenced, for example, by copying exams or projects, downloading code from the Internet, cooperating in more than discussions and study groups, misusing computer resources, or using outside help of any kind will be considered adequate reason for failure in the course.

Group projects are designed so that every person gains a significant amount of new material. In the workplace, each team member is expected to contribute. Participants in group projects in this course should keep this in mind and act accordingly. In evaluating each group, all team members will be asked to distribute a fixed set of "points" to the rest of their team based on how much each member contributed.

It is to be emphasized that knowledge of material and professional behavior are tied together; failure in one of them negates any excellence in the other. Students who stay in the course past the first three days agree to adhere to the strictest rules of professional behavior.

CLASS SCHEDULE

Class	Lecture Topic/Slides	Readings	In-Class Demos/Exercises	Projects
1	Intro		In-Class Exercise 1 In-Class Game Demo: 3rd Space Vest	ASSIGNED: Project 1: Closed-Loop
2	Very Quick Intro to Electrical Circuits		In-Class Game Demo: U-Dance	DUE: Project 1: Closed-Loop ASSIGNED:

				Project 2: Sensor to Application
3	Designing Interactive Systems (Slides) Designing Interactive Systems (Video Lectures)	**PI**: Ch. 1	In-Class Game Demo: NeuroSky Brain Interface	
4	Human Vision		In-Class Game Demo: Novint Falcon Haptic Device	DUE: Project 2: Sensor to Application
5	Visual Interface Elements		In-Class: Interface Hall of Shame In-Class Game Demo: Google Glass	
6	Human Audition		In-Class Game Demo: Oshare Majo Love & Berry	ASSIGNED: Project 3: Android Game
7	Human Audition (cont.)		In-Class Game Demo: Oculus Rift and Razer Hydra	
8	Human Haptic System (Slides) Human Haptic System (Video Lectures)		In-Class Game Demo: Synaptics ForcePad	DUE: Final Project Ideas Due
9	Designing Physical Input	**PI**: Ch. 7	In-Class Game Demo: EyeToy: Anitograv (Harmonix)	DUE: Project 3: Android Game
10	Designing Physical Feedback: Motors; Final Project Status Reports	**PI**: Ch. 11	In-Class Game Demo: Augmented Reality Toolkit (ARToolKit)	STATUS: Final Project Status Demo 1
11	Final Project Status Reports			STATUS: Final Project Status Demo 2
12	Mobile Devices			STATUS: Final Project Status Demo 3
13	Input Device Design: Game Controllers			STATUS: Final Project Status

				Demo 4
14	Final Project Presentations			DUE: Final Project

REFERENCES AND URLS

Course Web Page: http://web.cs.wpi.edu/~gogo/courses/imgd3100_2013a/
Android Development: http://developer.android.com/
Arduino: http://www.arduino.cc/
Arduino Starter Kit Example 1: http://store.arduino.cc/product/K000007
Arduino Starter Kit Example 2: https://www.sparkfun.com/products/12789
Arduino Starter Kit Example 3: https://www.sparkfun.com/products/12001
MakerBot 3D Printer: http://www.makerbot.com/
Maker Spaces: http://makerspace.com/
Vex Robotics: http://www.vexrobotics.com/