Computational Approaches to Interaction Design

John Williamson
University of Glasgow
Glasgow, UK
jhw@dcs.gla.ac.uk

Antti Oulasvirta
Aalto University
Helsinki, Finland
antti.oulasvirta@aalto.fi

Otmar Hilliges
ETH Zurich
Zurich, Switzerland
otmar.hilliges@inf.ethz.ch

Per Ola Kristensson
Cambridge University
Cambridge, UK
pok21@cam.ac.uk

CHI'17 Extended Abstracts, May 06-11, 2017, Denver, CO, USA
ACM 978-1-4503-4656-6/17/05.
http://dx.doi.org/10.1145/3027063.3027105

Abstract
This course will introduce computational methods in HCI. As interfaces become more sophisticated, designing them requires an exponentially expanding set of design decisions to be made. Hand-tuning can only explore a tiny fraction of the design space. Computational approaches are needed to delegate the minutiae of design and empower HCI researchers in building sophisticated, robust interfaces quickly and reliably. This course will introduce optimization and inference as core tools in HCI.

Author Keywords
computational; optimization; inference; machine learning

ACM Classification Keywords
H.5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces

Overview
The course will cover:

Optimization: solving interaction problems by deriving interface configurations which satisfy constraints and maximize performance criteria.

Inference: A principled and robust approach to designing a transformation from input to useful action.

This course will:

- demonstrate how computational approaches can turn new technologies into viable interfaces while reducing inconsistency;
- extend researchers' capabilities to build robust interactions across a wide range of contexts and devices;
- show how computational approaches can focus interaction design on the interesting work of specifying the questions, and letting computational methods resolve the answers.

Structure

This will be a fast-paced course surveying computational methods and contextualizing them in concrete HCI problems. The course will be organized as interactive lecture sessions with interleaved and very concentrated practical work. The material will be provided as interactive Jupyter (IPython) Notebooks, to allow participants to live execute examples during the lectures and immediately extend and apply the material present to practical problems. These notebooks will provide enough of a skeleton to allow participants to make rapid progress in applying the ideas presented. Although we will move quickly, we expect participants to follow along with live examples on their own machines during the courses. This is a format the instructors have refined over several years.

Intended audience

Intended Audience(s) We expect this to course to be primarily targeted at researchers and students in HCI. Industrial practitioners may also find the course valuable.

Content

The course will focus on optimization and inference and on applying these techniques to *concrete* HCI problems. The course will specifically look at vision-based interfaces, motion-based interaction, UI layout optimization, and efficient text entry. The outline is as follows:

- 1 Designing layouts with combinatorial optimization
- 2 Machine learning for vision-based interaction.
- 3 Robust motion-based interfaces using probabilistic state tracking
- 4 Probabilistic decoding for intelligent text entry

Optimization for design

1. Designing layouts with combinatorial optimization

This part introduces combinatorial optimization methods: algorithmically searching for combinations of design decisions that best fulfill given goodness criteria. After an overview, three hands-on exercises focus on core requirements: definition of design tasks as search problems, algorithmic approaches to searching the defined design spaces, and developing objective functions using design knowledge and theories. **After completing this section**, the attendee: has skills in formulating design problems such that an algorithm can help in exploring it; understands the requirements of this approach; understands how it can be tied to user-centered design and data-driven design.

Detailed contents and timing

Overview of optimization in HCI and UCD, including history, approaches, limitations, and state-of-the-art (10 mins); Defining design problems as search problems: Analysis and formal definition of layout problems (10); Searching design spaces algorithmically: random search and exact search methods, with keyboard layout as the example (10); Defining realistic objective functions: using design heuristics and models of human behavior and experience From Fitts' law to aesthetics.;Example: designing layouts for motor control, visual search, learnability, and clutter perception.;Wrap-up: Discussion of possibilities and pitfalls; Pointers to further literature (5)

Probabilistic inference for interaction

2. Machine learning for Human Behaivor Inference
Recovering intent from low-level sensor data is a core problem in many interactive systems. This section of the course will introduce the participants to the basic problems and available solutions surrounding human action recognition from sensor data. After an overview of relevant HCI problems we will give an introduction to succesful machine-learning models and how to apply these to common tasks such as static and dynamic gesture recognition. **After completing this section**, attendees will be able to: understand which types of HCI problems lend themselves well to data-driven approaches; will know how to collect and label data for training and testing; will have a basic understanding of which models to choose for which purpose; and will understand successful models (in the HCI context) in more detail.

Detailed contents and timing
Overview: Motivation why many activity recognition and sensing tasks should be tackled using data-driven algorithms and overview of existing approaches (10) ; Introduction to key concepts in supervised machine learning and practical concepts with relevance for HCI tasks (10); Introduction to select machine-learning models that have proven to be successful in a broad range of HCI projects (2x10); Discussion of end-to-end implementation of a sensing-based system leveraging above ML models. (10) ; Pointers to further literature and resources (5).

3. Robust motion-based interfaces using stochastic filtering
Probabilistic tracking is able to maintain a continuously evolving set of hypotheses about user intent, and update them according to evidence observed. This is a natural, principled way to robustly infer what a user wants to do from measurements observed from an input device. This section will cover applying sequential probabilistic tracking to a range of sensors. Sophisticated, high-dimensional, noisy sensors are becoming an important part of interactive systems. **After completing this section**, attendees will be able to: represent interaction problems as inference problems; apply probabilistic methods to robustly estimate the evolution of user intentions over time, even with very noisy input devices.

Detailed contents and timing
Overview: Bad motion-based interface examples. (10) ; Stochastic filtering of a mouse cursor using Kalman filtering. (10); The particle filter for flexible modeling. (10); Simultaneous inference across multiple sensors. (10); Example (10) ; Wrap-up (5)

4. Probabilistic decoding for intelligent text entry
This section will view text entry design as fundamentally a problem of decoding the user's intended text from noisy observations from a variety of sensors, such as touchscreens or depth sensors. The section will introduce key concepts in statistical language processing and give an overview of how to design and implement a probabilistic decoder for text entry. The section will illustrate how such decoding enable the design of several new text entry methods for a variety of use cases. **After completing this section**, attendees will be able to: understand how designing and implementing probabilistic decoders can result in faster, more flexible and more accurate text entry systems; be aware of key design decisions that affect decoding performance.

Detailed contents and timing
Introduction to probabilistic text entry (10); Key concepts in statistical language processing (10); Designing a prob-

abilistic text decoder (10); Implementing a decoder (10); Example implementation (10); Wrap-up (5)

Instructor Biographies

John Williamson

John Williamson a Lecturer at the University of Glasgow. He received his PhD from the University of Glasgow in 2006, after which he was awarded a SICSA Research Fellowship and a Lord Kelvin Adam Smith Fellowship in Sensor Systems, at the University of Glasgow. His research is focused on continuous interaction systems and computational approaches to human-computer interaction. This includes gesture recognition/motion sensor based interfaces, brain-computer interaction, and particularly developing new feedback mechanisms for such systems. He has received multiple Honorable Mention awards at CHI. He co-founded the international summer school series on Computaional Interaction.

Antti Oulasvirta

Antti Oulasvirta is an Associate Professor at Aalto University where he leads the User Interfaces research group. He was previously a Senior Researcher at the Max Planck Institute for Informatics at Saarland university. He received his doctorate in Cognitive Science from the University of Helsinki in 2006, after which he was a Fulbright Scholar at the School of Information in University of California-Berkeley in and a Senior Researcher at Helsinki Institute for Information Technology HIIT. He was awarded the ERC Starting Grant (2015-2020) for research on computational design of user interfaces. His work has been awarded the Best Paper Award and Best Paper Honorable Mention at CHI eight times between. At MobileHCI, he received the Best Note Award and the Most Influential Paper Award. He has held keynote talks on computational user interface design in NordiCHI, CoDIT, and EICS.

Otmar Hilliges

Otmar Hilliges is an Assistant Professor at ETH Zurich where he leads the Advanced Interactive Technologies Lab. Previously he was a Researcher at Microsoft Research Cambridge and prior to that a post-doctoral researcher at Microsoft Research. He was awarded a doctoral degree in Computer Science from LMU MÃijnchen, Germany (Summa Cum Laude 2009). His research interests are in Human Computer Interaction, interactive graphics and applied Machine Learning. He regularly publishes in the premier HCI conferences and his work has been awarded with multiple best paper awards at CHI, UIST, CSCW and ISMAR. He was awarded an ERC starting grant for his research on synthesis of interactive technologies.

Per Ola Kristensson

Per Ola Kristensson is a University Lecturer in the Department of Engineering at the University of Cambridge. He is interested in designing intelligent interactive systems that enable people to be more creative, expressive and satisfied in their daily lives. His PhD thesis (at Linköping University/IBM Almaden) was on gesture keyboard technology for touchscreens and in 2007 he co-founded ShapeWriter, Inc. to commercialize this technology. In 2008-2011 he was a Junior Research Fellow at the University of Cambridge and in 2011-2014 he was a Lecturer at the University of St. Andrews. In 2013 he was recognized as an Innovator Under 35 (TR35) by MIT Technology Review and appointed a Member of the Royal Society of Edinburgh Young Academy of Scotland. He has been awarded the ACM User Interface Software and Technology (UIST) Lasting Impact Award, the Royal Society of Edinburgh Early Career Prize in Physical Sciences and the Sir Thomas Makdougall Brisbane Medal.