Lehrstuhl für Medieninformatik
Institut für Information und Medien, Sprache und Kultur
Universität Regensburg

Interaction Technology and Techniques
Assignment 9: Pattern Recognition

Summer semester 2015

Submission due: Sunday, 5. July 2015, 23:55

Hand in in groups of max. two.

Your task is to turn your WiiMote into a *magic wand* which supports selecting objects and performing actions on them by performing different gestures.

9.1: Read up on Wand Interaction

Download and read the paper *"XWand: UI for Intelligent Spaces"* (Wilson & Shafer, CHI 2003). Read appropriate further literature.

Concisely answer the following questions:

- How is the XWand tracked (provide all relevant details)?
- How are gestures recognized?
- What are the technical differences between the XWand and the WiiMote?

Hand in the following file:

xwand.txt: a plain-text file containing your answers

Points

- **2** Good answer to first question
- **2** Good answer to second question
- **2** Good answer to third question

9.2: Implement a Magic Wand

Write a Python application which turns a WiiMote into a magic wand. The application should be built on PyQtGraph. To this end, develop a scenario (see 9.3) where the user can interact with the environment or content on a screen using the WiiMote.

Example: *The user points at a virtual candle on the screen. As they press a button, the candle lights up. When the user shakes the WiiMote, the candle turns off.*

Implement *at least two* interaction techniques for the WiiMote.

Examples:

- point at an object to select it (using IR tracking)
- perform a gesture to invoke an action
- drag and drop object across a screen

- allow the user to train new gestures
- track hand gestures in front of the wand (using IR tracking)
- …

Implement *at least two* non-trivial nodes for your application.

Examples:

- train and recognize simple gestures using an SVM
- train and recognize simple gestures using the $1 recognizer
- track the WiiMote or an object using the IR camera
- display beautiful visualizations of the WiiMote movement
- …

Hand in the following files:

- **magic_wand.py**: a Python script that implements this application.
- additional files as needed for running the application

(Please do not hand in `wiimote.py` or `wiimote_node.py`.)

Points

- **1** The python script has been submitted, is not empty, and does not print out error messages.
- **2** The script correctly and robustly implements a flowchart.
- **2** The script implements at least two interaction techniques.
- **2** The flowchart includes at least two non-trivial nodes.
- **1** The script is well-structured and follows the Python style guide (PEP 8).

9.3: Document and Present your Magic Wand

Write a short report (one page), including a screenshot, which describes your usage scenario and application. Include abstract implementation details.

Furthermore, prepare a presentation of your demo for the next session (three minutes!)

Hand in the following file:

magic_wand.pdf: a short report of your application and usage scenario

Points

- **1** Well-written report
- **2** Interesting usage scenario
- **2** Usage scenario illustrated well
- **2** Error-free and fluent presentation

Submission

Submit via GRIPS until the deadline

All files should use UTF-8 encoding and Unix line breaks. Python files should use spaces instead of tabs.

`Have Fun!`