



# WPI

---

# IMGD 3100 – Novel Interfaces for Interactive Environments: Introduction

Robert W. Lindeman

Associate Professor

Interactive Media & Game Development

Human Interaction in Virtual Environments (HIVE) Lab

Department of Computer Science

Worcester Polytechnic Institute

[gogo@wpi.edu](mailto:gogo@wpi.edu)

---

## Motivation

---

- Some interesting recent developments
  - Mobile computer systems are cheap, powerful, and everywhere
  - Wireless connections are everywhere
    - Cellular, WiFi, Bluetooth, ...
  - Sensors and actuators are cheap
  - Accessible robot systems are emerging
    - Roomba, MANY kits
  - Wild popularity of new games and platforms
    - Rockband, Kinect, Wii/U, iPhone/iPod/iPad, Android

## Motivation (cont.)

---

- I've been working on 3D User Interfaces for Virtual Reality for a loooooong time
  - About 20 years
- VR and gaming are very related
  - But games sell!
- Games have gotten a little stale
  - How many more FPSs can you make?
- Graphics are pretty good now!
- Sound is also pretty good!
- So, what's the next big thing?

## Questions

---

- ❑ Why are car navigation systems so popular?
- ❑ Why are smartphones and tablets so popular?
- ❑ What made the Wii different?
  - What was different about popular Wii games?
- ❑ What did Sony and Microsoft do to catch up?
- ❑ What can Nintendo do to stay ahead?
- ❑ Where does this innovation come from?!

## HCI in Real Environments

---

- Mobile devices
  - Car Navigation (useful during task)
  - Medical monitoring (people are aging)
  - Foursquare (we love collecting!)
  - Ingress (augmenting reality as a game)
  
- Multi-person coordination
  - Military operations
  - Search-and-Rescue
  - Fire fighting
  
- Stuff we don't know about yet!

## HCI in Virtual Environments

---

- Layouts for user interfaces
  - Heads-Up Displays (HUDs)
  - Chat windows
  - Game state
- Spatialized audio/voice
- Haptic (touch) displays
  - Hit by weapons fire
  - Virtual surgery training
  - Steering wheels
- Smell?
- Taste?
- What about input?

# HCI in Teleoperated Environments

---

- Teleoperated robot systems are used more and more
  - Disaster areas
  - Medical micro-robots
  - Space exploration
  - UXVs (UAV, UGV, UUV)
- Operator relies on remote sensors
  - Limited fidelity
  - Communication delays
- Remote actuators change the physical world

## Common Problems

---

- All three environments require the user to:
  1. Sense something (i.e., get input)
    - Perceive the environment
    - Limited fidelity (screen space, etc.)
  2. Make a decision
    - Draw on new and existing knowledge
    - Limited knowledge
  3. Carryout actions (i.e., produce output)
    - Make something happen
    - Limited expressiveness (mouse, gamepad, etc.)
- Errors can be made at each step
- In this course, we will focus on **1** & **3**



## What to Expect

---

- This course is about
  - How to build new interfaces for these environments
  - How to design applications (e.g., games) that take advantage of these devices
- This is really a **Chicken & Egg** proposition
  - Devices constrain the application
  - Application constrains the devices
  - User constrains both
  - Environment constrains both
- But, constraints are a good thing!!

## What to Expect (cont.)

---

- The groundwork to do this stuff right requires
  - A good understanding of the human sensory systems
  - A good understanding of building devices
  - A good understanding of application domains
- The projects you do in this course will help you learn all of this

## Summary of Syllabus

---

- Lectures and in-class work
  - Exercises designed to drive home concepts, or to get you thinking about projects
- 1 Application Design Report (33%)
  - Research a potential application (Real/Virtual/Tele)
  - Design (not build) a novel user interface for it
- ~4 "Smaller" Projects (33%)
  - Individual projects
  - Use the Arduino and Android to build stuff
- 1 Final Project (34%)
  - Team-based
  - Use Arduino/Android, plus other software you choose (Unity, Flash, C4, XNA, etc.)
- All material on class website ([www.cs.wpi.edu/~gogo/courses/imgd3100/](http://www.cs.wpi.edu/~gogo/courses/imgd3100/))

## Readings for the Course

---

- There will be material from several eBooks:
  - ***Programming Interactivity***, 2nd Edition, Joshua Noble, O'Reilly.
  - ***Getting Started with Arduino***, 1st Edition, Massimo Banzi, O'Reilly.
  - ***Arduino Cookbook***, 2nd Edition, Michael Margolis, O'Reilly.
  - eBooks can be accessed from any WPI computer
    - Check the course Web page for details.
- And also material from the Web:
  - Arduino/Android communities, Electrical engineering help, Project idea
- Excerpts from:
  - 3D User Interfaces: Theory and Practice, Bowman, Kruijff, LaViola, Poupyrev, 2005, Addison Wesley, ISBN: 0-201-75867-9

## More About the Projects

---

- One of the goals of this course is for you to feel confident to build devices
  - Find sensors to measure what you want to measure
  - Find interesting ways of getting input to the system
  - And output to the real world (e.g., the user)
    - Pinwheels for network traffic
  
- You can find stuff easily these days
  - Sparkfun ([www.sparkfun.com](http://www.sparkfun.com))
  - Digikey ([www.digikey.com](http://www.digikey.com))
  - Jameco ([www.jameco.com](http://www.jameco.com))
  - Allelectronics ([www.allelectronics.com](http://www.allelectronics.com))
  - RadioShack ([www.radioshack.com](http://www.radioshack.com))

## Engineering vs. Science

---

- Scientific Method
  - Define a hypothesis, test it, and make laws
- Engineering Approach
  - Come up with an idea, build it, refine
- Both of these require solid foundations!
  - You need to do your homework
- Software people are reluctant to mess with hardware
  - Might break something
  - Might burn your fingers

## Arduino Development

---

- You should all have Arduino Development Kits for this course from SparkFun.
- The kits work in the lab
  - And you can work at home too
- Cross-platform
  - Mac, Linux, Windows
- LOTS of help on the Web:
  - <http://www.arduino.cc/>
- We'll be using this Arduino Kit
  - <https://www.sparkfun.com/products/11930>

## Android Development

---

- Google gave us phones!
- The Android development environment is eclipse based
  - Java is main language
  - Emulator for testing
- Eclipse has been configured in the lab
  - And you can work at home too
- Cross-platform
  - Mac, Linux, Windows
- LOTS of help on the Web:
  - <http://developer.android.com/>



## Final Project

---

- ❑ Two- or three-person teams
- ❑ Choose an application area
- ❑ Define a set of interface devices and techniques that support the app
- ❑ Interim status demos in class
- ❑ Presentations will be done the last week of this course, where you will show your stuff
- ❑ Let's see some from previous years!

## Flipping the Class

---

- ❑ Some of the lectures for this course will be delivered by video+testing
- ❑ Classes will be used to dig much deeper
- ❑ It will be assumed you have watched the assigned video(s) prior to class
- ❑ This is called “Flipping the Classroom”
- ❑ More on this real soon (for Tuesday!)

## Course Support

---

- There is a GDC Forum for this course
  - <http://forums.gdc.wpi.edu/>
  - All project discussions should be posted there
  - You are encouraged to post screen-shots of your progress
  
- Contact me if you need to meet for office hours
  - I'll post some real soon...

## Expected Outcomes

---

- Think beyond the gamepad
  - Alternative I/O to support a particular application
- Feel comfortable building new things
- Know how best to provide output to humans
  - All the senses
- Build up your portfolio

## Final Thoughts

---

- I don't know which parts of this course will give you problems
  - We need to work together to tweak the content, presentation, etc.
- I welcome any and all feedback and suggestions on how to make the course better
- We have some flexibility to re-order/change topics
- Be playful!
- Be ambitious!