The Aachen Lab Demo: From Fundamental Perception to Design Tools

Jan Borchers
Anke Brocker
Sebastian Hueber
Oliver Nowak
borchers@cs.rwth-aachen.de
brocker@cs.rwth-aachen.de
hueber@cs.rwth-aachen.de
nowak@cs.rwth-aachen.de
RWTH Aachen University
Aachen, Germany

René Schäfer
Adrian Wagner
Paul Preuschoff
Lea Schirp
rschaefer@cs.rwth-aachen.de
wagner@cs.rwth-aachen.de
paul.preuschoff@rwth-aachen.de
lea.schirp@rwth-aachen.de
RWTH Aachen University
Aachen, Germany

Figure 1: Samples of the seven demos we will show at our Aachen Lab Demo: a small selection of the textile icons that we created to determine the best textile fabrication techniques and shape combinations for eyes-free operation (left); a kinetic, shape-changing jewellery serves as an example of soft robots whose design our SoRoCAD tool supports (middle); and a large decorative lamp consisting of a 3D-printed wireframe with textile-covered surfaces and touch controls, an object created with the help of our *FabricFaces* fabrication tool and workflow (right).

ABSTRACT

This year, the Media Computing Group at RWTH Aachen University turns 20. We celebrate this anniversary with a Lab Interactivity Demo at CHI that showcases not past achievements, but the range of currently ongoing research at the lab. It features hands-on interactive demos ranging from fundamental research in perception and cognition with traditional devices, such as experiencing input latency and Dark Patterns, to new input and output techniques beyond the desktop, such as user-perspective rendering in handheld AR and interaction with time-based media through conducting, to physical interfaces and the tools and processes for their design and fabrication, such as textile icons and sliders, soft robotics, and 3D printing fabric-covered objects.

CHI EA '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9422-2/23/04.
https://doi.org/10.1145/3544549.3583937

CCS CONCEPTS

• **Human-centered computing** → **Interaction techniques**; **Interactive systems and tools**; **Haptic devices**.

KEYWORDS

latency, dark patterns, user-perspective rendering, conducting, e-textiles, soft robotics

ACM Reference Format:

Jan Borchers, Anke Brocker, Sebastian Hueber, Oliver Nowak, René Schäfer, Adrian Wagner, Paul Preuschoff, and Lea Schirp. 2023. The Aachen Lab Demo: From Fundamental Perception to Design Tools. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23), April 23–28, 2023, Hamburg, Germany.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3544549.3583937

1 INTRODUCTION

The Media Computing Group was established by Jan Borchers as an endowed chair of the B-IT Foundation in the Computer Science Department at RWTH Aachen University in Germany in 2003. It soon became Germany's first research group with sustained success in publishing its work at CHI. Jan went on to invent the new Interactivity name and format for hands-on demos at CHI in 2005, so it is only fitting that, for its 20th anniversary, the lab showcases its latest research as an Interactivity Lab Demo at CHI'23.

Instead of focusing on past achievements though, we showcase hands-on demos of current research at the lab. The demo is structured to move from fundamental research in perception and cognition with traditional devices, to new input and output techniques beyond the desktop, such as handheld AR and interaction with time-based media, to physical interfaces and the tools and processes for their design and fabrication. Below, we describe each research demo in more detail.

2 USB & HDMI CONSIDERED HARMFUL: END-TO-END LATENCY IN HISTORICAL AND MODERN COMPUTING TECHNOLOGY

Ever since we started pressing keys and clicking buttons on computers, the delay between user actions and system responses has been a subject of research. Many HCI experiments to this day include some form of reaction time measurement to a stimulus on screen. There is only little awareness ([6] provides an excellent recent overview) in the HCI community, however, that such measurements can be distorted by as much as 100 to 200 ms—enough to render their results useless—when executed on off-the-shelf modern hardware: While the lag through signal processing in software is negligible as long as complex 3D rendering pipelines are not involved, today's input and output devices add both constant lag and random jitter (variable lag) to the timing. Our measurements showed that typical LCD monitors, e.g., introduce a fairly constant delay on the order of 40 to 80 ms (depending on model and digital connection used) from the moment the software decides to output something on the screen, to the moment that output actually becomes visible. Note that this is not the delay caused by double-buffering graphics techniques or 3D rendering engines—these will add several *frames*, or dozens of ms, of delay to the signal.

Worse, however, is the delay introduced by simple USB input devices. In the case of a commercial USB joystick, we measured a random added delay of between 0 and 125 ms for multiple measurements on the same system. This is due to the low USB polling rate of many simple devices.

This is a rather recent problem. Before the advent of USB and LCD screens, computers did not have this issue. For example, an analog monitor or TV with a cathode ray tube (CRT) displays its video signal without any noticeable delay, since the incoming analog voltages directly control the brightness of the photon beam as it travels across the screen. At a typical refresh rate of 50 (PAL) to 60 (NTSC) Hertz, this means that the time between the computer sending a pixel to the screen and the monitor showing it will be between 0 and 16.7 (NTSC) or 0 and 20 (PAL) ms, depending on when the photon beam returns to painting that pixel. We confirmed these numbers in our own measurements using a CRT TV.

Similarly, early peripherals like the ubiquitous Atari-compatible joystick were designed so simple that their internal switches were directly closing connections between pins on their 9-pin "DB9" connector. These, in turn, connected only through some discrete digital logic ICs directly to pins on the computer's CPU. An example is the Kempston interface for the Sinclair ZX Spectrum, a popular home computer from 1982 [7].We measured the latency on an original Spectrum using a 50 Hz analog CRT PAL TV and a Kempston joystick interface with an oscilloscope, and found the expected equal distribution of latencies between 0 and 20 ms.

The impact of this latency is significant if, e.g., an HCI researcher is running an experiment that involves reaction times on modern hardware. While the naive assumption here is that no significant time passes between (a) the software displaying its stimulus and the stimulus appearing on screen (display latency), and (b) between the user pressing, e.g., a button in return and the computer registering that button press (input latency), in reality, there will be a combined average input and display latency of over 100 ms that also varies by 125 ms if a USB input device with the low default polling rate and an HDMI display are being used.

A perfect way to experience latency hands-on is to play a simple game, in which the user needs to time their input precisely with what they are seeing on screen (such as jumping over obstacles in a side-scrolling game). Note that this is different from a task in which the user needs to react to a sudden event (like characters suddenly appearing on screen). The reason is that that simple reaction task would not allow the user to adjust their input to constant latency. In the timing task, such adjustments are possible — but if random variation is added to the latency, it defeats the user's attempts to get used to and predict a certain amount of lag [5].

How can this problem be solved in order to run, e.g., a reaction time study with correct measurements? The first step is to analyze the problem. In our initial experiments, we connected one channel of an oscilloscope to the button switch in, e.g., a joystick, while the other was connected to a light-dependent resistor in series with a fixed resistor and a voltage source to form a voltage divider. The oscilloscope then shows the time difference between the button closing and the screen beginning to display the change. (We confirmed that the LDR begins to react with no noticeable delay, even though it takes several ms to fully saturate.)

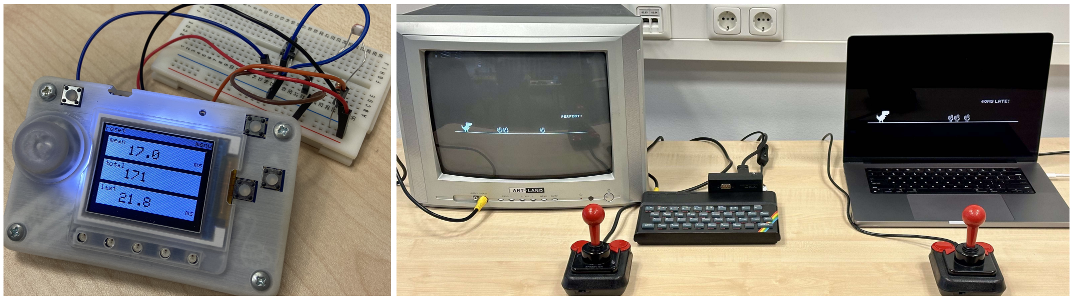

Figure 2: Our measurement tool for end-to-end constant latency and jitter (left). Our gaming setup lets visitors compare end-to-end latency on Jan's original Spectrum home computer from the 80s and a present-day M1 MacBook (right).

We then constructed a measurement device around a PyBadge embedded microcontroller (https://adafruit.com/product/4200) with the same LDR sensor that could automatically close contacts to trigger button presses and run a series of measurements (Fig. 2).

Once the problem is identified, there are several approaches that have been documented in the literature. One way to reduce USB input latency is to increase the USB polling rate to 1 kHz, resulting in only a 1 ms delay between button press and the event arriving in software [6]. Another is to avoid USB for input devices and to connect buttons and other components directly to an embedded microcontroller like an Arduino. Since these do not run an operating system, input latency can also be reduced to below 1 ms. On the output side, studies can resort back to using analog CRTs if displaying detailed content is essential, but these are becoming increasingly hard to find. For simple display purposes, LED rings, strips, or matrices that can be driven directly from a microcontroller like the Arduino can serve as virtually latency-free alternatives. Finally, FPGA-based emulators like the MiSTer (https://misterfpga.org) or Spectrum Next (https://specnext.com) can emulate early computers and their peripheral connectors with high accuracy, enabling users to connect old peripherals that avoid latency-riddled protocols.

Our Interactivity demo features our measurement tool and lets visitors play a little game both on Jan's original 80s Sinclair ZX Spectrum home computer, and on a modern laptop, to feel the difference for themselves (Fig. 2).

3 DARK PATTERNS

Our fundamental interactions with information are also governed by other subtle perceptual and cognitive cues. “Dark Patterns” on websites and in other software exploit this to trick users into decisions that go against their best interest, costing them money, time, or private data. They are becoming increasingly common and elaborate. While there are approaches that classify these patterns and investigate user awareness, there has been little work looking into visual countermeasures against them. We investigated concepts for six visual countermeasures against three common dark patterns, *Visual Interference*, *Confirmshaming*, and *Low Stock Messages*, which we embedded into a shopping website following a design by Bongard-Blanchy et al. [1]. Our results indicate two opposing forces for users: On the one hand, they dislike systems actively making hidden changes to the screen, preferring to be informed about the presence of dark patterns. On the other hand, they do not want applications to become visually cluttered, as this may impact their productivity. However, different types of patterns require different countermeasures, and individual preferences vary strongly.

Our Interactivity demo presents visitors with our screenshot prototype that flags, highlights, hides, and/or explains these Dark Patterns in the underlying shopping website. Visitors can gauge their own reactions to these different measures and engage in discussions regarding the field of manipulative designs.

4 USER-AWARE AUGMENTED REALITY

Moving beyond the desktop, we now look at handheld augmented reality. Here, users have only a small screen to see the augmented scene, making decisions about scene layout and rendering techniques crucial. Traditional device-perspective rendering (DPR) uses the device camera's full field of view, enabling fast scene exploration, but ignoring what the user sees around the device screen. In contrast, user-perspective rendering (UPR) emulates the feeling of looking through the device like a glass pane, which enhances depth perception, but severely limits the field of view in which virtual objects are displayed, impeding scene exploration and search.

We introduce the notion of User-Aware AR (see Figure 3). By following the principles of UPR, but pretending the device is larger than it actually is, it combines the strengths of UPR and DPR. In two studies, we have found that User-Aware AR imitating a 50% larger device successfully achieves both enhanced depth perception and fast scene exploration in typical search and selection tasks.

In our Interactivity Lab demo, visitors can try out all three methods hands-on on smartphones we provide. The results of our study are currently under submission at the MobileHCI conference.

5 PERSONAL ORCHESTRA RELOADED

Moving beyond basic questions of perception and cognition, our next demo is an example of a unique way to interact with time-based media, like a video recording. It lets visitors conduct a video of the Vienna Philharmonic orchestra using a simple reflective baton and infrared camera. This conducting exhibit was first developed more than 20 years ago [2] and has been on display at the HOUSE OF MUSIC, a museum in Vienna, with millions of visitors. Last year, however, we rewrote the system completely from scratch, and it now offers a fidelity in conducting that we consider unique in the world. But it also features realistic error messages: If visitors conduct badly, the orchestra stops playing and complains.

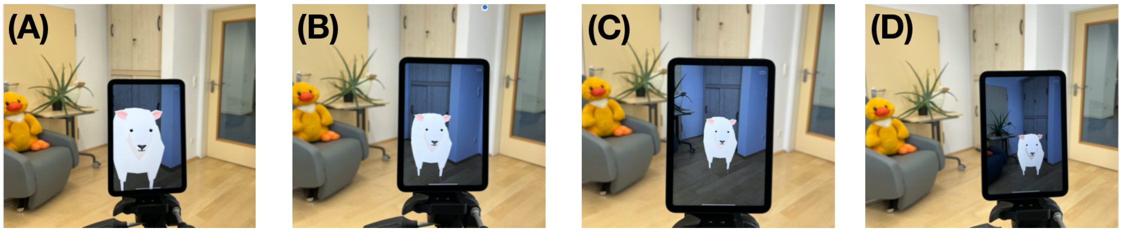

Figure 3: Different rendering techniques when holding a smartphone at an angle. UPR (A) and DPR (D) differ in both the orientation from which the camera looks at the scene and their FOV. In UPR (A), the device aims for virtual transparency, so that the cupboard in the background is aligned between device viewport and peripheral vision. However, the FOV is limited and the sheep is slightly too large to fit on the screen. In (D) there is a noticeable offset between screen and real world. Our user-aware techniques (B, C) serve as a middle ground between the two, combining a large FOV with approximate alignment.

6 TEXTILE INTERFACES IN THE HOME

Our last group of exhibits is all about fabrication and physical interfaces. We start with a set of textile sliders. We created these in a research project about using rich interactive materials in the home. In our Interactivity lab demo, visitors can feel the impact of different design decisions on eyes-free usability regarding form factors and tickmark designs. This demo accompanies our CHI'22 full paper [4]. Next, we show how to design textile *icons* using different fabrication variants so that they are easy to tell apart without looking. This demo accompanies our accepted CHI'23 full paper. It also includes some unpublished further explorations.

7 SOFT ROBOTICS

Soft robots use soft, flexible materials and elastic actuation mechanisms to become more adaptable and tolerant to unknown environments, and safer for human-machine interaction, than rigid robots. Pneumatic soft robots can be fabricated using affordable materials and technologies such as silicone and 3D printing, making them an attractive choice for research and DIY projects. However, their design is still highly unintuitive, and at two days, design iterations take prohibitively long: The behavior of, e.g., a pneumatic silicone gripper only becomes apparent after designing and 3D printing its mold, casting, curing, assembling, and testing it. Our SoRoCAD design tool supports a Maker-friendly soft robotics fabrication pipeline that incorporates simulating the final actuation into the design process. First user tests indicate that SoRoCAD encodes design expertise for pneumatic soft robotics in a way that enables novice users to rapidly design soft robots for specified movements.

In our Interactity Lab demo, we demonstrate SoRoCAD and show several working interactive soft robots that were designed with its help. SoRoCAD was first published as a Late-Breaking Work at CHI'22 [3], and a related study at CHI'23.

8 FABRIC FACES

Finally, we look at the challenge of mixing materials in personal fabrication. FabricFaces introduces a Personal Fabrication workflow to easily create feature-rich 3D objects with textile-covered surfaces. Our approach unfolds a 3D model into a series of flat frames with connectors, which are then 3D-printed onto a piece of fabric, and folded manually into the shape of the original model. This opens up an accessible way to incorporate established 2D textile workflows, such as embroidery, using color patterns, and combining different fabrics, when creating 3D objects. FabricFaces objects can also be flattened again easily for transport and storage. We provide an open-source plugin for the common 3D tool Blender. It enables a one-click workflow to turn a user-provided model into 3D printer instructions, textile cut patterns, and connector support. Generated frames can be refined quickly and iteratively through previews and extensive options for manual intervention.

In our Interactivity Lab demo, we show our editor, fabrication process, and samples to fold and unfold. The project is described in more detail in a Late-Breaking Work published at CHI'23.

ACKNOWLEDGMENTS

Special thanks to Lennart Becker and all our other students who helped make these research projects and associated demos a reality.

REFERENCES

[1] Kerstin Bongard-Blanchy, Arianna Rossi, Salvador Rivas, Sophie Doublet, Vincent Koenig, and Gabriele Lenzini. 2021. "I Am Definitely Manipulated, Even When I Am Aware of It. It's Ridiculous!" - Dark Patterns from the End-User Perspective. In *Designing Interactive Systems Conference 2021* (Virtual Event, USA) *(DIS '21)*. Association for Computing Machinery, New York, NY, USA, 763–776. https://doi.org/10.1145/3461778.3462086

[2] Jan Borchers, Wolfgang Samminger, and Max Mühlhäuser. 2002. Engineering a realistic real-time conducting system for the audio/video rendering of a real orchestra. In *Proceedings of the IEEE Fourth International Symposium on Multimedia Software Engineering (MSE '02)*. IEEE, IEEE, Newport Beach, CA, USA, 352–362. https://doi.org/10.1109/MMSE.2002.1181633

[3] Anke Brocker, Jakob Strüver, Simon Voelker, and Jan Borchers. 2022. SoRoCAD: A Design Tool for the Building Blocks of Pneumatic Soft Robotics. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, Article 330, 7 pages. https://doi.org/10.1145/3491101.3519770

[4] Oliver Nowak, René Schäfer, Anke Brocker, Philipp Wacker, and Jan Borchers. 2022. Shaping Textile Sliders: An Evaluation of Form Factors and Tick Marks for Textile Sliders. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 214, 14 pages. https://doi.org/10.1145/3491102.3517473

[5] Robert J. Teather, Andriy Pavlovych, Wolfgang Stuerzlinger, and I. Scott MacKenzie. 2009. Effects of Tracking Technology, Latency, and Spatial Jitter on Object Movement. In *Proceedings of the 2009 IEEE Symposium on 3D User Interfaces (3DUI '09)*. IEEE Computer Society, USA, 43–50. https://doi.org/10.1109/3DUI.2009.4811204

[6] Raphael Wimmer, Andreas Schmid, and Florian Bockes. 2019. On the Latency of USB-Connected Input Devices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300650

[7] Colin Woodcock. 2012. *ZX Spectrum on Your PC*. LULU PRESS INC. https://books.google.de/books?id=4yG4AwAAQBAJ