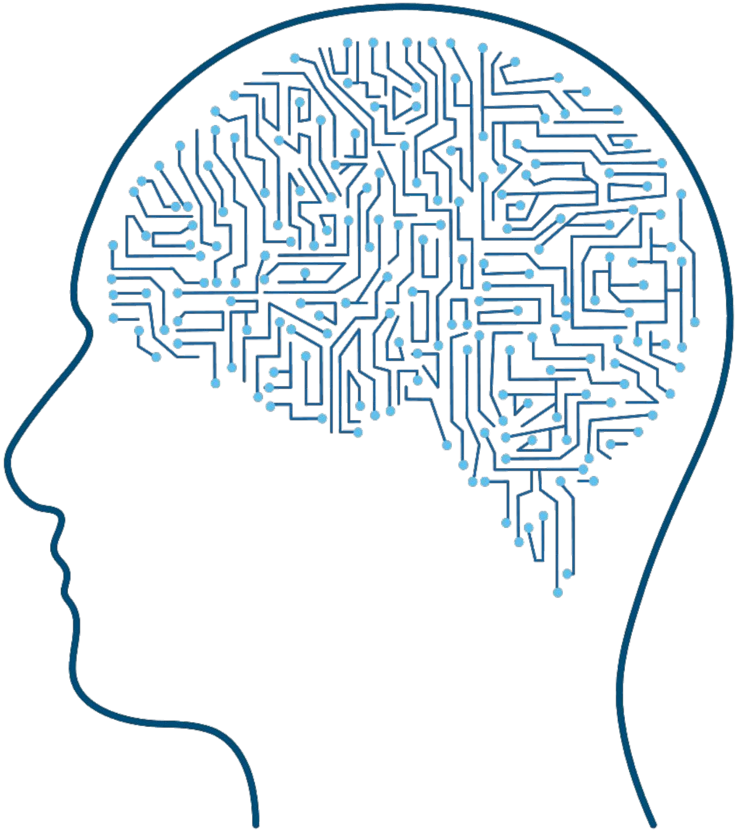
Ubiquitous Computing

Exercise Sessions

Kristof Van Laerhoven



Track 1: Distilling Scientific Content

- Slides will have published papers
- Each paper has a DOI
- How do I know this paper is good?
  - before publication: review process
  - after publication: trust
    - where was it published?
      - conference acceptance rate
      - journal impact factor
    - paper quality after a certain time is measured in citations

Track 1: Distilling Scientific Content

- Where was it published?
  - publisher often dictates the review process
    e.g., ACM *tends* to push for 3 quality reviews per paper
  - Conferences publish acceptance rates and are ranked
    (e.g.: A+, A, B, …)
  - Journals publish impact factor
- Who published it?
  - Hirsch-index (h-index), i-10 index, …
- Is it cited a lot?
  - Google scholar: https://scholar.google.com
  - Semantic scholar: http://semanticscholar.org
  - Scopus: https://www.scopus.com
  - ResearchGate: https://www.researchgate.net
  - ...

Track 1: Presenting Scientific Content

1. descriptive title
2. captivating abstract

   https://users.ece.cmu.edu/~koopman/essays/abstract.html
3. first figure + caption
4. accessible introduction
5. clear structure:
   - Introduction / motivation
   - Background / related work
   - Own work / evaluation
   - Conclusions
   - references

Using the eSense Wearable Earbud as a Light-Weight Robot Arm Controller

Henry Odoemelem
University of Siegen
henry.odoemelem@student.uni-siegen.de

Alexander Hölzemann
University of Siegen
alexander.hoelzemann@uni-siegen.de

Kristof Van Laerhoven
University of Siegen
kvl@eti.uni-siegen.de

ABSTRACT

Head motion-based interfaces for controlling robot arms in real time have been presented in both medical-oriented research as well as human-robot interaction. We present an especially minimal and low-cost solution that uses the eSense [1] ear-worn prototype as a small head-worn controller, enabling direct control of an inexpensive robot arm in the environment. We report on the hardware and software setup, as well as the experiment design and early results.

CCS CONCEPTS

• **Computer systems organization** → **Robotics**; • **Human-centered computing** → *Ubiquitous and mobile computing*.

ACM Reference Format:
Henry Odoemelem, Alexander Hölzemann, and Kristof Van Laerhoven. 2019. Using the eSense Wearable Earbud as a Light-Weight Robot Arm Controller. In *Proceedings of 1st International Workshop on Earable Computing (EarComp'19).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3345615.3361138

1 INTRODUCTION

The past decade has seen an increasing interest in developing and investigating the interface between humans and robots. For many scenarios in this research, precision and speed are crucial, with cost factors and size and mobility of such systems being less of a focus. In contrast, we present here a human-robot interface that aims at being minimal in terms of size and costs, and intend to investigate the trade-offs that are caused by this minimalism in terms of accuracy and speed. Our application domain is head motion-based robot control, as it is for instance required to enable tetraplegics

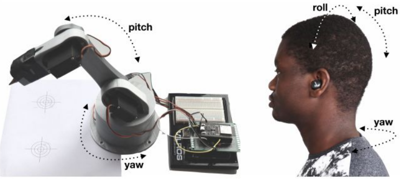


Figure 1: This work presents an affordable and light-weight system that uses an in-ear IMU to control a robot arm's yaw and pitch motions by moving the head.

to control a multi-degree of freedom robot arm in real-time using solely head motion (as for instance motivated in [4]).

2 DESIGN OF HARDWARE AND SOFTWARE

For our design, we use an open-source design 3-DOF robot arm, using STL Files available on [1]. This design uses 3 low-cost mg996r servo motors capable of 180 deg rotation and is powered by an Arduino (connected to a 9V dc power adapter). This design results in an extensible, re-programmable, and low-cost robot arm that costs about 70$ for all components. We use this model as this design explicitly represents an entry model for interactive control applications, where users need to adapt to the speed and accuracy of the servo motors.

In order to control the robot arm, IMU (Inertial Measurement Unit) data are sent via Bluetooth Low Energy(BLE) from an eSense ear-worn unit to a ESP32-WROOM-32D-equipped robot for local processing. eSense[2] is a multi-sensory earable platform for personal-scale behavioural analytics research. It is a True Wireless Stereo (TWS) earbud, composed of a

Track 1: Presenting Scientific Content

- **Use LaTeX**

- Cite papers properly: Bibtex + DOI
  - http://api.crossref.org/works/**10.3390/s19143079**/transform/application/x-bibtex
  - Copy the bibtex entry in the project's bib file
  - Refer to the bib file in the latex document

- https://webdemo.myscript.com/views/math.html

Track 1: Start your engines!

- Open the ACM template in OverLeaf: https://www.overleaf.com/latex/templates/association-for-computing-machinery-acm-sigplan-proceedings-template/rfvsrhgmghtc

- Write a 2-page review, citing results from papers, using images (mention source), structuring content

- ***Analyze*** a paper from the first conferences on ubiquitous and wearable computing (1997 and 1999, see moodle)

- ***Search and include*** content from 2 or more additional papers

Track 2: Rapid programming of wearables

- Programming for the Espruino Bangle 1: http://espruino.com/ide/ and https://banglejs.com
- Use the first tutorials to get familiar with the environment: https://www.espruino.com/Bangle.js+Clock
- Write a **clock face** (displaying day, month, year, hour, minute) that also displays the user's watch interactions over the day, i.e. how often the user pressed a button, or touched or swiped left or right (so 7 numbers in total) for the Bangle 1.
- ***Ensure that the user can quickly retrieve the needed information -> use color, font size, graphics, etc.***

Track 2: Rapid programming of wearables

```
// Load fonts
require("Font7x11Numeric7Seg").add(Graphics);

var IOString = "No Input";

function draw() {
  var d = new Date(), h = d.getHours(), m = d.getMinutes();
  var time = (" "+h).substr(-2) + ":" + ("0"+m).substr(-2);
  g.clear();
  g.setFont("7x11Numeric7Seg",4);
  g.setFontAlign(1,1); // align right bottom
  g.drawString(time, 160, 140, true /*clear background*/);
  g.setFont("7x11Numeric7Seg",2);
  g.drawString(("0"+d.getSeconds()).substr(-2), 190, 140, true);
  g.setFont("6x8");
  g.drawString(IOString, 70, 185, true); // draw the input
}

var secondInterval = setInterval(draw, 100);

setWatch(() => {
  IOString = "Button 1";
}, BTN1, {repeat: true});
```

```
setWatch(() => {
  IOString = "Button 2";
}, BTN2, {repeat: true, edge: "falling"});

setWatch(() => {
  IOString = "Button 3";
}, BTN3, {repeat: true});

Bangle.on('swipe', (sDir) => {
  if (sDir==1) {
    IOString = "Swipe left";
  } else {
    IOString = "Swipe right";
  }
});

Bangle.on('touch', (sDir) => {
  if (sDir==1) {
    IOString = "Touch left";
  } else {
    IOString = "Touch right";
  }
});
```