Crawling traces

An interactive projection to manipulate public space.
Concept and technical details.

Deborah Schmidt
March 27, 2013

TECHNISCHE
UNIVERSITÄT
DRESDEN

Fakultät Informatik
Professur Mediengestaltung

Author

Deborah Schmidt is a computer scientist who is enthusiastic about media manipulation, tinkering and open source software and hardware. She grew up doing analog stuff and a lot of painting. Her interest in mathematics and arts lead her to enter the digital world by studying media computer science at TU Dresden. She graduated with distinction in 2013. During her studies she co-founded KAZOOSH!, an open, non-commercial community for creating computational, electronic and artistic installations together in a group. She organized and participated in workshops and project weeks in the field of interactive installations and interfaces. The results were shown in exhibitions, conferences and festivals. “Crawling traces” is part of her diploma thesis. Currently she works at TU Dresden as a part of an interdisciplinary group to establish a postgraduate course uniting computer science and arts. Besides that, she writes software with openFrameworks to create generative music visualizations and facade mappings.

Contents

Short description

Cities are highly occupied by advertisement. The aim of this project is to give everyone a chance to manipulate their public environment legally and uncensored.

The main part is a projection on a facade. The audience is able to manipulate the projection. Each participant can draw one solid line.

There are two interaction methods. First, you can open a URL with your mobile phone, which is shown in the projection. You get a colour assigned and you navigate the drawing point with simple gestures on the display. The second interface consists of two wheels placed in sight of the facade. They also control one drawing point. One wheel can move the point up and down, the other one left and right, like in the game Etch A Sketch.

The projection is adaptable to the facade. Participants are not able to draw over window areas. The drawing lines are visually enhanced by connecting adjacent drawing positions with each other and with points on close-by window borders. This leads to net-like structures which can be customized visually within the projection software to match with the underground texture. The result is an organic, growing image of traces on the wall.

The implemented software and hardware parts are based on open source software and hardware. The installation was presented at the CYNETART'12 in Dresden, Germany on two different facades simultaneously (SCHMIDT 2012). The simplicity of the concept gave everyone a chance to be free in their drawings and even lead to the spontaneous creation of user game concepts. The response was marvelous.

1 Motivation

This work is based on the analysis of the townscape and possibilities to manipulate public space in general. Particulary relating to temporary media, advertisment highly occupies public areas. On the other side, art objects in the public are done by professional artists and / or supervised (censored) by the city government. But should public space not be shaped by public life? How many people walking through our cities are monetary capable of publishing their own advertising or are skilled to be an artist? However, many people show their desire to influence their environment in the field of graffiti and street art. Even though many techniques are not as durable as spraying, most of them are illegal. Projections can therefore be used to create large sized, transient images in the public. This work aims to give any audience the chance to draw content on public areas as freely as possible, obtaining an artistic unity throughout the visualization process.

2

Concept

2.1 Overview

Setup. One projector and a computer is placed at each exhibition location (see figure 2.1, first row). The computer receives the interaction signals of the participants, creates the projection image and provides adaption of the visualization to the surroundings of the projection. This program is hereafter called projection software.

2.2 Interaction

Trace route. The drawing of the participant within the projection consists of one solid trace. This trace can be continued arbitrary and unbroken, forming free shapes, structures or signs. The starting point is adaptable to the projection substrate, enabling you to chose prominent spots on the wall. The current position of the drawer is highlighted with a circle. The contexture of virtual traces and real wall substrate leads to different paradigms of morphology. Superimposed traces fuse into one planar structure, while adjacent elements of the facade remain separate (cf. GROH 2012). Gaps and textures even get exposed by the projection in the dark.

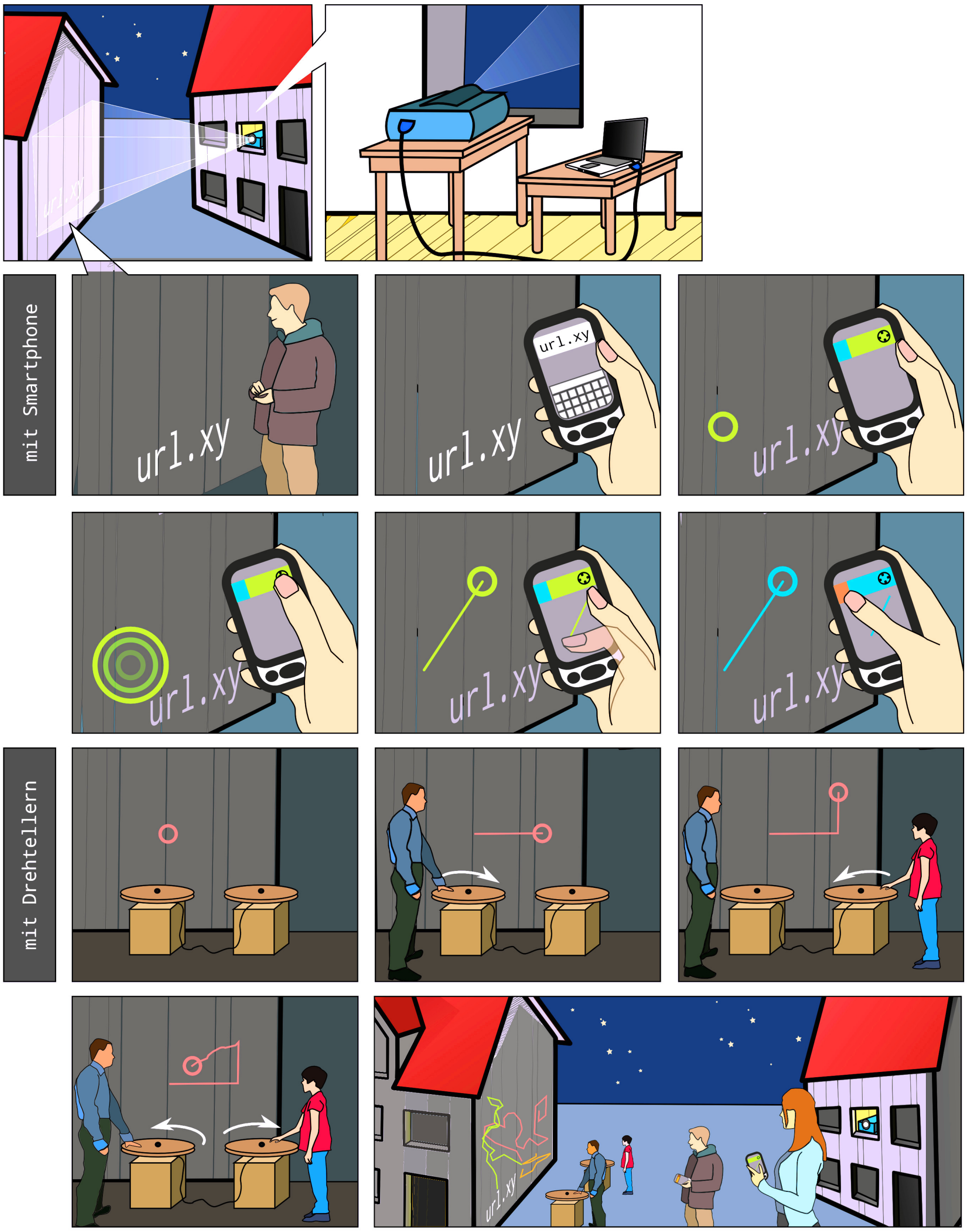

Figure 2.1: Concept - Storyboard to visualize the interaction processes.

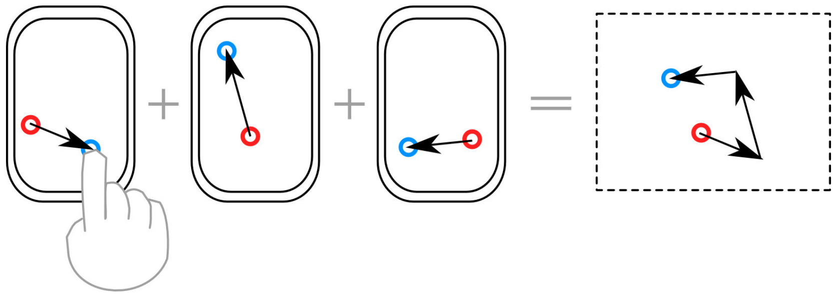

Figure 2.2: Gesture on the phone to create a drawing. The starting point of the gesture is coloured red, the ending point blue. The right image shows the resulting drawing within the projection including its starting and current drawing position.

Interfaces. The projection software receives unencrypted signals. The signals contain an identification of the participant and his triggered drawing positions. Such a signal can basically be sent by any device being connected to the projection software. This enables the enhancement and creation of interfaces independent of the installation developer.

There are two interaction methods. They are based on the analysis of everyday interaction types in the public. They are visualized in figure 2.1 in the second row. The aim of the interface is to navigate the drawing point to any coordinate on the facade. Therefore two continuous degrees of freedom are necessary. In addition, as many bystanders as possible should get a chance to participate.

Mobile phones are steady companions of people in public spaces. These devices provide increasingly touch-sensitive surfaces and a mobile connection to the Internet. This concept employs a simple touch interaction on the display to manipulate traces on the wall. Every smart-phone connected to the projection software represents one participant. By touching the display, a start position is defined. The end position is defined by the point, where the gesture ist finished. The straight line between these two points matches the segment which is appended to the trace within the projection (see figure 2.2). One advantage of this method is the successively composition of the image by the participant without being constrained by the display size of the mobile device.

A second interface is based on the use of rotary objects which are already part of our daily routines. The circle form can be found in wheels of traffic signs, cars or bikes and the rotational process is intuitive. The principle of drawing works analog to the toy Etch A Sketch. Two rotatable elements represent the main axes of a plane movement. De facto, two turntables play one trace on the wall and are located in the sight of the projection. One plate can move the drawing point up and down, the other one left and right (see figure 2.1, row 4 and 5). Because this interface can be used by two people simultaneously, their communication plays an important role.

Motivation. Since the projection is lacking in content without participants, a text-based motivation to interact is included in the projection. It also shows a text with a technical note about the connection with the smart phone to the projection.

Collaboration. Regarding communication processes, participants are left to their fate. Their traces may intersect within the projection, but they are free to cooperate, to compete or to ignore each other. The traces are distinguished by different colours. Each participant is informed about his colour through the mobile interface. He can temporarily highlight his position on the wall using the interface. In this case, the circle at the drawing position pulses (see figure 2.1, third row).

2.3 Placement

Distribution. The projection can be installed on any places simultaneously. In the present concept, the traces of all participants are actively shown on all projections. The installation could as well be used to draw together with people from far-away places. The stations and interfaces are connected over a server to exchange drawing data.

Awareness of the substrate. The drawing of traces takes place in the context of the projections environment. The drawing area can be skewed. Walls and windows hold very different characteristics. While walls outdoors are massive and repulsing, windows grant insight into private spaces. They may emit light at night. Areas like windows are therefore declared as save areas and left out from the drawing area. Traces may not cross the edge of a window. The mapping of the areas is part of the projection software.

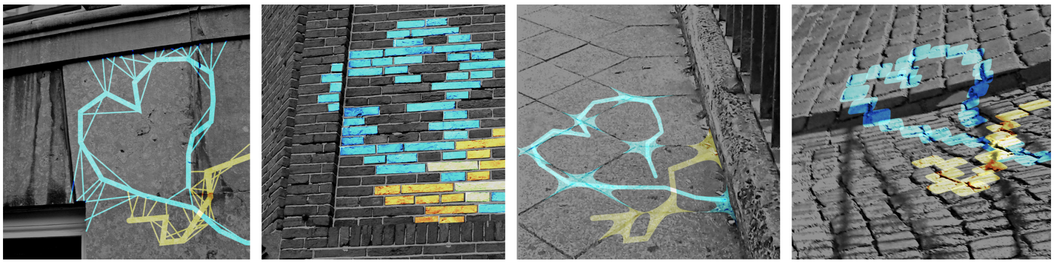

Figure 2.3: Facades and places - line design.

2.4 Visualization

Design. Physical characteristics of the projection substrate are also taken into account for image visualization and line design. Figure 2.3 shows four suggestions for adapted trace visualization. In this work, the first one is implemented. The drawing lines are visually enhanced by connecting adjacent drawing positions with each other and with points on close-by window borders. This leads to net-like structures which can be customized visually within the projection software to match with the underground texture. The result is an organic, growing image of traces on the wall.

3

Implementation

3.1 Signal communication

One essential part of the installation is the network to exchange drawing lines over the server. The signal process is displayed as a schematic view in figure 3.1. The server is identifiable over the World Wide Web by an address. Over the ports the server receives signals. Participants are able to send messages over the Transmission Control Protocol (TCP). Such a connection can be accomplished with a mobile phone as well as with native applications. The content of the messages are exemplary entered in figure 3.1. The projection software registers with the server. The server saves the IP and the port of the application which sent the registration. From that moment, all drawing actions from participants are sent to the projection application over this connection.

The server is implemented using *Node.js*, a Javascript platform to realize efficient, scalable network applications (NODE.JS 2013). *Node.js* is MIT-licensed and running under many different operating systems.

3.2 Projection software

The aim of the projection software is to monitor the connected participants, to map the drawing onto the right spot of the facade and to place window areas. Additionally you can adjust the visual characteristics of the projected traces. The projection software currently runs on Ubuntu x64 and is realized with openFrameworks (LIEBERMAN, WATSON, and CASTRO

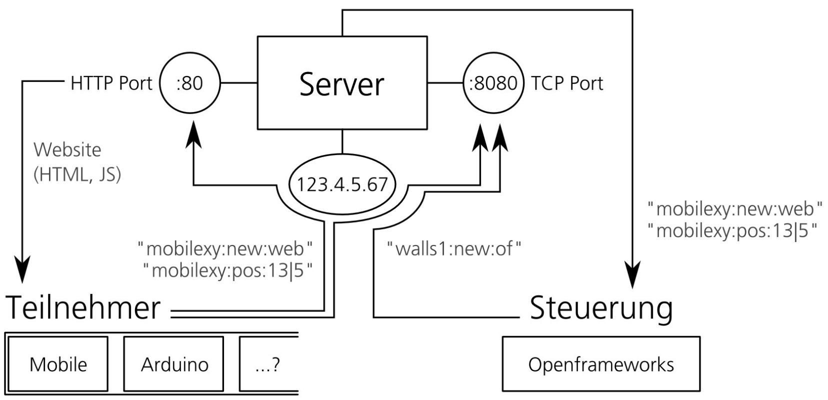

Figure 3.1: Implementation: Signal process communication schema.

2012). The kernel of Linux is in contrast to other operating system based on a free license and on that account being deployed in this scenario as a runtime environment. OpenFrameworks is based on C++ and also licensed under the MIT. It can be expanded by various addons. The development of the graphical user interface is realized by using a modified version of the addon *ofxUI* (ALI 2012).

The screenshots in this documentation are inverted due to print optimization. The color values are preserved to keep semantic colorations.

The projection software consists of three essential areas of activity:

- **Server**: Visual display of the server status, connected interfaces and activities of participants.
- **Layer**: Configuration of the drawing traces' visualization.
- **Mapping**: Adaption of the image onto the projection environment.

It would go beyond the scope to explain the details of the user interface. Instead, some effects in the drawing appearance and quality related to adjustable parameters will be shown in chapter 4.

3.3 Interfaces

3.3.1 Webpage

As already mentioned, people can draw traces via their mobile phone. The webpage interface is implemented with basic HTML, CSS and Javascript. Its visualisation is kept plain and clear. On the first call the page shows usage hints. After touching the display, the hints fade and a symbol appears, motivating you to move the finger. While dragging, a straight line between the starting position and the current touch position is shown. It represents the segment that is appended to the drawing when the dragging gesture is finished.

The top region of the page visualizes the individual drawing color of the participant. It can be changed by touching the colored button on the top left. The button on the top right edge lets you highlight your current drawing position on the wall (figure 3.2).

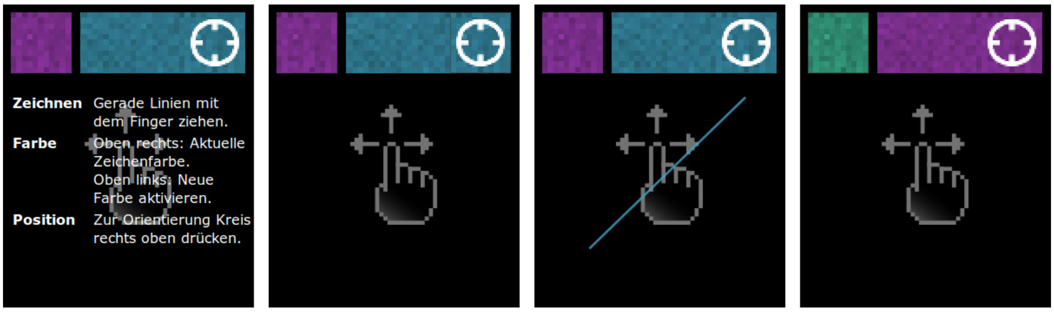

Figure 3.2: Implementation: Mobile interaction.

3.3.2 Wheels

The second kind of implemented interface is a pair of wheels as shown in figure 3.3. One wheel moves a drawing point on the wall horizontally, the other one moves it vertically.

The low cost basis of this interface are two rotary cheeseboards from IKEA (IKEA 2013). To measure their rotation, each board is equipped with a printed black and white code wheel. It sticks on the rotating plate. Two infrared emitters and sensors are attached to the fixed part of the board. They are wired to an Arduino Leonardo (SMARTPROJECTS, SPARKFUN ELECTRONICS, and GRAVITECH 2012). The microcontroller is programmed to calculate the rotational direction and speed.

It is extended by a Redfly-Shield to be able to communicate with the server via Wifi (WATTEROTT ELECTRONIC 2013). Everytime someone rotates one or both plates, a new drawing position is sent to the server. It communicates with the server through the same messages the website interface does. This process can be achieved with arbitrary interfaces which support TCP.

The electronics is shielded by kitchen boxes. They provide easy access while protecting sensitive components. This is not optimized visually, but it makes the interface useable in lively places and in the outdoor space despite its prototypical status.

Figure 3.3: Implementation: Prototype of the rotary tables.

4

Exemplary outputs

4.1 Screenshots

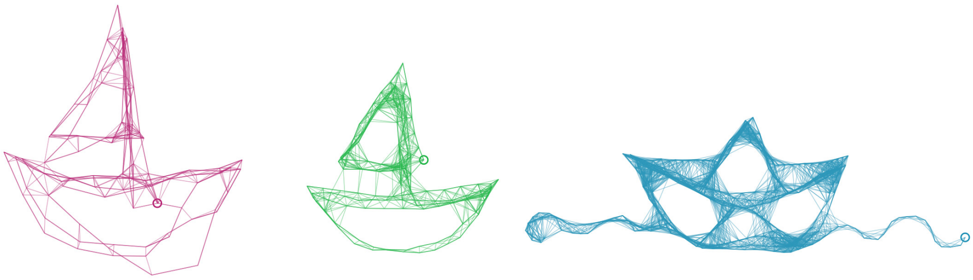

Figure 4.1: These drawings were made using different step sizes. The step size describes the scaling of the gesture on the mobile phone to the line segment within the drawing. If a gesture results in a large line segment, the drawing becomes unprecise. If gestures produce small line segments, the drawing is more delicate. Multiple narrow drawing points also lead to a tighter net structure.

Figure 4.2: Screenshot: Free drawings created with the mobile phone interface.

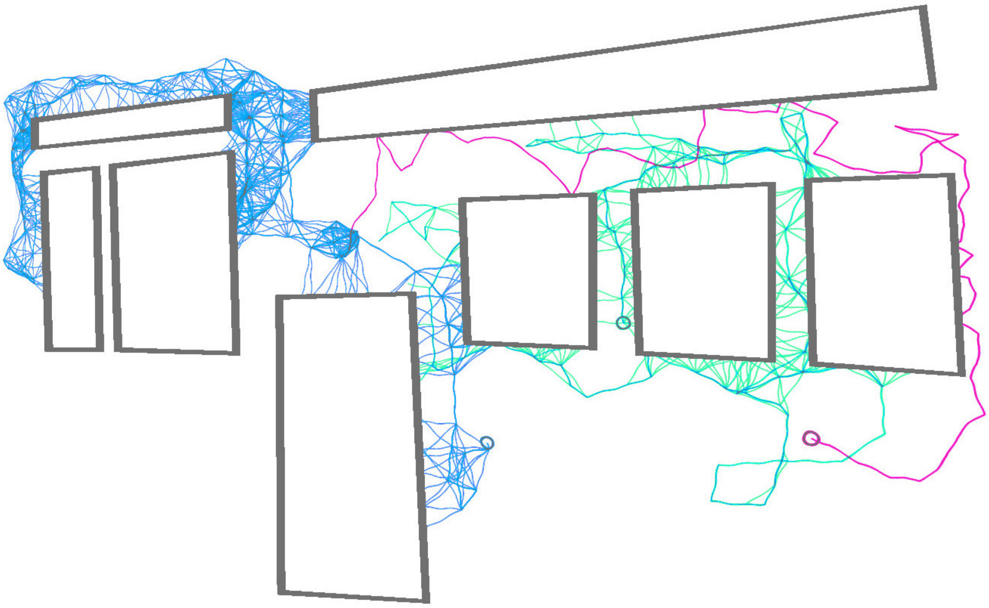

Figure 4.3: Screenshot: Drawing traces with different connection distances. The pink line is calibrated to create no connection lines at all. The blue line allows connections between drawing points that are more distant than connected points within the cyan line. This screenshot also illustrates how traces are not allowed to cross window borders.

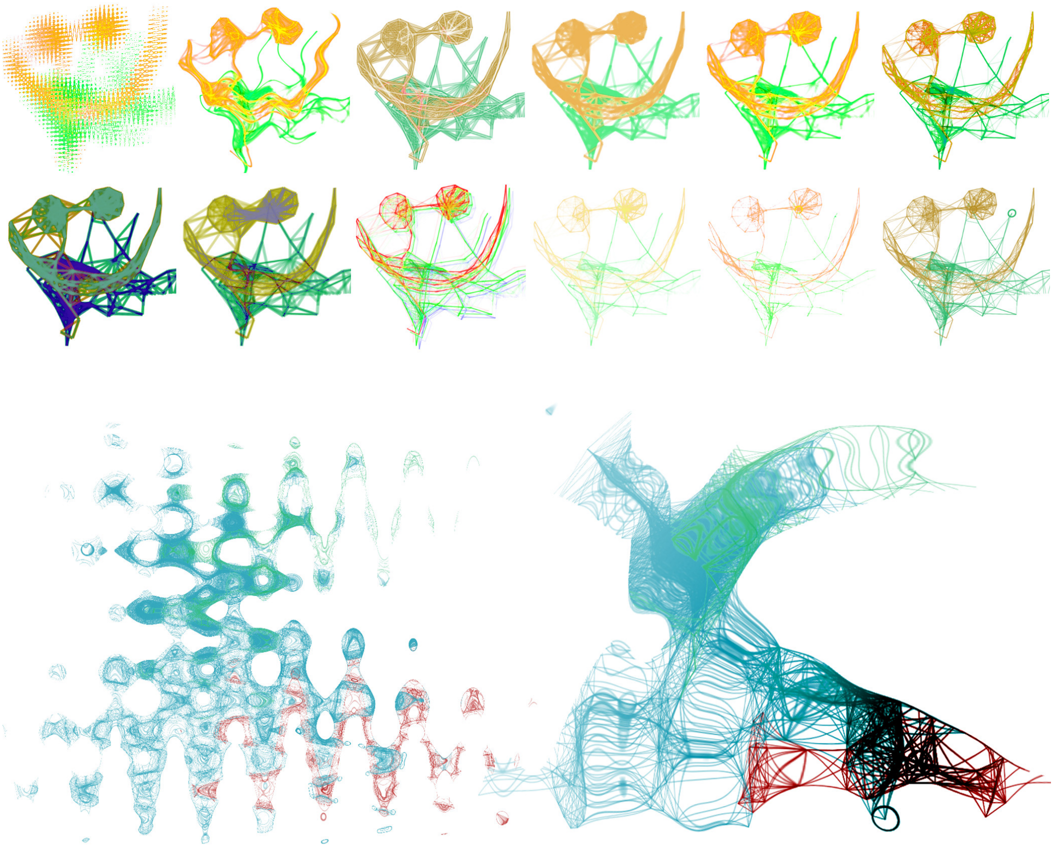

Figure 4.4: Screenshot: On top, you see the same drawing of two traces to which different effects are adapted. Those effects relate to the traces' width, opacity and various blendings of the drawing with itself. The bottom two screenshots show shaders that are applied on the drawing to create natural structure effects like waves. These effects can help to adapt a drawing to its substrate.

4.2 In action

Figure 4.5: In action: Facades. Photographs: David Pinzer.

Figure 4.6: In action: Textual drawings.

Figure 4.7: In action: Image drawings.

Figure 4.8: In action: Rotary wheel usage. Left: Placing of the boards in the bar Bon Voyage. Middle: White traces. Right: Purple traces.

Figure 4.9: In action: Traces playing with the windows. Photograph top: David Pinzer.

Bibliography

ALI, Reza (2012). *ofxUI: A User Interface Addon for OF*. [Online; eingesehen am 14-Jan-2013]. URL: http://www.syedrezaali.com/blog/?p=2172.

GROH, Rainer (2012). *MG II Schemata*. Folien der Vorlesung Mediengestaltung für Fortgeschrittene.

IKEA (2013). *SNUDDA*. [Online; eingesehen am 13-Jan-2013]. URL: http://www.ikea.com/de/de/catalog/products/90074483/.

LIEBERMAN, Zach, Theodore WATSON, and Arturo CASTRO (2012). *openFrameworks. Open-source C++ toolkit for creative coding*. [Online; eingesehen am 24-Jul-2012]. URL: http://www.openframeworks.cc/.

NODE.JS (2013). *Website der Plattform*. [Online; eingesehen am 13-Jan-2013]. URL: http://nodejs.org/.

SCHMIDT, Deborah (2012). "Kartesische Räder". In: *CYNETART 2012*, pp. 98–99.

SMARTPROJECTS, SPARKFUN ELECTRONICS, and GRAVITECH (2012). *Arduino. Open-source electronics prototyping platform*. [Online; eingesehen am 24-Jul-2012]. URL: http://arduino.cc/.

WATTEROTT ELECTRONIC (2013). *RedFly-Shield - Wifi/WLAN*. [Online; eingesehen am 13-Jan-2013]. URL: http://www.watterott.com/de/Arduino-RedFly-Shield.