A Scalable Measurement Tool to Capture Back-of-Device Touch Data

Bachelor's Thesis
submitted to the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
José Mínguez Garrigues

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Francisco Abad Cerdá

Registration date: 13.05.2016
Submission date: 28.09.2016

Eidesstattliche Versicherung

______________________________ ______________________________

Name, Vorname Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum Unterschrift

Contents

List of Figures

List of Tables

Abstract

In this thesis we present a scalable measurement tool that captures back-of-device touch data. We created a series of prototypes that mimic different mobile devices' screen sizes that share the same form factor $(16 : 9)$. This tool is compatible with scaled devices and can be used in future research to determine how accurate humans can select targets in the back of a device using an absolute direct input. We also provide design guidelines, so that everyone can scale our prototype designs to create bigger or smaller models that work with our tool.

Überblick

In dieser Arbeit steht ein skalierbares Messgerät zur Erfassung der Dateneingabe per Touch auf der Rückseite des Geräts im Vordergrund. Die entsprechende Prototypenserie emuliert verschiedene Bildschirmauflösungen von Mobilgeräten im Seitenverhältnis $16 : 9$. Die vorgestellte Soft- und Hardware ist skalierbar und kann in zukünftigen Forschungsprojekten eingesetzt werden, um zu ermitteln, mit welcher Genauigkeit Menschen in der Lage sind per absolutem direkten Input Targets auf der Rückseite des Geräts auszuwählen. Außerdem werden Richtlinien zur Skalierung der Prototypen zur Verfügung gestellt, mit denen der Anwender größere bzw. kleinere Modelle erstellen kann, welche mit dem bereitgestellten Werkzeug kompatibel sind.

Acknowledgements

I would like to thank Prof. Dr. Jan Borchers for allowing me to be part of his chair to work on this project for the last months.

I would also like to thank my supervisor Christian Corsten, M. Sc. for his time, feedback and constructive criticism on the topic, it was a pleasure to work with you.

Also thanks to Jan Thar for helping me out in the lab to create the prototypes and to Francisco Abad Cerdá for being my second examiner.

Finally, I would like to thank my family for supporting me in my years of study, especially during the last months.

Conventions

Throughout this thesis we use the following conventions.

Definitions of technical terms or short excursus are set off in coloured boxes.

EXCURSUS:
Excursus are detailed discussions of a particular point in a book, usually in an appendix, or digressions in a written text.

Definition: *Excursus*

The whole thesis is written in American English. We use the plural form for the first person. Unidentified third persons are described in female form.

Chapter 1

Introduction

Smartphones and tablets have become over the last years one of the main media consumption devices for the masses. Their touch screens provide an *absolute direct input* that is very convenient for content interaction. If we look at the front of these devices we can find lots of input methods and interaction areas such as cameras, touch screens, pressure sensors, buttons, fingerprint scanners... However, if we take a look at their back we find a big black surface that is not being used for interaction and has a big potential.

HaptiCase wants to take advantage of the unused area on the back of mobile devices for interaction. Only tested with a 4-inch device.

HaptiCase from Corsten et al. [2015] wants to change this. Let's consider the scenario where a user is mirroring her apps and games to a bigger screen. Here the user's attention is on the *output device* (the distant screen), rather than on the *input device* (the smartphone or tablet). The idea of this project is to put tactile landmarks on the back of the device that compensate this lack of visual feedforward with tactile feedback. Using *HaptiCase* users improved their performance in absolute indirect pointing tasks. Hueber [2015] continued this work investigating how visual information is mapped in absolute indirect touch tasks and used the results to provide design guidelines to implement back-of-device tactile landmarks. However, *HaptiCase* research has only been made with 4-inch devices, so it is unknown if it also improves eyes-free tapping for other device sizes.

Definition: *Feedforward*

FEEDFORWARD:
Feedforward, Behavior and Cognitive Science is a method of teaching and learning that illustrates or indicates a desired future behavior or path to a goal.[1] Feedforward provides information, images, etc. exclusively about what one could do right in the future, often in contrast to what one has done in the past.

Scaling the *HaptiCase* designs means changing the *size* of the interaction area as well as the *resolution* of the tactile landmarks.

To answer this question one could scale the designs of Hueber [2015] to work with bigger devices and then measure their performance by absolute indirect pointing tasks using devices of different sizes. However, this approach would not only affect the *size* of the devices, but also the *resolution* of the tactile landmarks of the *HaptiCase* models as well. Having these two variables changed, the results from this study would be inconclusive, since there is no way of knowing which one of them was responsible for the alterations in the result.

This thesis focuses on the scaling of the interaction area, hence we got rid of the tactile landmarks.

In this thesis we present a scalable measurement tool that captures back-of-device touch data. This tool will only focus on how the scaling of a back-of-device touch input surface affects the user's performance, hence we will get rid of the tactile landmarks concept from Corsten et al. [2015] in all of our prototypes. Doing this, we assure a direct link between the changes in the *size* of the interaction area and the results of the study. We will also not use smartphones or tablets as input devices, but prototypes built from scratch based on the *Frustrated Total Internal Reflection (FTIR)* technology. Not using existing devices, we are able to completely control variables such as the *size*, the *weight*, the *width* and the *form factor*, minimizing their influence in the results.

We will use *reacTIVision* from Kaltenbrunner [2009] to harvest information of the touches. This image recognition software is capable of analyzing a video stream looking for finger blobs and determine their *position*, *angle*, *width* and *height*. An example of a back-of-device interaction with some relevant finger information is given in Figure 1.1.

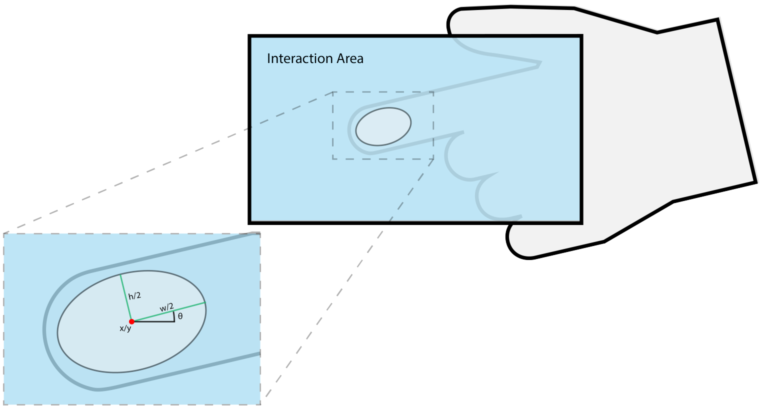

Figure 1.1: Description of a back-of-device interaction with relevant finger information harvested by *reacTIVision*, with x/y being the relative position of the blob, θ its angle, w its width and h its height.

FRUSTRATED TOTAL INTERNAL REFLECTION (FTIR): When light crosses a material-air interface beyond a certain critical angle, it will be totally internally reflected (TIR). However, if another material is in optical contact at this interface, it can frustrate this TIR, and causes some light to scatter out of the surface (Han [2008]).

Definition:
Frustrated total internal reflection (FTIR)

This tool and these prototypes are easily scalable and can be used in future research to determine how *HaptiCase* performs with different device sizes.

Chapter 2

Related work

For the developing of the software and hardware that build our tool we needed to consider different research fields:

On the hardware side our project is based on the *HaptiCase* project, we are building prototypes using FTIR technology, scaling these prototypes to provide interaction areas of different sizes and using a back-of-device type of interaction.

On the software side, our tool is composed of a *Server* and a *Client*. The *Server* is a computer vision toolkit for multi-touch interaction called *reacTIVision*. The *Client* is a program we wrote to organize the data harvested by the *Server*.

2.1 HaptiCase

HaptiCase from Corsten et al. [2015] presents the scenario of a user interacting with a distant screen using a smartphone as input device without breaking eye contact with the output monitor. As the user's attention is on the distant screen and not on the input device itself, *HaptiCase* aims to compensate this lack of visual feedforward with tactile feedback on the back of the device. To achieve this some tactile landmarks are used on the back of the smartphone, so that the user can sense with her fingers the desired posi-

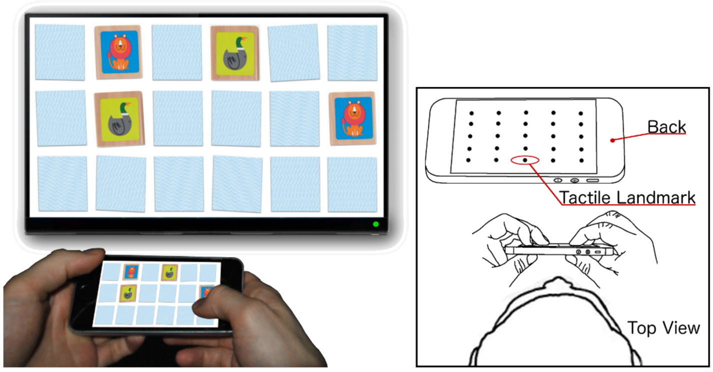

Figure 2.1: *HaptiCase* concept. Image taken from Corsten et al. [2015].

tion on the back and then transfer the touch to the front by pinching the finger at the back to the thumb resting above the touch screen at the front. Figure 2.1 describes this interaction.

HaptiCase does improve target acquisition in this particular scenario, but as stated in Section 1, only 4-$inch$ smartphones were tested, so it is unknown how it performs with other device sizes. To answer this question our tool may be used for future research on the matter.

2.2 FTIR

FTIR makes light ”trapped” inside of an acrylic plate get scattered when another material gets in contact with the surface. This material could be a finger, which would create a blob of light at its exact same position on the reverse side of the acrylic. Using a camera and a simple image recognition software, one can analyze the touches data. This principle is illustrated in Figure 2.2.

Due to the low cost of building a touch recognition system with FTIR, this technology is widely used for building

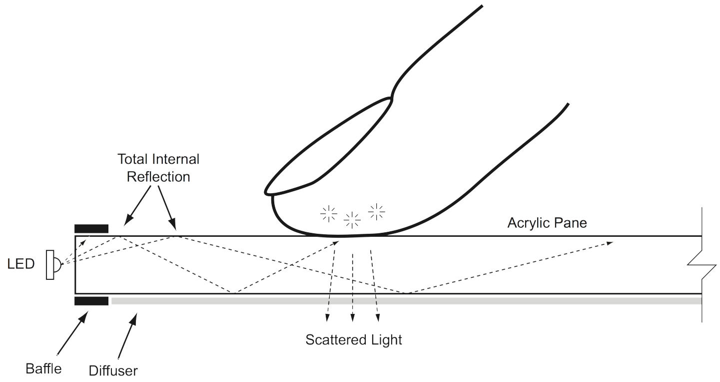

Figure 2.2: Example of FTIR system. Image taken from from Han [2008].

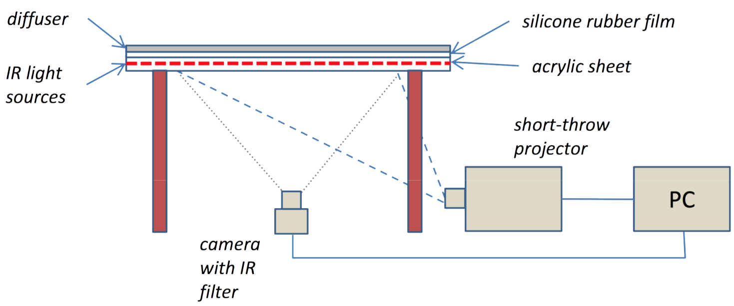

Figure 2.3: Components of FTIR multi-touch table. Image taken from Wolfe et al. [2008].

cheap and efficient research prototypes.

Wolfe et al. [2008] adopted FTIR to build *EquisFTIR*, a multi-touch table for gaming. They used a normal table with an acrylic surface illuminated sideways by IR light. Under the table a projector displays the game image on the surface and a camera with an IR filter detects the IR blobs created when the user touches the surface. Figure 2.3 shows all the table components. This approach is very common when building multi-touch tables based on FTIR and will inspire the construction of our prototypes.

FTIR was used to develop multi-touch tables. This concept inspired the construction of our prototypes.

2.3 Scaling

In Gilliot et al. [2014] they investigated how the size of an input device affects the minimum target size a user can acquire. They performed an experiment using two input devices of different screen sizes: an iPad 1 ($196 \times 147\ mm$) and an iPod Touch 3 ($66 \times 50\ mm$) that was modified to provide the same $4:3$ aspect ratio as the iPad.

The users had to look at a distant screen that showed them some targets in a window with a $4:3$ aspect ratio. Then they had to touch on a blank touch screen (the targets were not shown there) where they thought the target would be located if the image from the distant screen was mapped to the smaller one. The experiment was performed as shown in Figure 2.4 in different stages: using one hand, using one hand and blinder glasses and using two hands and blinder glasses.

Targeting error increases with bigger interaction areas. Similar aspect ratio mappings are beneficial.

Their findings concluded that the targeting error increases as the interaction area gets bigger and that similar input and output aspect ratios (between the distant screen and the interaction area) led to better user performance.

Following this guidelines we designed all of our prototypes with the same aspect ratio ($16:9$) and made our software adapt the area where the targets are shown to always have an aspect ratio of $16:9$ independently of the screen being used.

2.4 Back-of-device Interaction

Wobbrock et al. [2008] studied how humans perform when interacting with the back of a device. They executed a study to examine interaction with thumb and index finger on the front and on the back of a device, using one and two hands to hold it.

Index finger performs well on the back of a device. Holding the device with two hands is beneficial.

Figure 2.5 shows the touchpad they used in their experiment as well as all the input combinations tested. One

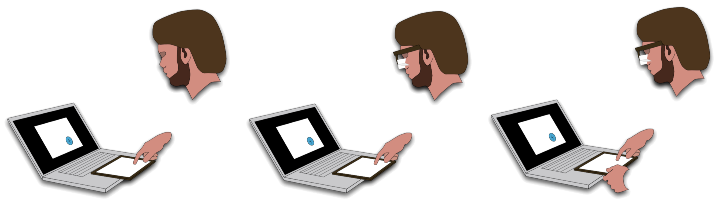

Figure 2.4: Experiment performed by Gilliot et al. [2014] where users had to acquire targets shown in a distant screen by touching a touch screen. On the left: experiment executed with one hand and without blinder glasses. On the center: experiment executed by one hand and blinder glasses. On the right: experiment executed by two hands and blinder glasses. Image taken from Gilliot et al. [2014].

of their findings pointed out, that the index finger performs surprisingly well on the back of a device, even outperforming the thumb on the front when executing horizontal movements. They also found out, that people were more accurate at performing horizontal and vertical gestures while holding the device with two hands as opposed to holding it only with one.

Our prototypes are designed to be used with both hands.

According to the findings of Wobbrock et al. [2008] and in concordance with the *HaptiCase* concept, we designed our prototypes to be used with two hands in landscape mode, touching the backside surface with either the *index*, the *middle*, the *ring* or the *little finger*. We expect users to use the *index finger* as their preferred finger for interaction in our future experiments.

Wolf et al. [2014] also explored how people perform when interacting with the back of a device. They performed an experiment where the users had to hold a 12.6-*inch* tablet in landscape mode using both hands like Figure 2.6 shows. The task consisted in drawing with each finger a line as long as possible that started in the position where the finger was resting and tried to reach the center of the device. The participants solved this task with the thumb in the front of the device and with the rest of the fingers in the back of the device.

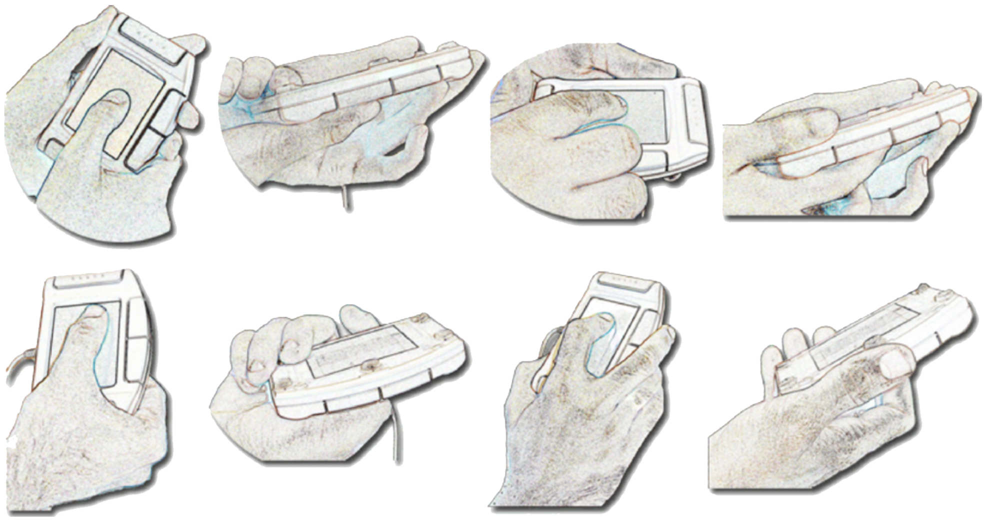

Figure 2.5: The eight postures tested in the study of Wobbrock et al. [2008]. *Top Row*: two-handed postures. *Bottom Row*: One-handed postures. *Left to Right*: thumb-on-front, thumb-on-back, index-on-front, index-on-back. Image taken from Wobbrock et al. [2008].

Figure 2.6: Tablet used in the experiments of Wolf et al. [2014]. Image taken from Wolf et al. [2014].

Very outer vertical positions are hard to access.

Their findings concluded that center areas that are further away from the edges than the fingers length are out of reach for touch input. They also found out that outer vertical regions are accessible, but very outer vertical positions are hard to access, since pointing with fingers is more ergonomic if the finger is neither fully flexed nor completely stretched.

Our software leaves a margin on the sides where no targets are shown for interaction.

For this reason, we designed our software to leave a margin on the sides of the screen where no targets can show up. A more detailed explanation about this is given in Section 3.2. Since our biggest prototype has an interaction area of 7 $inches$, we do not expect the center areas to be unreachable in future experiments.

2.5 Software

As stated before, the software used to work with the prototypes is composed of a *Server*, which is used to analyze the camera image information to detect the position and orientation of the touches, and a *Client* that harvests and arranges the information provided by the server.

The *Server* we use is *reacTIVision* from Kaltenbrunner [2009], a computer vision toolkit for fiducial and finger tracking. This software allows the rapid development of cheap multi-touch tabletop user interfaces, like *EquisFTIR*. This application uses the TUIO protocol to deliver UDP packets of information over a local network, which are then received by our *Client*.

FIDUCIAL:
A fiducial marker or fiducial is an object placed in the field of view of an imaging system which appears in the image produced, for use as a point of reference or a measure. It may be either something placed into or on the imaging subject, or a mark or set of marks in the reticle of an optical instrument.

Definition: *Fiducial*

The average finger size can be adjusted in the program for optimal results, and the blob contour information of the finger gets encoded in a compact format that describes its approximate *position*, *size*, *orientation*, *velocity* and *acceleration* (the last two of them are not relevant for our research). Figure 2.7 shows how the information of a blob is processed. The message syntax sent by the server looks as follows:

The server provides information such as the position, size and orientation of the finger.

$/tuio/2Dblb$ set sid xpos ypos angle width height area xvel yvel rvel macc racc

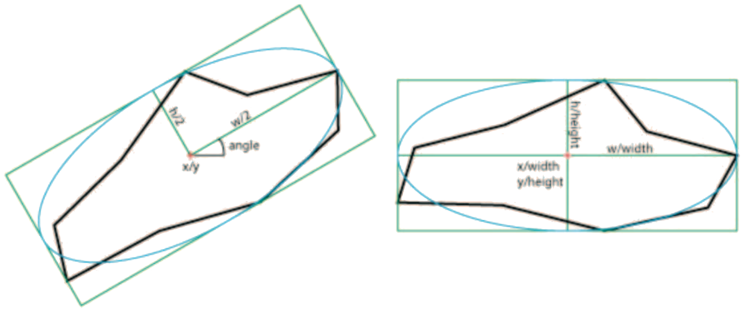

Figure 2.7: Simple description of a blob enclosure in *reacTIVision*. Image taken from Kaltenbrunner [2009].

Where *sid* is a unique ID for each finger, *xpos & ypos* describe the position of the finger, *xvel*, *yvel & rvel* its velocity and *macc & racc* its acceleration. These messages are received and stored by the *Client* for later analysis.

Chapter 3

Design & Fabrication

To be able to test our *reacTIVision Client* as we developed it, we first built an evaluation prototype. We imitated the construction of a multi-touch table, scaling it to a smaller form factor and inverting the design to allow a back-of-device interaction. We used this first prototype to test our client software as we were developing it.

We learned from this first prototype and built then our final prototypes upgrading the design of the first one to achieve lighter models with a smaller ambient light influence.

3.1 First Prototype

As an acrylic plate compatible with FTIR we used a $9.26 \times 4.98 \times 0.8$ cm *Plexiglas* plate. To construct the prototype, we needed 3 different parts:

- A case to encapsulate the plate and surrounds it with IR (infrared) LEDs. We will refer to this part as *part A*.
- A case to encapsulate the camera. We will refer to this part as *part B*.
- Two pieces to connect *part A* to *part B*. We will refer to each of these pieces as *part C*.

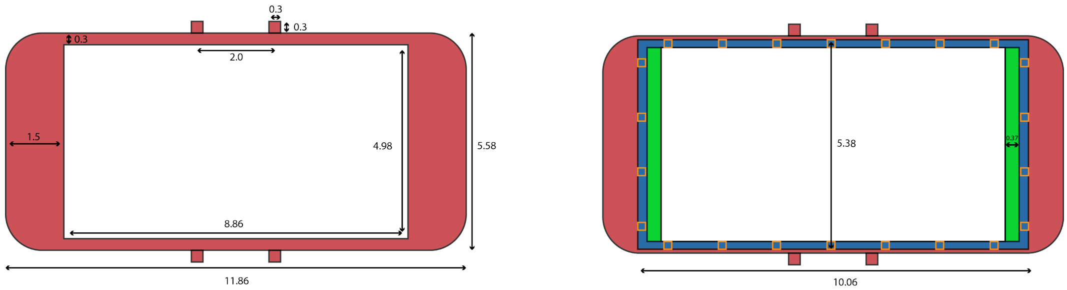

Figure 3.1: Design of *part A* for the first prototype. The unit used is $[cm]$. The size of the interacting area is $8.86 \times 4.98\ cm$, imitating a 4-inch smartphone display. A small margin of $0.3\ cm$ is left on the upper and the lower part to give room to the $0.2 \times 0.2 \times 0.8\ cm$ LEDs (orange squares). A bigger margin of $1.5\ cm$ is left on the sides for the user to be able to grip the device. On the left: Front of the prototype, where the camera will be recording. On the right: Back of the prototype, where the interaction will take place. A recess of $9.66 \times 5.38 \times 0.8\ cm$ (blue and green region) is left for the LEDs, which will be covered afterwards to keep an interaction area of $8.86 \times 4.98\ cm$ and to give the user tactile feedback of the borders. The green area shows the regions of the recess where the plate will be glued afterwards. The prototype is $0.9\ cm$ thick.

Figure 3.1 describes our design for *part A* in detail. The case has four $0.3 \times 0.3 \times 0.3\ cm$ cubes (two on the upper and two on the lower side) that will serve to attach two *part Cs* to it. The plate will get glued to the green area to fix it to the part. The LEDs will be facing the plate and will get covered afterwards with a border made of a thin cardboard.

For this test prototype, we used a *FOculus IEEE1394 Digital Camera* for image acquisition. The *Part B* described in Figure 3.2 was designed accordingly to work with the dimensions of this camera. It has a recess (blue region) in the middle to hold the camera firmly and a hole for the camera lens to go through it.

Part A and part B need to move together for the software to work. Part C joins them.

Both *part A* and *part B* need to be connected, so that when the user moves the plate the camera moves along with it, keeping a stable image stream for the software to analyze. To do this we designed *part C* as seen in Figure 3.3. The square holes will get the part attached to a side of *part A* and the circular holes will let a screw fix the part to one side of *part B*. Since we did not know which was the optimal distance between the camera lens and the plate to get a

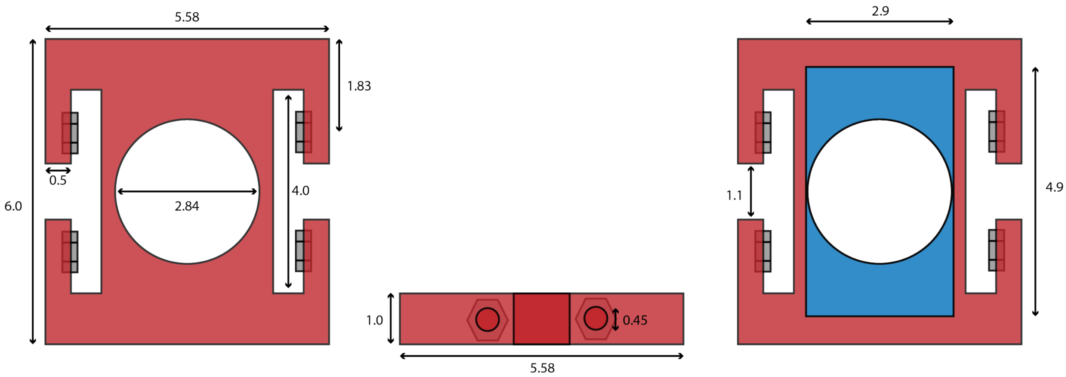

Figure 3.2: Design of *part B* for the first prototype. The unit used is $[cm]$. A circular hole of $2.84\ cm$ in the middle lets the camera lens go through the case. On the left: Back of the case, part that will be facing the plate. On the middle: the case as observed from the right. Four holes with a diameter of $0.4\ cm$ will be used to connect this part to *part C*. A $M4$ nut (grey color) is glued to the inner side of each hole to be able to fix the part with screws later. On the right: Front side of the case, where the camera will be located. A recess of $2.0 \times 4.9 \times 0.5\ cm$ will hold the camera in place (blue color).

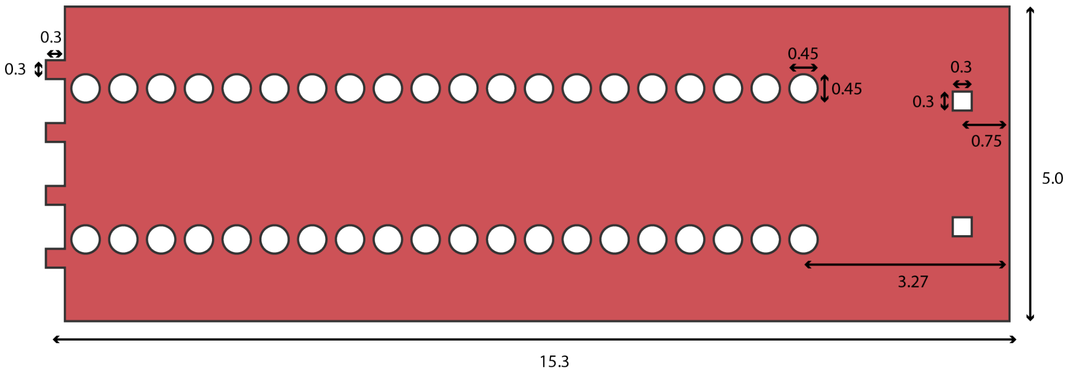

Figure 3.3: Design of *part C* for the first prototype. The unit used is $[cm]$. Round holes connect to *part B*, square holes connect to *part A*.

good image, we designed different levels of rounded holes. The square bumps on the top of the part were meant to connect to a fourth piece that should have joined the two *part Cs* in case the prototype was not stable enough. Since the screws provided a steady balance, this part was never designed and the square bumps became obsolete.

We created a 3D version of *part A* and *part B* with the design

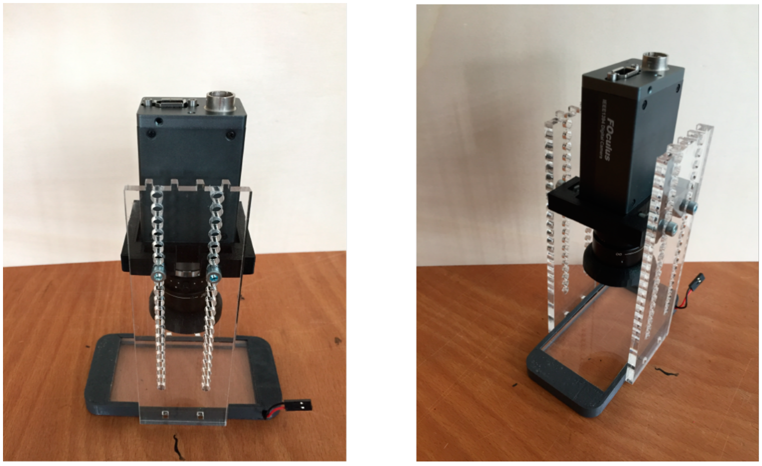

Figure 3.4: First working prototype using *Plexiglas* for *Part C*, back *PLA* filament for *part A* and textitpart B and a *FOculus IEEE1394 Digital Camera*.

software *Vectorworks* and a 2D version of *part C* with *Adobe Illustrator*. We printed the first two parts with a *Dremel 3D20 Idea Builder* using black *PLA* filament. *Part C* was created by cutting a $5.0\ mm$ *Plexiglas* plate with a *Cameo Zing 6030* using the Visicut software. The settings used are stated in Table 3.1.

	Power	**Speed**	**Frequency**
Cut	95	22	5000

Table 3.1: Settings used in Visicut to cut a $5.0\ mm$ thick *Plexiglas part C*.

The parts were mounted as shown in Figure 3.4, creating our first working prototype. We used a fisheye lens to encrease the angle of vision, which reduced the necessary distance between the camera and the plate to get the whole plate in frame. An IR filter was attached to the camera lens so that only IR light gets captured.

The first prototype was too heavy for a user study.

This prototype works great for testing, but it would be very heavy for a user study, since it weights $317\ g$. For our final prototypes we worked on reducing their weight to make them feel as close as possible to a regular smartphone.

3.2 Client Software

On the software side, the *reacTIVision Server* sends the information about the touches in UDP packets over a socket in the local network using the *TUIO Protocol*. We wrote a Mac OS application using the programming language Swift, which works as a client and is capable of receiving these UDP packets using the *TUIO framework*.

Our client is composed of two windows: the *Control Panel Window* and the *Preview Window*.

3.2.1 Control Panel Window

The *Control Panel Window* is where the tester can manage the client options. Here the tester can specify a user ID for the person that is performing the experiment, her age and the size of the device that she is going to use before starting the test.

The client saves the received information in two CSV archives. The first CSV (CSV1) works as an automatic backup that saves all the data sent from the server as soon as it is received. The second CSV (CSV2) saves only touch data that the tester wants to be saved. For saving a touch information, the tester has to specify which finger was used by the person performing the test. This data is then also added to the CSV2. Every update of the CSV archives is notified in two text fields placed on the right of the window.

The client does an automatic backup (CSV1) and a manual backup (CSV2).

The *Control Panel Window* is shown in Figure 3.5.

3.2.2 Preview Window

The *Preview Window* is composed of a white region in the middle that represents the interaction area, delimited by two black frames on the upper and lower side. These black frames ensure that the white region always has an aspect

The Preview Window always has an aspect ratio of 16:9.

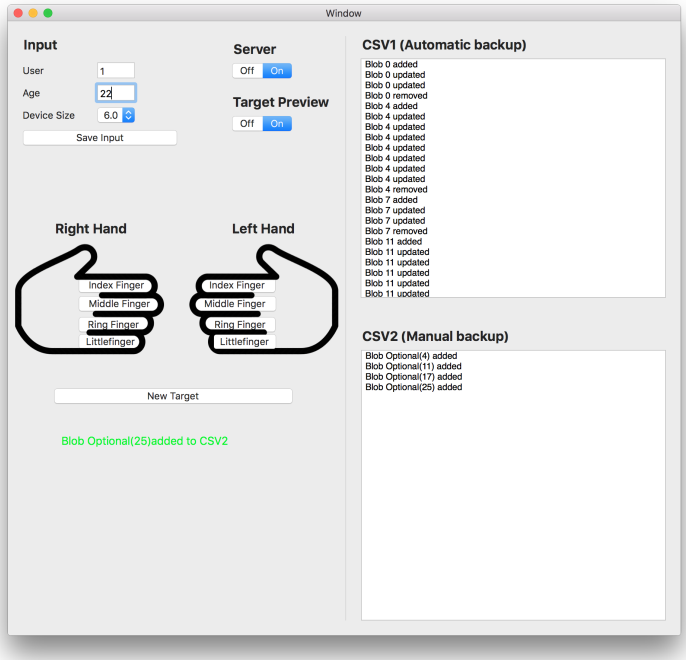

Figure 3.5: *Control Panel Window*. On the upper left side: Input options (user ID, Age, Device Size) and toggles to activate the server to start receiving data and to preview the touch target in the *Preview Window*. On the lower left side: buttons to specify the finger used to be saved in CSV2 and button to show a new target in the *Preview Window*. On the right side: text fields notifying every update performed in the two CSV archives.

ratio of $16:9$ independently of the window size and hence independently of the monitor size the window is going to be displayed on.

This window shows the position of the target that has to be touched by the user as a blue blob, and, if the option was activated in the *Control Panel Window*, a live update of the finger of the user as a red blob.

The targets are shown in a smaller region of the screen that leaves a margin of 12.5% in all sides. We left these margins to avoid targets being shown too close to the screen

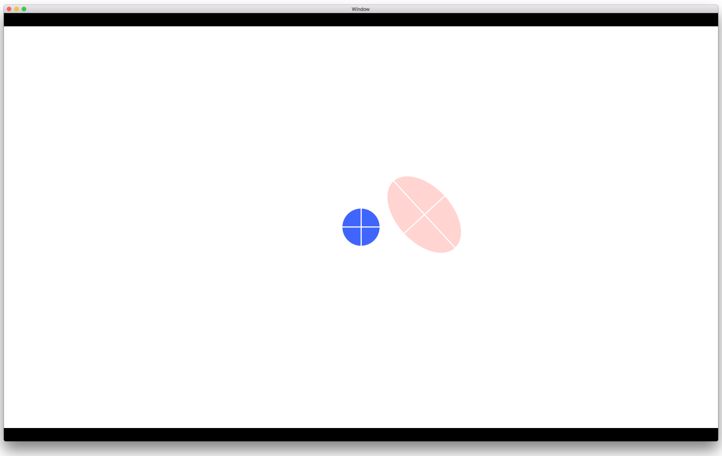

Figure 3.6: *Preview Window*. Here the blue blob shows the target where the user has to tap. The red blob is a live preview of where the user is touching. The red blob can be turned off in the *Control Panel Window*.

borders to prevent users from performing unnatural finger positionings, since it is hard to access areas with the finger completely flexed, as Wolf et al. [2014] concluded. This smaller region was divided in an imaginary 5×5 grid numbered from top to bottom and left to right. Figure 3.5 shows the grid with the target positions. The selected targets are $(1, 3, 9, 10, 11, 13, 17, 19, 22, 25)$, the same as in Gilliot et al. [2014], except for number 17, that has been added to obtain an equal number of targets on the left hand side and on the right hand side. Only one target is shown at a time and their position array is permuted by our software before every experiment.

We designed our application to be scalable, so the grid size and the target position array can be adjusted by changing the variables *targetPositionsArray* and *GridSize* in our software.

The grid size and target position arrays may be adjusted.

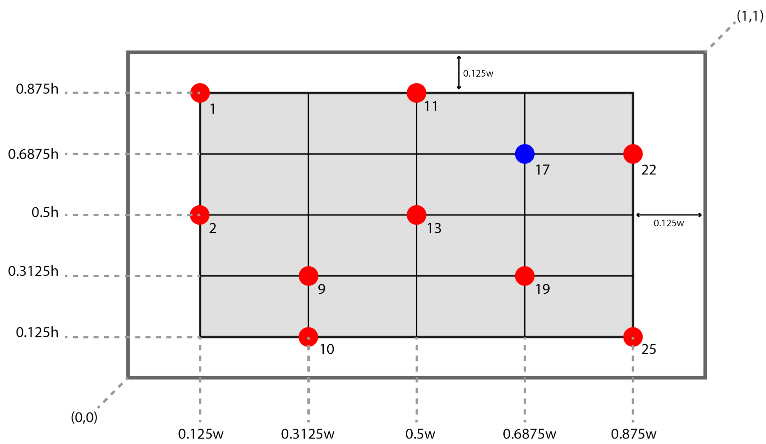

Figure 3.7: Area where the targets are shown in the *Preview Window*. Targets are the same as in Gilliot et al. [2014], except for the blue target, that has been added to obtain an equal number of targets on the left hand side and on the right hand side. Targets are enumerated from top to bottom, left to right on a 5×5 imaginaty grid: The Target in the top left corner is Target 1, the target in the lower right corner is Target 25.

3.3 Final Prototypes (Design)

All of our designs are for prototypes with interaction areas with an aspect ratio of 16:9.

We designed four prototypes that provided interaction areas with diagonals of $4, 5, 6$ and 7 inches with a form factor of $16 : 9$, imitating popular sizes of screens of mobile devices. We will refer to these prototypes as *Prototype A*, *Prototype B*, *Prototype C* and *Prototype D* respectively. Table 3.2 shows the width and height values for the interaction areas of the different prototypes both in $[inches]$ and $[cm]$.

	Diagonal [Inches]	Width [inch]	Height [inch]	Width [cm]	Height [cm]
Prototype A	4.0	3.49	1.96	8.86	4.98
Prototype B	5.0	4.36	2.45	11.07	6.23
Prototype C	6.0	5.23	2.94	13.28	7.47
Prototype D	7.0	6.10	3.43	15.50	8.72

Table 3.2: Width and Height of the interaction areas of all the prototypes in $[inches]$ and $[cm]$.

Using the values of Table 3.2 we designed our prototypes.

We imitated the design of the first prototype and scaled it for the bigger form factors.

After testing the first prototype, we set ourselves two main goals for the development of our final models: to reduce its weight and to decrease the influence of ambient light.

Goals for the final prototypes: reduce weight and ambient light influence.

To reduce the IR ambient light influence in the results provided by our prototypes, we decided to tweak the design to create a dark room between the camera and the plate, where no light comes inside. To do so we had to perform some changes in *part B* and *part C* and introduce a new part that acts as a wall. We will refer to this new piece as *part D*.

The new designs for the different parts try to achieve these goals. The final designs will be discussed in the following sections.

3.3.1 Part A (Design)

For *part A* we stuck with the previous design of the first prototype, with the desired interaction area in the middle, a margin of $1.5\ cm$ on the left and right sides and a margin of $0.3\ cm$ on the upper and lower side. Figure 3.8 describes the final designs for *part A* for the different sizes.

We designed the sizes of the *Plexiglas* plates for the different prototypes according to the values shown in Table 3.2, but we added $0.2\ cm$ on the left and on the right side so that it could be glued to the green areas showed in Figure 3.8. We used a $0.8\ cm$ thick *Plexiglas*. The dimensions of the final plates are stated in Table 3.3.

To cover the recess formed to store the LEDs, prevent the light from scattering outside of the part and to give the user tactile feedback of the limits of the interaction area we designed a border for each prototype size that will be glued to the back of each *part A*. Figure 3.9 shows the designs in detail.

Border provides tactile feedback and traps the light inside of the prototype.

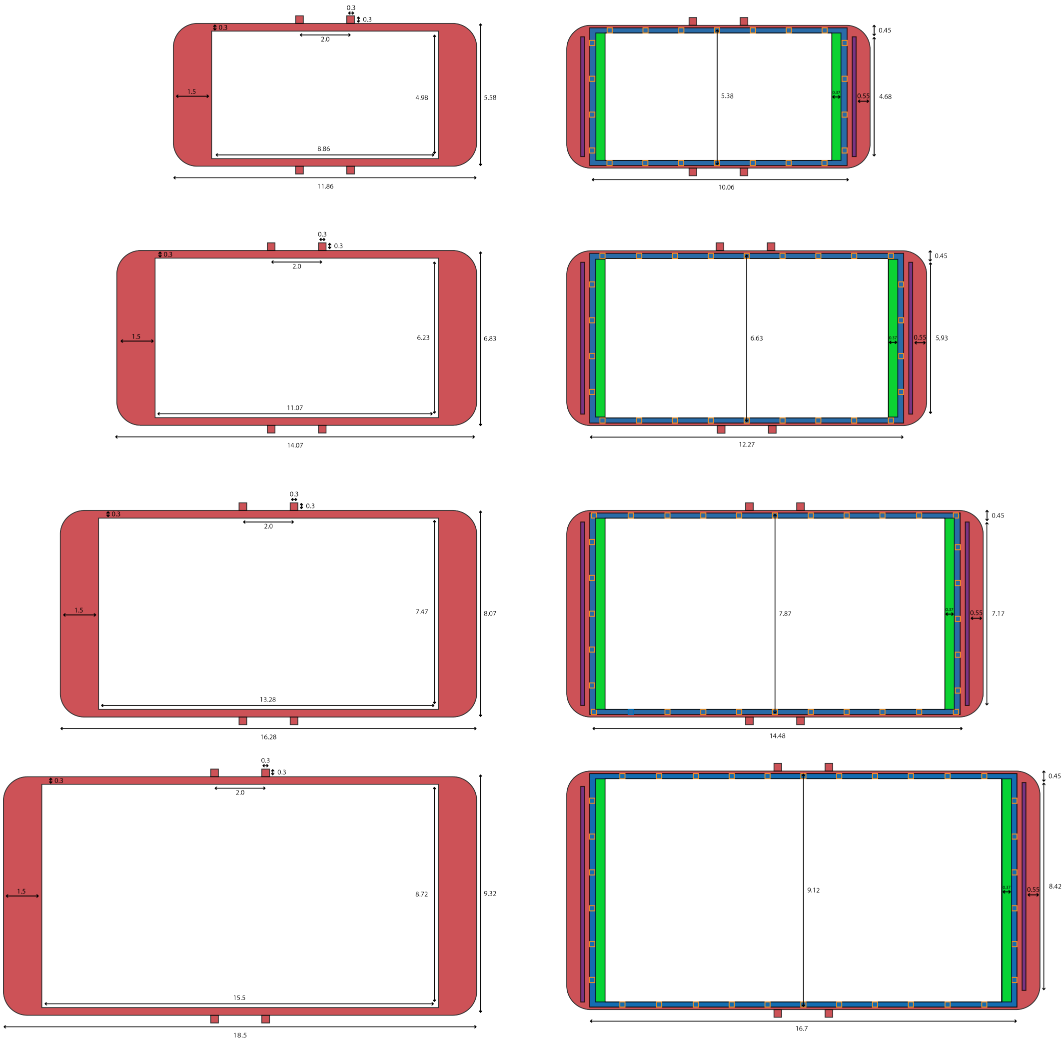

Figure 3.8: Design of *part A* for the last prototypes. The unit used is $[cm]$. A small margin of $0.3\ cm$ is left on the upper and the lower side to give room to the $0.2 \times 0.2 \times 0.8\ cm$ LEDs (orange squares). A bigger margin of $1.5\ cm$ is left on the sides for the user to be able to grip the device. Four cubes on the upper and lower side will connect the part with *part C*. The blue and green surface indicate a recess of $0.8\ cm$. From the first to the fourth row: Design for *Prototype A, B, C and D*. On the left column: Front of the prototype, where the camera will be recording. On the right column: Back of the prototype, where the interaction will take place. The LEDs will be covered afterwards with borders. The purple area shows a protrusion of $0.05\ cm$ that will serve as a guide to glue the borders. The protrusion has a width of $0.15\ cm$. The green area shows the regions of the recess where the *Plexiglas* plate will be glued. All designs are $0.9\ cm$ thick.

	Diagonal [inches]	**Width [cm]**	**Height [cm]**
Prototype A	4.0	9.26	4.98
Prototype B	5.0	11.47	6.23
Prototype C	6.0	13.68	7.47
Prototype D	7.0	15.90	8.72

Table 3.3: Width and Height of the used *Plexiglas* plates in $[cm]$.

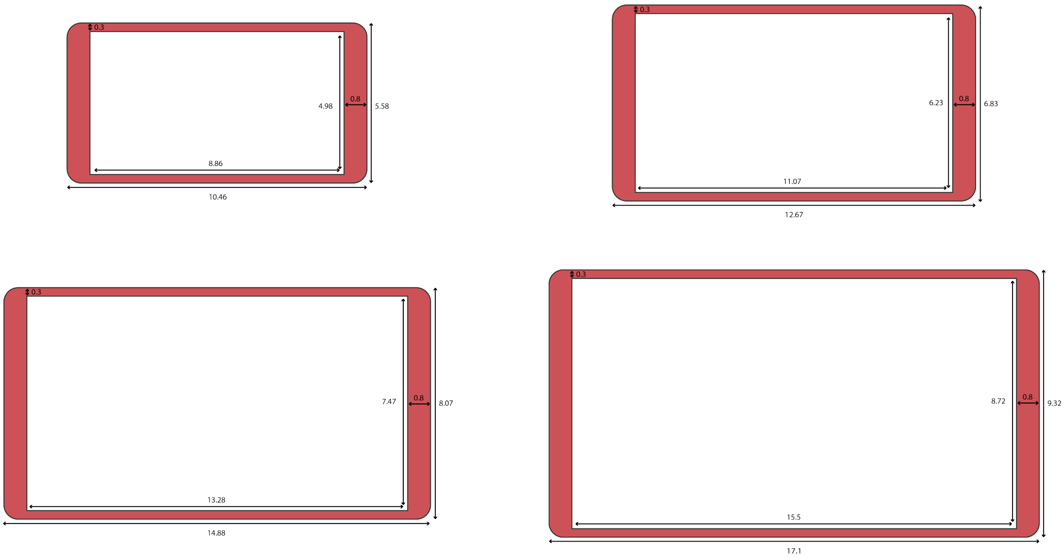

Figure 3.9: Design of the borders for *part A*. The unit used is $[cm]$. The black lines of the design were cut with a *Cameo Zing 6030* using the settings stated in Table 3.4. On the first row and from left to right: border for *Prototype A* and for *Prototype B*. On the second row and from left to right: border for *Prototype C* and for *Prototype D*.

3.3.2 Part B (Design)

To reduce weight a lighter camera was used.

One of the main factors that increased the weight of the first prototype was the *FOculus IEEE1394 Digital Camera*. We needed a lighter camera that provides a wide angle of vision without distorting the image (the previous fisheye lens twisted the image frame perverting the measurements). We chose the camera *USB500W05G-FD100* from *ELP*, a 5MP USB 2.0 camera with a 100-degree lens that does not alter the image. The camera weights only 50 g and has a board size of only 38×38 mm.

Part B was tweaked to work with the new camera and unnecessary holes were removed to reduce the ambient light influence.

Since we changed the camera, the previous design of *part B* had to be adjusted. We tweaked *part B* to work with the new smaller camera and we also removed all unnecessary holes so that no light gets through it. The final designs of *part B* for all the prototypes are described in detail in Figure 3.10. All remaining holes showed in the image will be covered afterwards: the big squared hole in the middle will be covered by the camera and the four rounded holes will be covered by screws. We also added a region on the upper and lower side of the part, where it will connect to *part D*.

3.3.3 Part C (Design)

To reduce weight, we also tested other materials for *part C*. The previous 5.0 mm *Plexiglas* weighted 40 g. We manufactured the part with 3.0 mm and 5.0 mm poplar wood. The first plate weighted 8.0 g and the second one 16.0 g. Since the 5.0 mm thick part was more robust than the other one, and it already provided a significant weight improvement, we chose this one for our final prototypes.

To reduce ambient light influence, unnecessary holes were removed from part C.

We tested the new camera with the old prototype and determined the optimal distance between the camera and the acrylic plate for each size, so that a clear image of the whole plate stayed in frame. Using this information, we removed all the additional holes from the previous design and left only the holes that were going to be used to attach the part to *part B*, which helped to reduce the ambient IR light influence. The construction details of each individual *part C* for our final prototypes are described in Figure 3.11.

3.3.4 Part D (Design)

Part D creates a dark room between the camera and the acrylic plate, what reduces ambient light influence.

The walls should cover the open space left between *part A* and *part B* that can be observed in Figure 3.4 and create a dark room between them. The design of *part D* is composed of a body and two arms positioned in a T-shape. We added

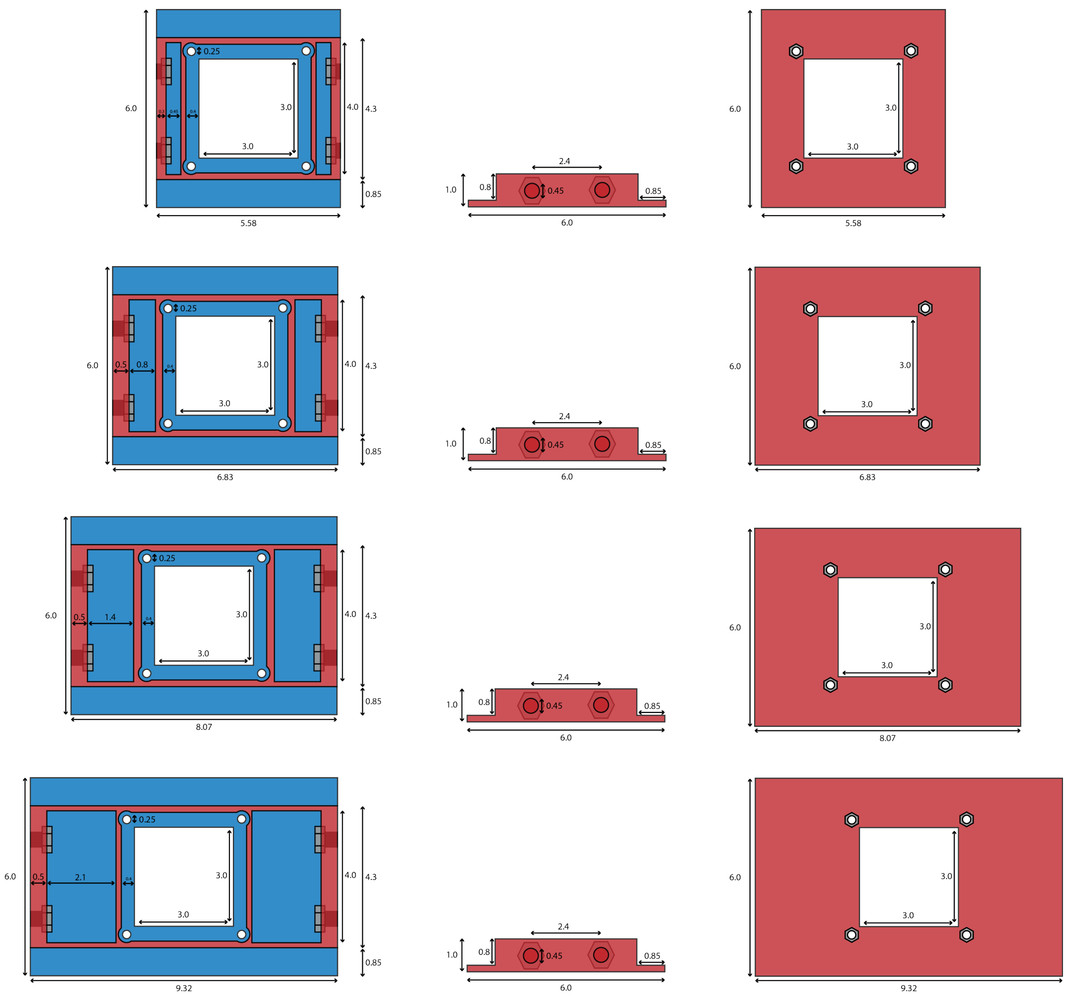

Figure 3.10: Design of *part B* for the last prototypes. The unit used is $[cm]$. From the first to the fourth row: Design for *Prototype A, B, C and D*. The square hole in the middle will let the camera go though the part. On the left column: Front of the part that will be facing the user. The blue surface indicates a recess of $0.8\ cm$. The four rounded holes with a $0.25\ cm$ diameter will let $M2$ screws fix the camera to the part. The upper and lower side will connect the part to *part D*. On the middle column: Part as seen from the right. The rounded holes with a $0.45\ cm$ diameter will connect the part to *part C*. A $M4$ nut is glued to the inner side of each hole to be able to fix the part with screws later. On the right column: Back of the part that will be facing the plate. Four $M2$ nuts are glued around the holes to fix the camera properly.

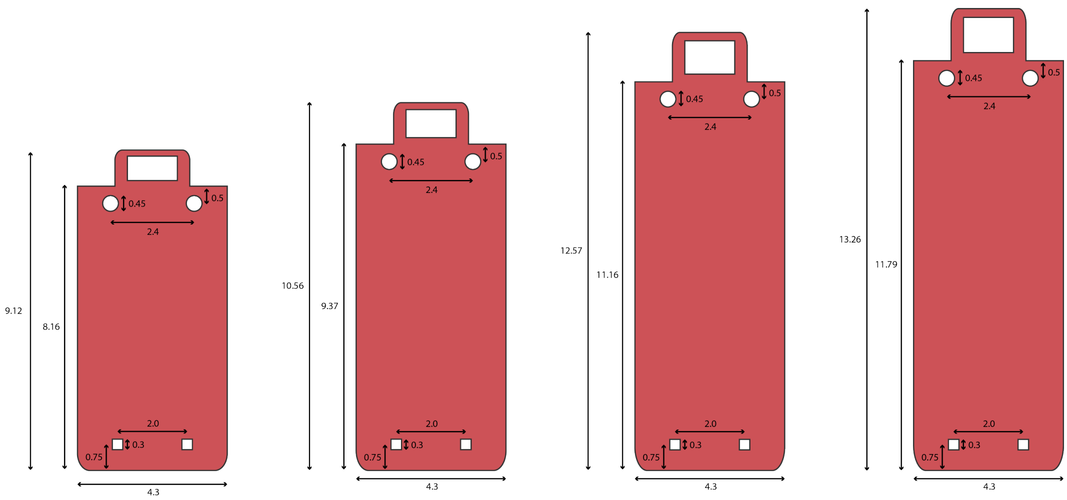

Figure 3.11: Design of *part C* for the last prototypes. The unit used is $[cm]$. Round holes connect to *part B*, square holes connect to *part A*. The camera USB cable will go through the rectangular hole on the upper side. From left to right: Design for *Prototype A, B, C and D*.

a $0.3\ cm$ border in the regions where the piece should connect with other parts to glue them together. A more detailed description of the design in given in Figure 3.12.

3.4 Final Prototypes (Fabrication)

3.4.1 Part A (Fabrication)

We printed the designs from Figure 3.8 for *part A* with a *Stratasys Dimension Elite* using black *ABS*, which created more stable parts than the previously used *Dremel 3D20 Idea Builder* using *PLA* filament.

The borders of Figure 3.9 were glued to the back of *part A*. They were cut on a $1.0\ mm$ thick cardboard with a *Cameo Zing 6030* using the Visicut software. The settings used in the Visicut software are stated in Table 3.4.

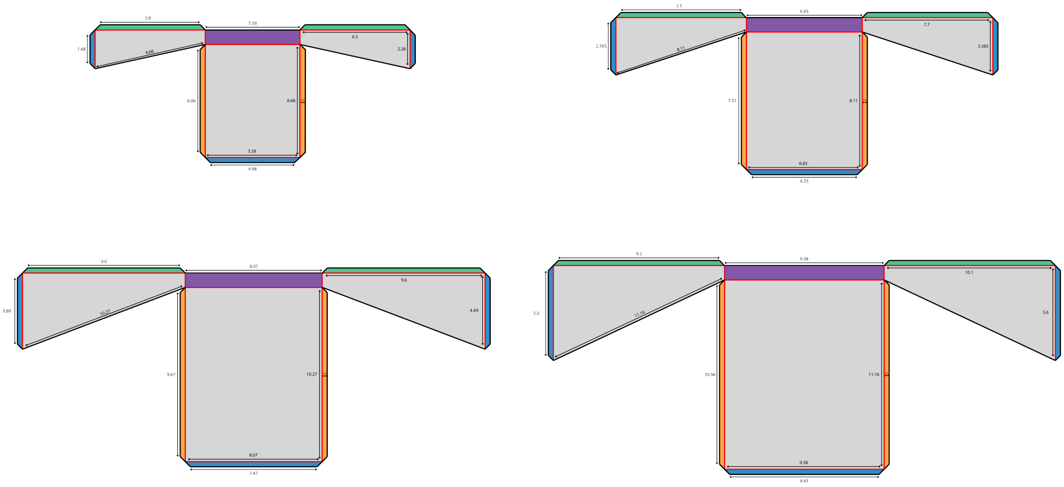

Figure 3.12: Design of *part D* for the last prototypes. The unit used is $[cm]$. The piece is composed of a body and two arms. The black lines were cut. The red lines were marked to fold the material at these positions. Blue regions connect to *part A*. The purple region connects to *part B*. Green regions connect to *part C*. Orange regions are glued to the inner side of the arms. On the first line and from left to right: Design for *Prototype A* and *Prototype B*. On the second line and from left to right: Design for *Prototype C* and *Prototype D*.

	Power	**Speed**	**Frequency**
Cut	35	100	2500

Table 3.4: Settings used in Visicut to cut a $1.0\ mm$ thick Cardbord for the borders of *part A*.

3.4.2 Part B (Fabrication)

We printed designs from Figure 3.10 with a *Stratasys Dimension Elite* using black *ABS*, like we did with *part A*.

3.4.3 Part C (Fabrication)

We cut the designs of Figure 3.11 for *part C* on a $5.0\ mm$ *Poplar Wood* plate with a *Cameo Zing 6030* using the Visicut software. The settings used in the Visicut software are stated in Table 3.5.

	Power	Speed	Frequency
Cut	100	80	500

Table 3.5: Settings used in Visicut to cut a $5.0\ mm$ thick *Poplar Wood* plate for *part C*.

3.4.4 Part D (Fabrication)

We cut and marked the designs of Figure 3.12 for *part D* on a $0.4\ mm$ thick black cardboard with a *Cameo Zing 6030* using the Visicut software. The settings used are stated in Table 3.6.

	Power	Speed	Frequency
Cut	20	100	5000
Mark	5	100	5000

Table 3.6: Settings used in Visicut to cut a $0.4\ mm$ thick Cardbord for *part D*.

3.5 Assembly

After all parts were printed/cut we assembled them together creating the final prototypes, which are shown in Figure 3.13. We attached an IR filter to the camera lens so that only IR light gets captured. We inserted the cubes on the upper and lower side from *Part A* in the holes on the lower side of *part C*, connecting the two parts. We screwed *part C* to *part B* and we glued *part D* to all the other parts as described in Figure 3.12.

More detailed pictures from different angles of *Prototype D* are found in 3.14. The achieved dark room can be observed in the first picture from the last row of Figure 3.14.

Prototypes are in the weight range of mobile devices.

We weighted all prototypes for a comparison. The weight of each prototype is stated in Table 3.7. As seen on the table the weight was reduced considerably, so much in fact that our biggest 7.0-*inch* prototype (*Prototype D*) weights less

Figure 3.13: Picture of the assembled final prototypes. The cable that comes out of the lower part of each prototype is connected to the LEDs and can be plugged to a power supply. From left to right: *Prototype A*, *Prototype B*, *Prototype C*, *Prototype D*.

than the 4-*inch* prototype we used for testing. The weight of all the prototypes is in the weight range between the Apple iPhone 7 and the Apple iPad Mini 4, so a mobile device weight was achieved.

These prototypes are light weight and their dark room ensures a very low interference caused by the ambient IR light.

Figure 3.14: Pictures of *Prototype D* from different angles. On the first row: prototype as seen from the front and from the back. On the second row: prototype as seen from the right and from the left. On the third row: prototype as seen from below.

Apple iPhone 7	138
Prototype A	139
Prototype B	171
Prototype C	198
Prototype D	243
Apple iPad Mini 4	299
First Prototype	317

Table 3.7: Weight of each one of the final prototypes. The unit used is $[g]$.

Chapter 4

Scalability

As discussed in Section 3.2 we designed a scalable software that works with diverse prototype sizes, even different ones as the ones predefined in this thesis. In the following paragraphs we will give some design guidelines to create a prototype with a desired interaction area with a width w and a height h.

4.1 Part A (Scalability)

First we need to define the width and height of the 0.8 cm thick *Plexiglas* plate to cut. As described in 3.3 these plates were enlarged by 0.2 cm on the left and on the right side to have an area that could be glued to *part A* afterwards. The size of the plate is given in Table 4.1 relative to the desired width w and height h of the interaction area.

Width	w + 0.4
Height	h

Table 4.1: Size of the 0.8 cm thick scalable *Plexiglas* plate to cut relative to the desired width w and height h of the interaction area. The unit used is $[cm]$.

Then we need to determine the design of *part A*. As seen in Figure 4.1 many values remain unchanged and independent from the interaction area. However, some other values have to be adjusted:

- $A.Width1$ and $A.Height1$ will be equal to the chosen w and h respectively, since they define the interaction area of the part.
- $A.Width2$ will have a length equal to w plus the two margins of 1.5 cm we designed on the sides. $A.Height2$ analogous with margins of 0.3 cm.
- $A.Width3$ and $A.Height3$ will have a length equal to w and h respectively plus the $2 \cdot 0.2$ cm margins reserved to give room to the LEDs (blue area in Figure 4.1). To $A.Width3$ will also be added the $2 \cdot 0.37$ cm margins reserved to glue the *Plexiglas* plate (green area in Figure 4.1).
- $A.Height4$ will have a length equal to $A.Height2$ minus the $2 \cdot 0.45$ cm that separate the protrusion from the upper and lower side of the part.

All values are stated in Table 4.2.

A.Width1	w
A.Height1	h
A.Width2	w + 3.0
A.Height2	h + 0.6
A.Width3	w + 1.14
A.Height3	h + 0.4
A.Height4	h - 0.3

Table 4.2: Relative values of *part A* from Figure 4.1. The unit used is $[cm]$.

To complete *part A* we need to define the borders. As seen in Figure 4.2 the margin values remain unchanged, independent from the interaction area. The values to be adjusted are:

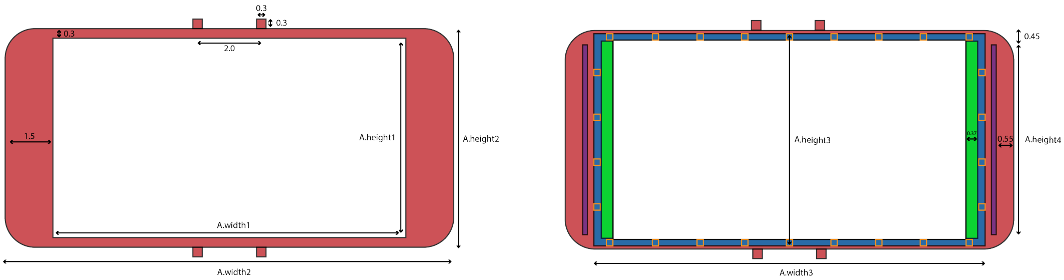

Figure 4.1: Design of *part A* relative to a prototype with an interaction area width w and an interaction area height h. The unit used is $[cm]$. A small margin of 0.3 cm is left on the upper and the lower part to give room to the $0.2 \times 0.2 \times 0.8$ cm LEDs (orange squares). A bigger margin of 1.5 cm is left on the sides for the user to be able to grip the device. Four cubes on the upper and lower side will connect the part with *part C*. The blue and green surface indicate a recess of 0.8 cm. The green area shows the regions of the recess where the *Plexiglas* plate will be glued. The purple area shows a protrusion of 0.05 cm that will serve as a guide to glue the borders. The protrusion has a width of 0.15 cm. The values relative to the chosen width w and height h are stated in Table 4.2. The part has a thickness of 0.9 cm.

- $Border.Width1$ and $Border.Height1$ will be equal to the chosen w and h respectively, since they define the interaction area of the part.
- $Border.Width2$ will have a length equal to w plus the two margins of 0.8 cm we designed on the sides. $Border.Height2$ analogous with margins of 0.3 cm.

All values are stated in Table 4.2.

Border.Width1	w
Border.Height1	h
Border.Width2	w + 1.6
Border.Height2	h + 0.6

Table 4.3: Relative values of the border of *part A* from Figure 4.2. The unit used is $[cm]$.

4.2 Part B (Scalability)

Assuming that the camera going to be used is the

This design for part B only works for the camera we selected.

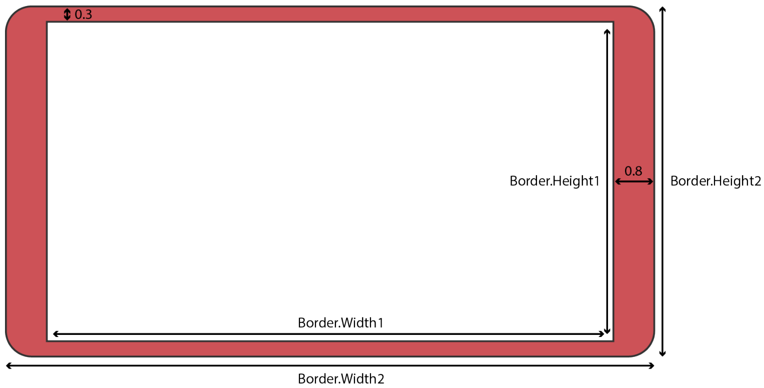

Figure 4.2: Design of Border for *part A* relative to a prototype with an interaction area width w and an interaction area height h. The unit used is $[cm]$. The black lines have to be cut. The values relative to the chosen width w and height h are stated in Table 4.3.

USB500W05G-FD100 from *ELP*, the region of *part B* where the camera gets encapsulated will remain unchanged. If another camera is to be used the design of this region will have to be adjusted accordingly. The height of the part is also uniform across all prototypes, so it will also remain fixed at $6.0\ cm$. As seen in Figure 4.3 the only relative value to adjust is the width of the part:

- $B.Width1$ will be equal to the previously defined $A.Height2$, since this part has to connect two *part Cs* separated by the length of the height of *part A*.
- $B.Width2$ will be equal to half of the calculated $B.Width.1$ minus half of the width of the camera hole ($1.5\ cm$) and minus the borders $(0.5 + 0.2 + 0.4)\ cm$.

All values are stated in Table 4.4.

4.3 Part C (Scalability)

For *part C* we only have to adapt its height, since this part is responsible of keeping the camera at a proper distance

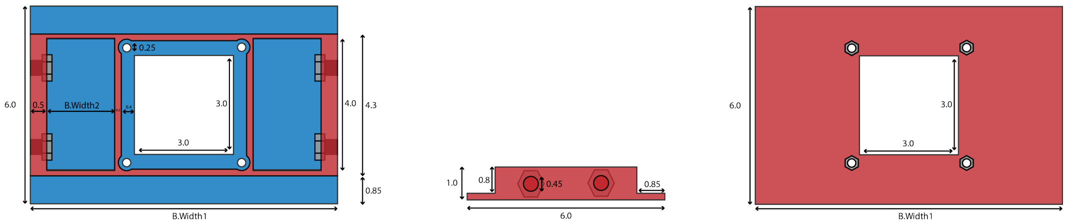

Figure 4.3: Design of *part B* for a scaled prototype. The unit used is $[cm]$. The square hole in the middle will let the camera go though the part. On the left: Front of the part that will be facing the user. The blue surface indicates a recess of $0.8\ cm$. The four rounded holes with a $0.25\ cm$ diameter will let $M2$ screws fix the camera to the part. The upper and lower side will connect the part to *part D*. On the middle: Part as seen from the right. The rounded holes with a $0.45\ cm$ diameter will connect the part to *part C*. A $M4$ nut is glued to the inner side of each hole to be able to fix the part with screws later. On the right: Back of the part that will be facing the plate. Four $M2$ nuts are glued to fix the camera properly. The values relative to the chosen width w and height h are stated in Table 4.4.

B.Width1	$h + 0.6$
B.Width2	$\frac{h}{2} - 2.3$

Table 4.4: Relative values of *part B* from Figure 4.3. The unit used is $[cm]$.

from the interaction area so that the complete plate stays in the image frame. Figure 4.4 shows an example of a scaled *part C*. From our previous designs we learned that the value for *C.Height1* increases by an average of $1.863\ cm$ by every inch the diagonal of the interaction area increases. So the formula to calculate *C.Height1* would be as follows:

$$C.Height1 = \frac{\sqrt{w^2+h^2}cm}{2.54\frac{cm}{inch}} \cdot 1.863\frac{cm}{inch}$$

However, this is only an approximation. It would be preferable to cut a part with different rounded holes like we did with our first prototype, test the camera in different distances, select the best one and cut another part with only the selected holes on the top.

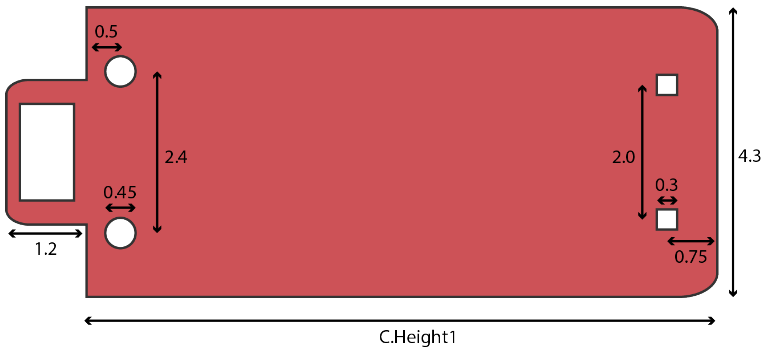

Figure 4.4: Design of *part C* for a scaled prototype. The unit used is $[cm]$. Round holes connect to *part B*, square holes connect to *part A*. The camera USB cable will go through the rectangular hole on the upper side.

4.4 Part D (Scalability)

The design for *part D* is a little more complex, since it connects all the parts, therefore it has to be adapted to its sizes. Figure 4.5 shows a scalable *part D* with a lot of variable values:

- The region measuring $D.Width1$ has to be attached to a side of *part C*, so its length should be equal to *C.Height1* minus the thickness of *part A* ($0.9\ cm$) and part of the thickness of *part B* ($0.8\ cm$).
- The region measuring $D.Height1$ has to be attached to the top of *part A*. Its length will be equal to half of the desired width w minus half of the width of *part C* ($2.15\ cm$).
- $D.Width2$ length is equal to the diagonal that connects *D.Width1* with *D.Height1*, since they have to be connected to form the desired 3D wall. Hence its length will be $\sqrt{D.Width1^2 + D.Height1^2}$.

All values are stated in Table 4.5.

Following these design guidelines one can assemble a prototype of a desired size for future testing. Thanks to the

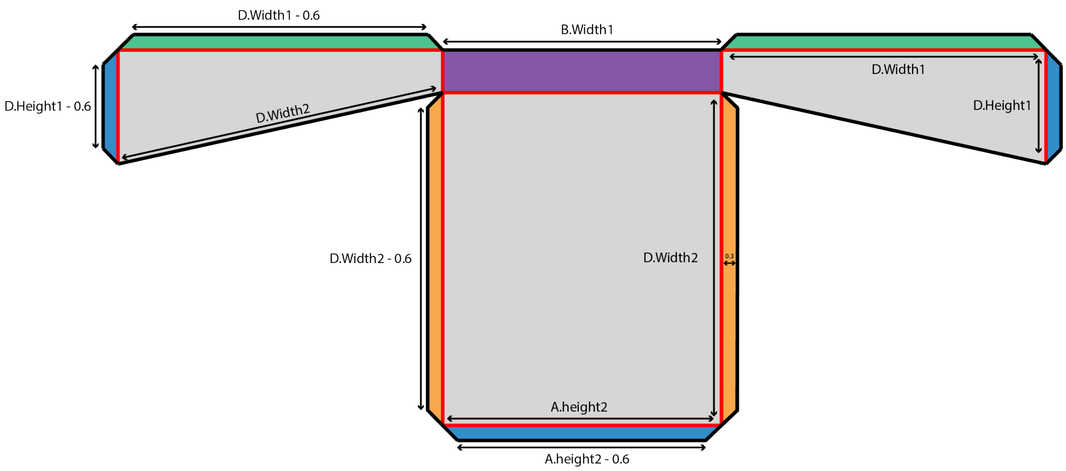

Figure 4.5: Design of *part D* for a scaled prototype. The unit used is $[cm]$. The piece is composed of a body and two arms. The black lines were cut. The red lines were marked to fold the material at these positions. Blue regions connect to *part A*. The purple region connects to *part B*. Green regions connect to *part C*. Orange regions are glued to the inner side of the arms. The values relative to the chosen width w and height h are stated in Table 4.5.

D.Width1	$C.Height1$ - 1.7
D.Height1	$(w/2$ - 2.15
D.Width2	$\sqrt{D.Width1^2 + D.Height1^2}$

Table 4.5: Relative values of *part D* from Figure 4.5. The unit used is $[cm]$.

scalability of the software we use, no changes have to be made in it other than the calibration of the camera.

Chapter 5

Evaluation

To test the reliability of the prototypes we performed a technical evaluation. We defined five points to be measured on the interaction area. The chosen relative values are stated in Table 5.1. We cut four transparent films of the size of the interaction areas of each prototype (as stated in Table 3.2) and marked the defined points with a *Cameo Zing 6030* using the Visicut software. The settings used in the Visicut software were the same as the ones used in Table 3.6. An example of a marked film is given in Figure 5.1.

	X-Position	Y-Position
Measurement 1	0.2	0.2
Measurement 2	0.3	0.8
Measurement 3	0.5	0.5
Measurement 4	0.6	0.3
Measurement 5	0.9	0.8

Table 5.1: Relative Values marked in a transparent film to test all Prototypes.

Measurements were performed with an IR-LED, what provided more precise targeting than a finger.

We placed each marked film over the interaction area of its corresponding prototype and targeted the predefined marks to obtain the results provided by the *reacTIVision Server*. Since the finger was not in direct contact with the plate, the IR light "trapped" inside of it did not scatter. Hence, we performed the measurements by placing an IR-LED on the marked spots instead of a finger. The IR-LED

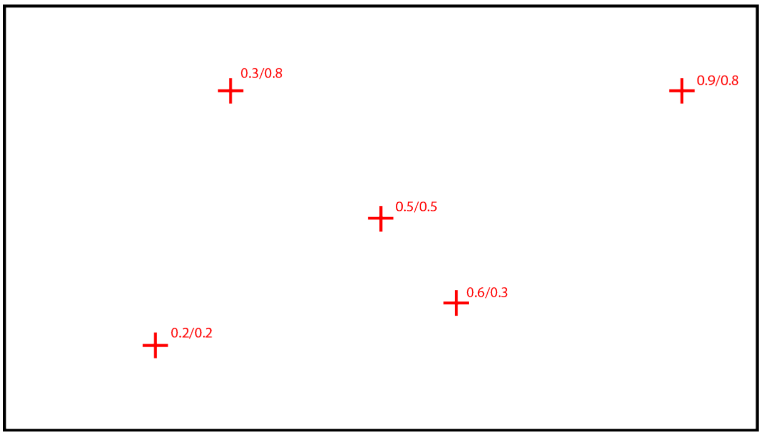

Figure 5.1: Design of the marked transparent film used for the evaluation. The black lines were cut, the red lines were marked. The design showed in the image is from the *Prototype A*. For the other prototypes it was scaled keeping the same relative values for the marked points.

that we used was much smaller than a finger, hence the targeting of the marks was much more precise. Considering that we were testing the performance of the prototypes and not the human performance using fingers, this way of measuring was preferable.

The results provided to our *reacTIVision Client* by the *Server* are stated in Table 5.2, 5.3, 5.4 and 5.5 for the measurements performed in *Prototype A*, *Prototype B*, *Prototype C* and *Prototype D* respectively.

Using the information of the tables we can calculate the mean error of each prototype. For *prototype A* the mean error was 0.02795, for *prototype B* 0.020895, for *prototype C* 0.00745 and for *prototype D* 0.007835. This results in a general mean error of 0.0160325. We calculated a standard deviation of 0.02093 and a 99% confidence interval between 0.009782 and 0.02222.

Mean error of 0.0160325 is very accurate considering the small size of the interaction areas.

Keeping in mind that we are working with relatively small interaction areas of smartphones or tablets, an error offset of 1.6% is very accurate. Also noteworthy is the fact that 47 out of the 80 measurements had an average error of less than 1.0%, while other measurements like *Measurement 4.2* of Table 5.2 or *Measurement 4.1* of Table 5.3 have errors of over 8.0%. These last measurements do not fit in the regular result tendency and are probably due to misplacements of the LED while measuring.

Measurements Prototype A	Target X-Position	Target Y-Position	Measured X-Position	Measured Y-Position	X-Error	Y-Error
1.1	0.2	0.2	0.1990	0.2084	0.0010	0.0084
1.2	0.2	0.2	0.2079	0.1998	0.0079	0.0002
2.1	0.3	0.8	0.3017	0.7526	0.0017	0.0474
2.2	0.3	0.8	0.3017	0.7461	0.0017	0.0539
3.1	0.5	0.5	0.4835	0.5043	0.0165	0.0043
3.2	0.5	0.5	0.4883	0.5109	0.0117	0.0109
4.1	0.6	0.3	0.5795	0.3465	0.0795	0.0465
4.2	0.6	0.3	0.5816	0.3447	0.0816	0.0447
5.1	0.9	0.8	0.8501	0.8159	0.0499	0.0159
5.2	0.9	0.8	0.8480	0.8233	0.0520	0.0233

Table 5.2: Measurements for Prototype A. All values are relative. The position of the target is compared to the measured position and the error (distance between target and touch) is calculated.

Measurements Prototype B	Target X-Position	Target Y-Position	Measured X-Position	Measured Y-Position	X-Error	Y-Error
1.1	0.2	0.2	0.1994	0.2547	0.0006	0.0547
1.2	0.2	0.2	0.2026	0.2619	0.0026	0.0619
2.1	0.3	0.8	0.3056	0.8192	0.0056	0.0192
2.2	0.3	0.8	0.3056	0.8142	0.0056	0.0142
3.1	0.5	0.5	0.5026	0.5002	0.0026	0.0002
3.2	0.5	0.5	0.5037	0.5084	0.0037	0.0084
4.1	0.6	0.3	0.5959	0.3151	0.0959	0.0151
4.2	0.6	0.3	0.5936	0.3264	0.0064	0.0264
5.1	0.9	0.8	0.9011	0.8407	0.0011	0.0407
5.2	0.9	0.8	0.9000	0.8530	0.0000	0.0530

Table 5.3: Measurements for Prototype B. All values are relative. The position of the target is compared to the measured position and the error (distance between target and touch) is calculated.

Measurements Prototype C	Target X-Position	Target Y-Position	Measured X-Position	Measured Y-Position	X-Error	Y-Error
1.1	0.2	0.2	0.2002	0.2170	0.0002	0.0170
1.2	0.2	0.2	0.2013	0.2219	0.0013	0.0219
2.1	0.3	0.8	0.2923	0.7729	0.0077	0.0271
2.2	0.3	0.8	0.2947	0.7767	0.0053	0.0233
3.1	0.5	0.5	0.4935	0.5000	0.0065	0.0000
3.2	0.5	0.5	0.4937	0.5001	0.0063	0.0001
4.1	0.6	0.3	0.6006	0.2987	0.0006	0.0013
4.2	0.6	0.3	0.6010	0.3013	0.0010	0.0013
5.1	0.9	0.8	0.8901	0.7953	0.0099	0.0047
5.2	0.9	0.8	0.8870	0.8005	0.0130	0.0005

Table 5.4: Measurements for Prototype C. All values are relative. The position of the target is compared to the measured position and the error (distance between target and touch) is calculated.

Measurements Prototype D	Target X-Position	Target Y-Position	Measured X-Position	Measured Y-Position	X-Error	Y-Error
1.1	0.2	0.2	0.2053	0.1782	0.0053	0.0218
1.2	0.2	0.2	0.2074	0.1780	0.0074	0.022
2.1	0.3	0.8	0.3113	0.8021	0.0113	0.0021
2.2	0.3	0.8	0.3117	0.8032	0.0117	0.0032
3.1	0.5	0.5	0.4916	0.5040	0.0084	0.0040
3.2	0.5	0.5	0.4966	0.4973	0.0034	0.0027
4.1	0.6	0.3	0.6056	0.3165	0.0056	0.0165
4.2	0.6	0.3	0.6054	0.3110	0.0054	0.0110
5.1	0.9	0.8	0.8951	0.7994	0.0049	0.0006
5.2	0.9	0.8	0.8929	0.7977	0.0071	0.0023

Table 5.5: Measurements for Prototype D. All values are relative. The position of the target is compared to the measured position and the error (distance between target and touch) is calculated.

Chapter 6

Summary and future work

In this thesis we described an *FTIR* based scalable measurement tool to capture back-of-device touch data. We presented a software capable of analyzing touch events on an acrylic plate as well as four hardware prototypes that work with that tool. We also gave design guidelines for scaling the hardware prototypes to other sizes.

6.1 Limitations

The client software does not fully support multi-touch. If more than one finger is present, the information of all the fingers will be registered and saved in the CSV1 archive, but only one finger will be shown in the *Preview Window* (given that this option was previously activated in the *Control Panel Window*) and only one finger will be saved in the CSV2. If fully multi-touch support is desired, our software will have to be modified.

Client software does not show a preview of multiple fingers.

We designed our prototypes to work for a specific eyes-free back-of-device interaction experiment. The camera, *part B* and *part D* block the field of vision of the user and make it impossible for her to see the interaction area. Hence our

Interaction area is always out of the field of view of the user.

designs will not work for performing an experiment where the user gets visual feedforward of the input area.

Only back-of-device interaction,

Our designs only work for a back-of-device interaction, since they would not have an ergonomic grip while using it in reverse.

Wired designs.

All designs use a wired camera and wired IR-LEDs. A wireless prototype would resemble better a mobile device, but keeping them wired allowed us to keep the weight as low as possible.

6.2 Future work

The tool presented in this paper may be used in future research to explore how human accuracy varies at indirect eyes-free back-of-device touch pointing tasks when using interaction areas of different sizes while keeping an equal aspect ratio.

This will help answer the question of how *HaptiCase* from Corsten et al. [2015] performs for devices with different screen sizes.

Our design guidelines could also potentially be used in any future research related with back-of-device interaction for the quick and cheap development of FTIR based prototypes.

Bibliography

Christian Corsten, Christian Cherek, Thorsten Karrer, and Jan Borchers. Hapticase: Back-of-device tactile landmarks for eyes-free absolute indirect touch. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 2171–2180. ACM, 2015.

Jérémie Gilliot, Géry Casiez, and Nicolas Roussel. Impact of form factors and input conditions on absolute indirect-touch pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 723–732. ACM, 2014.

Jefferson Y Han. Multi-touch sensing through frustrated total internal reflection, November 20 2008. US Patent 20,080,284,925.

Sebastian Hueber. *Back-of-device Tactile Landmarks for Eyes-free Touch Input*. PhD thesis, RWTH Aachen University, 2015.

Martin Kaltenbrunner. reactivision and tuio: a tangible tabletop toolkit. In *Proceedings of the ACM international Conference on interactive Tabletops and Surfaces*, pages 9–16. ACM, 2009.

Jacob O Wobbrock, Brad A Myers, and Htet Htet Aung. The performance of hand postures in front-and back-of-device interaction for mobile computing. *International Journal of Human-Computer Studies*, 66(12):857–875, 2008.

Katrin Wolf, Robert Schleicher, and Michael Rohs. Touch accessibility on the front and the back of held tablet devices. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*, pages 161–168. Springer, 2014.

Christopher Wolfe, J David Smith, and TC Graham. A low-cost infrastructure for tabletop games. In *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*, pages 145–151. ACM, 2008.

Index

Typeset September 27, 2016