CS415
Human Computer Interaction

Lecture 1 - Introduction
Preparation for CS 415 - SE’s

- C (CS125) and C++ (CS225)
- C for CLI, with starter code for fork (CS420)
- QtCreator (C/C++) - WIMP GUI
- OpenCV - C/C++ - 3D and 4D
- Other VHLL’s for HCI prototyping as you wish - Project

© Sam Siewert
Dr. Sam Siewert

Johnson Space Center – Shuttle Ascent and Entry GN&C, Mission Control Center

JPL AI Group, CU Space Grant and Ball Aerospace

CU Boulder Adjunct Professor, CTO, Architect, Developer/Engineer in Local Start-ups and Intel Architecture / Labs

U. of Alaska, Assistant Professor, Computer Systems Engineering, Alaska Space Grant, DHS ADAC

Embry Riddle Prescott, Assistant Professor, Computer, Electrical and Software Engineering, ARI Drone Net

© Sam Siewert
Industry & Teaching Background

- **General Experience (~25 Years in Systems)**
  - 12 Years NASA JSC, JPL, CU, Ball Aerospace
  - 12+ Years Commercial Scalable and Embedded Systems

- **Instrumentation and Machine Vision (17 Years)**
  - Spitzer Space Telescope, Unmanned Aerial Systems
  - Robotics, Arctic Sensors at U. of Alaska Anchorage

- **Software & Computer Engineering (17 Teaching)**
  - Shuttle Ascent/Entry Guidance, Deep Space
  - Intel and HPC, Storage, Networking Start-ups

- **Consulting (Founder)**
  - Graphics, Storage and Networking, UAS/UAV
  - Advanced RAID and Erasure Codes

LinkedIn

Featured Skills & Endorsements

- Embedded Systems: 99+
- Linux: 83
- Embedded Software: 59

1,662
Your connections
See all

© Sam Siewert
Learning Objectives

1D Interaction: CLI and Scripting - Useful in Data Centers, Automation

2D Interaction: Windowed GUIs, Mobile Touchscreen - Current State of Practice

3D Interaction: Machine Vision, AR/VR, Rendering, Domain Specific [CAD]

4D & AI: Intelligent Systems - Computers Acting more like Humans
  – Computer Vision for HCI
  – Natural Language Processing
  – Machine Learning and Rule-based inference

Usability: Metrics, Evaluation, Design for Non-functional requirements

Practice with 1D, 2D, 3D, and 4D (HCI from simple to complex)

Build or Improve a “Friendly” HCI Prototype
Current Research

- Real-Time Color Optimization with RIT [2011-2014]
- UAS Real-Time Video + GIS Overlays [2011-2013]
- DHS Arctic Domain Awareness – Smart Cameras [2013-15]
- ERAU **ICARUS** Drone Net [2016 – present]
Course Goals and Outline


- HCI – GUI, CLI, Visualization, Services, Tasks, Status, Monitoring, 2D/3D, I/O Devices, Shared-Control, Configuration, Collaboration, Social Networks, Animation, VR, AR … ?

- http://mercury.pr.erau.edu/~siewerts/cs415/

- Tues / Thurs – Lecture and Discussion

- Thurs – Q&A and Assignment Discussion

http://www.hcibook.com/e3/
At Dawn of Computing – Limited Interaction

- Batch Processing
- Punch Card
- Input/Output (Decks)
- Line Printer Output
- Status Lights, Tones

Not Much to Go on…

Computer Time Was More Important than User Time!

© Sam Siewert
Batch Processing Evolves and Continues …

- Tape (Analog, Digital)

- Graphic Printers (Laser, Ink-jet – 1980’s to 90’s)

- Scanners (OCR) – Xerox Document Services

- 3D Scanners and Printers (3D Systems Corp.)

- Publishing on Demand, Self-Publishing

© Sam Siewert
HCI and HMI (from Computer History)

1. ENIAC – Electronic Computer - 1946
2. Punch Cards, Switches, LEDs, Line Printer – 1950
3. Artificial Intelligence – 1956
4. Serial Terminals and Minicomputers - 1960
5. Command Line Interface (glass TTY, Unix) - 1965-70
7. Scripting (Unix) - 1970
11. MS DOS - 1981
12. WIMP (Windows, Icons, Menus, Pointers - Xerox PARC, Apple Mac) - 1985
17. Mobile WIMP (PDA, mobile touchscreen, voice recognition - iPhone) – 2005
18. SmartPhone, Tablet PC, SmartTV, VR CAVE, Gesturing - 2010
20. Future?

© Sam Siewert
Brief History of Interactive – 1970 to 2000

- Binary I/O, Switches & LEDs, **Joystick**

- **Command Line Interface (CLI)**
  - ASCII Terminals or Shells
  - Canonical Mode (Buffer until Return)
  - Non-Canonical Mode (Capture and Transmit Each Character)
  - Echo or No-Echo, e.g. password
  - Cmd.exe, Power Shell (MS), BASH (Linux, OS-X), Terminal

- **2D Mouse and Bit-mapped Window Manager (GUI)**
  - Xerox PARC (Palo Alto Research Center) – 1970
  - Mac OS with Toolbox (1979) – Jef Raskin
  - X-Windows (MIT) and Windows 3.x, NT, Win7, 8 …, 10 (Microsoft)

- **Web, Mobile and Wearable …**

© Sam Siewert
Interactive I/O Devices – Brief History

- **Trackball (Tom Cranston – 1952)**
  - Douglas Engelbart
  - Stanford Research Institute (SRI)

Newer Devices
- **Data Gloves**
- **Trackers, Motion Capture**
  - Leap Motion
- **Gesture Recognizers (PrimeSense)**
  - MS Kinect
  - Creative Cam
  - Occipital (3D Scanner)
- **Resistive Touch Screen (1975 ... Modern Tablet)**
- **Voice Command (Smartphones, Siri)**
- **3D, Data Gloves/Suits, Haptic Feedback, VR, AR, Holographic**
3D Animation and Motion Trackers

- Used for Animation in Digital Cinema – Motion Capture

- Compared to Stick Figure with Degrees of Freedom and Kinematics (Avars) – e.g. Toy Story

- Robotics & Kinematics (joint rotations to define position) and Inverse Kinematics (math optimization to get joint to position in X,Y,Z space)
Virtual and Augmented Reality

- **HMDs**
  - Monocular or Binocular
  - Replace 2D (3D rendered desktop frames) with Immersive Device
  - Changes Scene with Head Motion
  - Stereo – Frames specific to each Eye

- **CAVE (Cave Automatic Virtual Environment)**

- **Rudimentary AR - Hold Phone with Camera and Display**

- **Wearable Camera and Display – Google Glass Explorer Program**

- **First Person Sensing - Drones**

© Sam Siewert
NUC/Jetson or VM - Linux

**Option #1 – Use King 122 NUC / Jetson Lab**
- Learn Host and Embedded Linux
- Getting Started
- Jetson - Used for Self-Driving Cars
- NUC - LDAP login

**Option #2 – Virtual-Box Linux with Ubuntu 16.04 LTS**
- Use Windows or Macintosh PC
- Learn Linux, it’s Easy
- Ryan Sutton has SE VM Image you can Use!!
Embedded Linux

- Jetson **TK1** – Tegra SoC
  - 4 64-bit ARM Processor Cores
  - 192 Vector Co-processor Cores (Graphics Processing Unit)
- SoC Used in **Google’s Project Tango**, **NVIDIA Shield**
- Jetson Family - TK1, TX1, TX2
- AI Platforms - HPC Deep AI [Volta, Pascal, Tesla]
- One of Four Key Architectures - GP-GPU [Cluster-on-Chip, SoC FPGA, Tensor Processing Unit - TPU]

Logon NUC using ERAU account
Logon to Jetson from NUC IP+1
9 Stations – Shared with CEC450
Why HCI is Critical Going Forward…

- Non-Linear Growth in Computing Capability and Complexity

- Cooperative / Supervisory Automation – Safety Improvement

- Super Intelligence = AI + Human Intelligence
Intelligent Transportation - Safety

- Dr. Clark Chapman (SWRI - http://www.boulder.swri.edu/clark/ )
- Motivation for Driverless Car - Safety – Sebastian Thrun
- Safety HCI Monitors -> Shared Control -> Fully Autonomous

Driver to Driver Interaction, Fatigue, Distraction, Human Error, Autopilot?

Chances of dying from selected causes in the United States

<table>
<thead>
<tr>
<th>Cause</th>
<th>Chances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motor Vehicle accident</td>
<td>1 in 100</td>
</tr>
<tr>
<td>Homicide</td>
<td>1 in 300</td>
</tr>
<tr>
<td>Fire</td>
<td>1 in 800</td>
</tr>
<tr>
<td>Firearms accidents</td>
<td>1 in 2,500</td>
</tr>
<tr>
<td>Electrocution</td>
<td>1 in 5,000</td>
</tr>
<tr>
<td>Asteroid/Comet impact</td>
<td>1 in 20,000</td>
</tr>
<tr>
<td>Passenger aircraft crash</td>
<td>1 in 20,000</td>
</tr>
<tr>
<td>Flood</td>
<td>1 in 30,000</td>
</tr>
<tr>
<td>Tornado</td>
<td>1 in 60,000</td>
</tr>
</tbody>
</table>

Classic HCI Problem [HMI]
Autopilot or Assistant - AI
Machine Vision
Machine Learning
How We’ll Do It

1/3 Theory – Lectures, Reading, Discussion (On-going)
- Lectures related to HCI Textbook and Instructor’s Experience
- History of HCI and Current Best Practices
- Discussions

1/3 Practice – NUC/Jetson or VB-Linux
- Linux Coding (C and C++ or Java if you wish)
- Basic HCI Coding Skills
- HCI Applications - Original Implementations, Walk-throughs, Design Validations

1/3 Project [CLI, WIMP, Mobile App, or Advanced]
- Group Project to Build HCI Proof-of-Concept Application
- Useability Studies, Design of Experiments, Statistics
- Present Design and Prototype
Administrivia

Introductions
- Instructor (Office Hours)
- Students (Introductions) – Please do Collaborate, but cite well!
- Policies - http://mercury.pr.erau.edu/~siewerts/cs415/policies/

ERAU Canvas (https://ernie.erau.edu/ , https://erau.instructure.com/)
- Primarily for Assignments and Assignment Grading
- Mercury Website - http://mercury.pr.erau.edu/~siewerts/cs415/

Course Information
- Attendance & E-mail list (please sign up on sheet being passed around)
- Lecture Notes at http://mercury.pr.erau.edu/~siewerts/cs415/documents/
- Current Syllabus

Use NUC/Jetson Lab or VB-Linux - Windows for Python, Java, and Other VM Programming Languages

You can program in any language you wish - surprise me!
- Common Used - Java, Python, C/C++, C#
- We Use GUI Builders and GUI APIs
Tools We Will Use

- Linux [C, CLI]

- GUI Builders and GUI APIs [Python, Java, C/C++, C#]

- NC Renderman [3D Graphics] or Unity 3D

- Mobile App Dev Kits - Xamarin or AOS [Option For Project]

- OpenCV - Computer Vision + AI, Cameras (RGB, RGB Depth Mappers)
Linux Skills

Introduction Session
Linux Getting Started Help

- NUCs Run Ubuntu 14.04 and [MATE Desktop](#) [X-windows]

- Jetson Boards Run Ubuntu LTS - JetPack 3.1 Installed on All

- You Can Run the Same on VB-Linux on Windows, Max OS-X, etc.

- Linux is the NDK Layer in the AOS (Android OS) and Java is the SDK

- Much of Interactive Computing has Gone Mobile! Either iOS or AOS

- How-To’s
  1. [Development-Getting-Started.pdf](#)
  2. Makefiles by Example
  3. [Integrated Development Environments on Linux](#)
Next Time …

**HCI History – Come to Class Prepared to Discuss**
- Gurus (e.g. TBD)
- Organizations for HCI – [IEEE Systems, Man, Cybernetics – HCI](#), [ACM SIGCHI](#)

**Assignment #1 Discussion**
- I will Post Every Other Tuesday, We’ll Discuss, Due Following Week on Friday
- Late Assignments – 10% Penalty for Monday Turn-in, After Monday, only with Instructor Permission