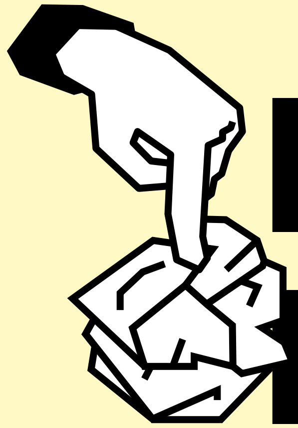
Team
Thea Eckert
Nassim Laarmann

Winter semester 2023/24

Course
Interaction Engineering

Supervisor
Prof. Dr. Michael Kipp

Augsburg University
of Applied Sciences



PAPER PROTOTYPING FOR PHYSICAL INTERFACES

ABSTRACT

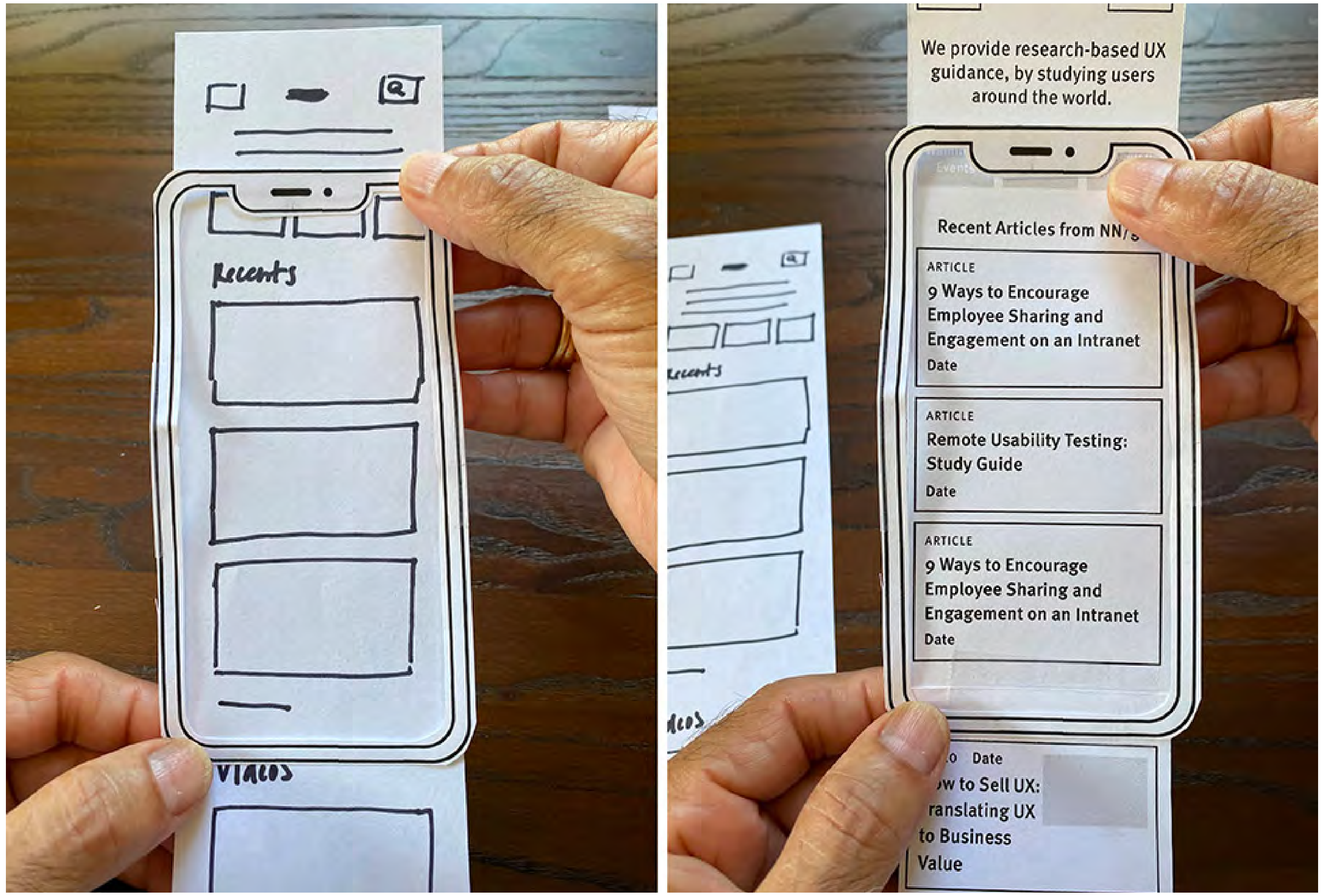
This project explores the integration of tangible interfaces with paper prototyping, utilizing fiducial markers and computer vision. The primary aim is to provide designers with a versatile and accessible method for rapidly prototyping tangible interactions. By employing household materials such as cardboard, rubber, and paper, along with common tools like scissors and glue, designers can easily craft tangible interface elements. The manipulation of ArUco Markers enables the creation of diverse interaction modalities by adjusting marker visibility, position, and orientation. The implementation of this concept is exemplified through the design and evaluation of tangible interfaces for controlling the classic game Pong. These interfaces include a range of haptic elements such as knobs, buttons, sliders, all constructed from household materials. The project showcases the potential of paper-based prototyping for tangible interactions and provides practical insights into the design process.

1 Motivation 4
2 Related work 5
3 Concept *6*
3.1 Crafting *7*
3.2 Interaction techniques *8*
3.3 Application *12*
4 Implementation 13
4.1 Interface setup *13*
4.2 Technical setup *15*
5 Evaluation 16
5.1 Study setup 17
5.2 Results 18
Conclusion 20

TABLE OF CONTENTS

1_MOTIVATION

Paper prototypes serve as tangible representations of digital interfaces and are commonly employed for user testing to evaluate initial concepts before committing resources to a refined product development. While they are widely utilised for digital interfaces such as mobile applications or websites, their application to physical objects such as handheld gaming controllers or other tangible interfaces is still limited. Our project closes this gap by enabling the fast and straightforward testing of preliminary interface concepts and ensuring validation of initial concepts, reducing the need for significant investment of time, resources and capital in polished prototypes at an early stage of development. By using fiducial markers, our project improves paper prototypes and enables an integration with digital screens and applications without the need for the Wizard-of-Oz principle for interaction control. Using computer vision technology, interaction inputs can be processed directly, resulting in digital outputs on the screen.



[Fig. 1] Paper Prototype by Norman Nielsen Group

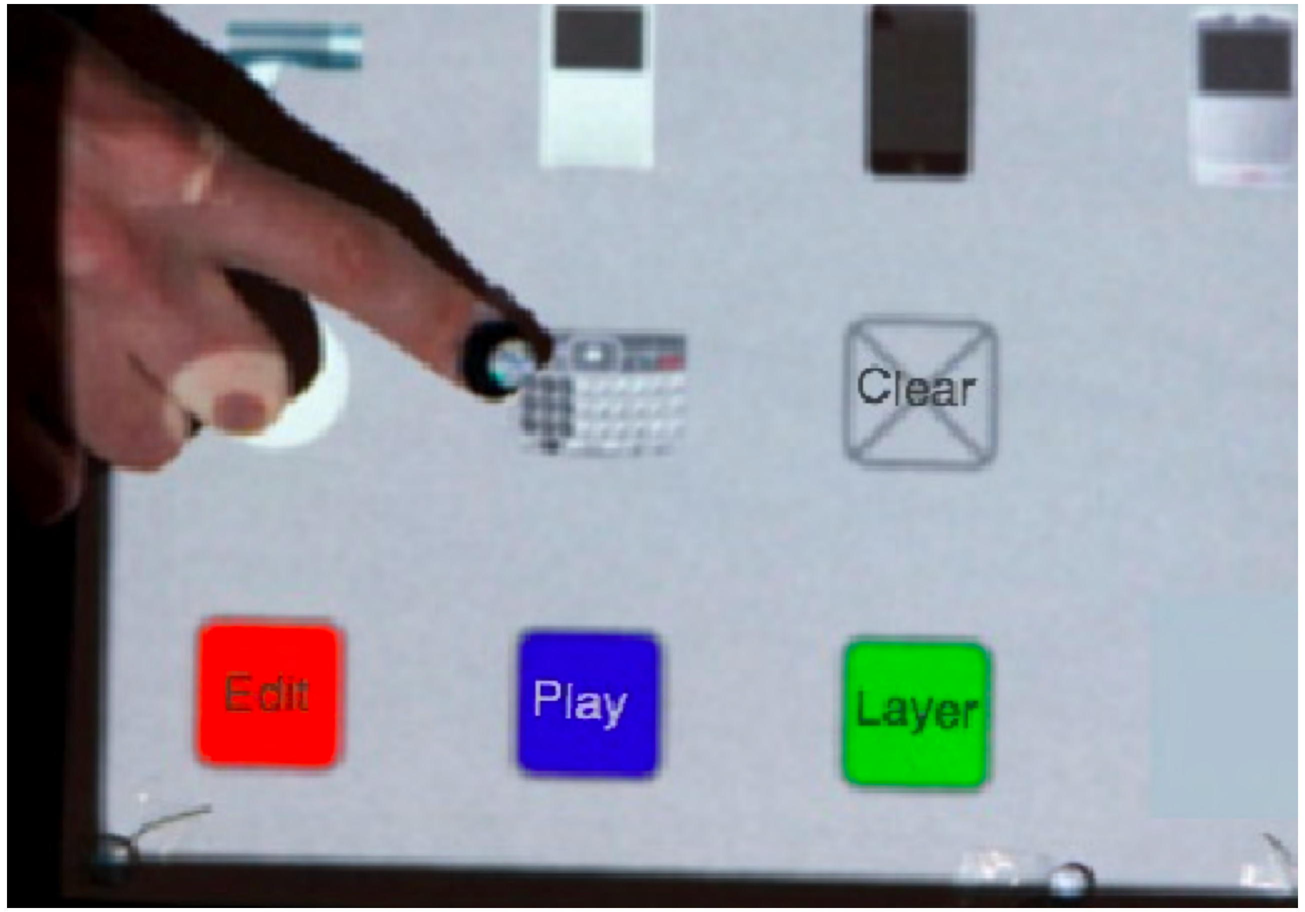
2_RELATED WORK

In our exploration of related work, we primarily reference the research conducted by Clement Zheng, Peter Gyory and Ellen Yi-Luen Do, presented in their paper “Tangible Interfaces with Printed Paper Markers” at the ACM Designing Interactive Systems Conference (DIS ‘20) 2020 [1][Fig. 2]. This study investigates the range of interaction possibilities using paper with printed ArUco markers. Building on their findings into effective practices for working with printed fiducial markers, our project extends the application of these markers to introduce new interaction components, thereby expanding the possibilities of paper-based tangible interfaces. In addition, we are exploring the use of alternative media beyond traditional paper and experimenting with different designs of interface elements.

[Fig. 2] Zheng et al.

Another related research is one presented by Eric Akaoka, Tim Ginn, and Roel Vertegaal in their paper titled “DisplayObjects: prototyping functional physical interfaces on 3D styrofoam, paper, or cardboard models,” showcased at the fourth international conference on Tangible, Embedded, and Embodied Interaction (TEI ‘10) [2][Fig. 3]. This paper introduces a rapid prototyping method where functional interfaces are projected onto physical representations made out of styrofoam, enabling an exploration and evaluation of interface concepts.

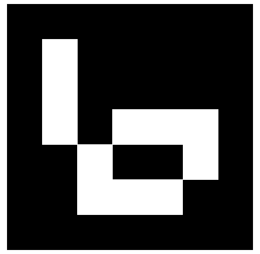
[1] Zheng et al.
[2] Akaoka et al.



[Fig. 3] Akaoka et al.

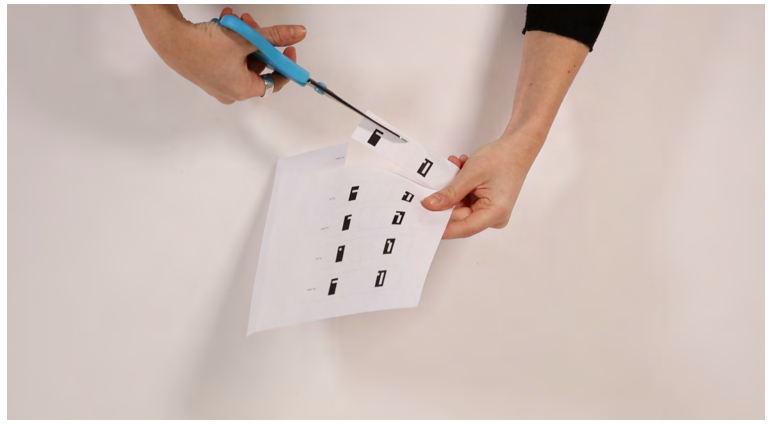
3_CONCEPT

Fiducial markers represent the core of this research, marking the boundaries of interaction possibilities. The insights and overview of potential markers provided by Zheng et al. (2020) have been essential in the research of this project. Additionally, our exploration of open-source marker libraries, such as ReacTIVision and ARToolkit, led us to select ArUco markers from OpenCV due to their alignment with project requirements. ArUco markers [Fig. 4], geometrically square, are identified using a dictionary from OpenCV. This marker type was chosen for its wide set of functions, enabling a seamless integration of software output with real-world interactions. Markers sourced from the OpenCV dictionary have proven to be reliable and stable in our experiments.



[Fig. 4] ArUco Marker with the ID 27.

[Fig. 5] *Collection of used materials.*



[Fig. 6] *Cutting out the desired shapes of the ArUco Marker.*

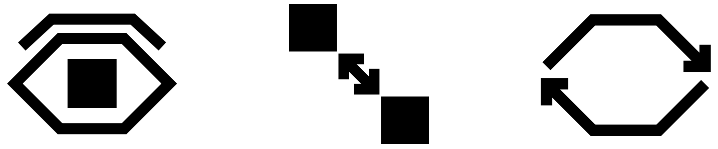
CRAFTING

All three interaction techniques required different crafting implementations. We opted for materials readily available in households or easily obtainable from nearby stores. The materials used for constructing the interfaces for testing included cardboard, paper, pins, rubber bands, clamps, and sticky notes. To enhance the haptic experience and improve interface ergonomics, sturdier and larger components such as coffee pods, LEGO bricks, and cups were also added [Fig. 5].

A critical consideration during interface creation was to cut out the areas where interaction occurs. This involved ensuring that the markers corresponding to the desired interactions were positioned within the camera's viewport. The interface was divided into two distinct sections: the upper part, which was visible and used by the user, and the bottom part, which contains markers linked to each interaction.

The markers were printed on normal printer paper using a laser printer. When the markers were used, there were severe signs of wear and tear. In this case, when selectively fixing white spots, a black pen was used.

INTERACTION TECHNIQUES

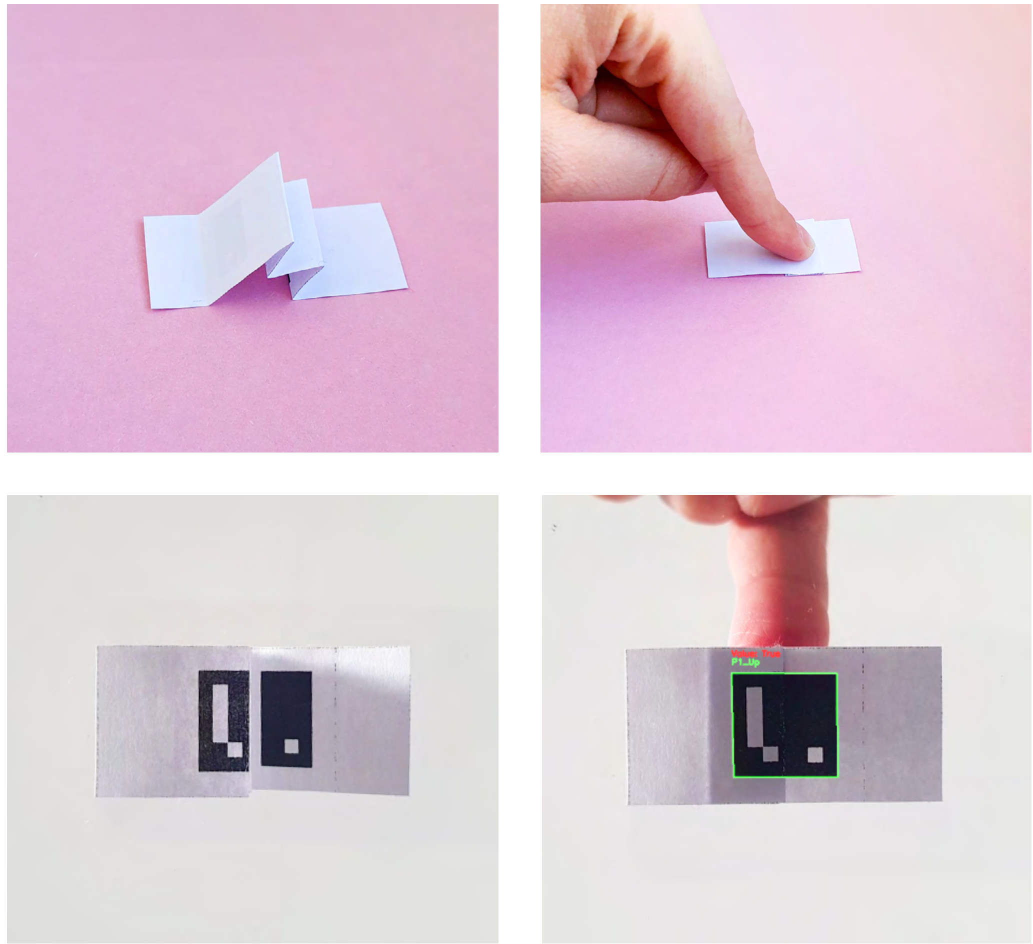


Interaction techniques are centered around the data collected by computer vision from ArUco markers. We have identified three primary types that will be described in the following paragraph: (1) marker visibility, (2) marker position, and (3) marker rotation. Marker position and orientation can be determined across all three axes.

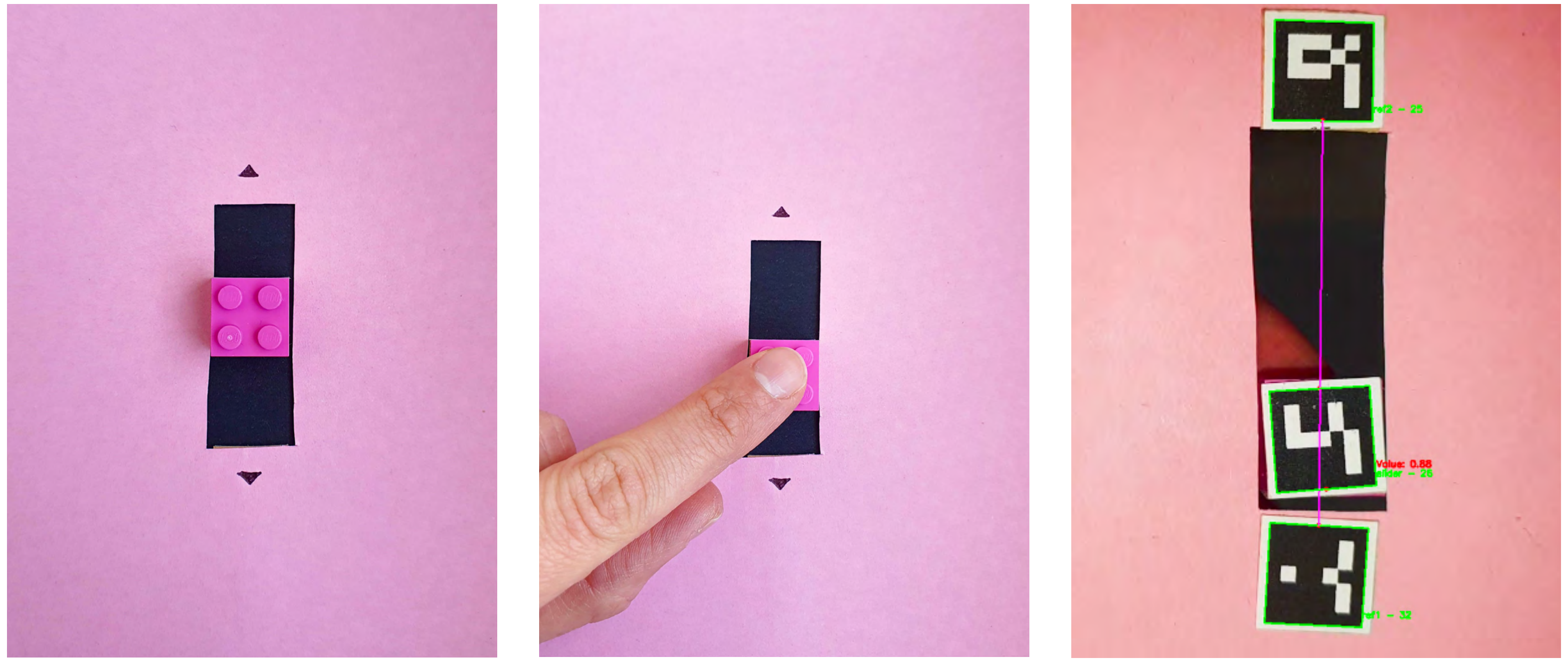
Marker Visiblity

The interaction technique resulting from marker visibility can be compared to existing interface elements, such as buttons, toggle switches, checkboxes and radio buttons. All those elements have in common that they only allow binary states - ON and OFF that are mapped to visible and invisible.

Creating interaction components of this nature with the materials referenced in this project relies on a method for revealing or concealing the marker for a specific duration. This can be achieved either by manually manipulating the marker's visibility to the camera or by crafting a component that facilitates interaction with visibility. For the latter approach, we utilized the crafting instructions for buttons featured in the paper by Zheng et al. (2020). The button design provided in their fabrication file included templates for printing, cutting, and folding. This button design translated a pushing motion into the completion of the printed marker for detection by computer vision. The folding technique reinforced the structure, resulting in increased tactile feedback upon pushing.



[Fig. 7] Interaction with a crafted button and marker recognition with computer vision.



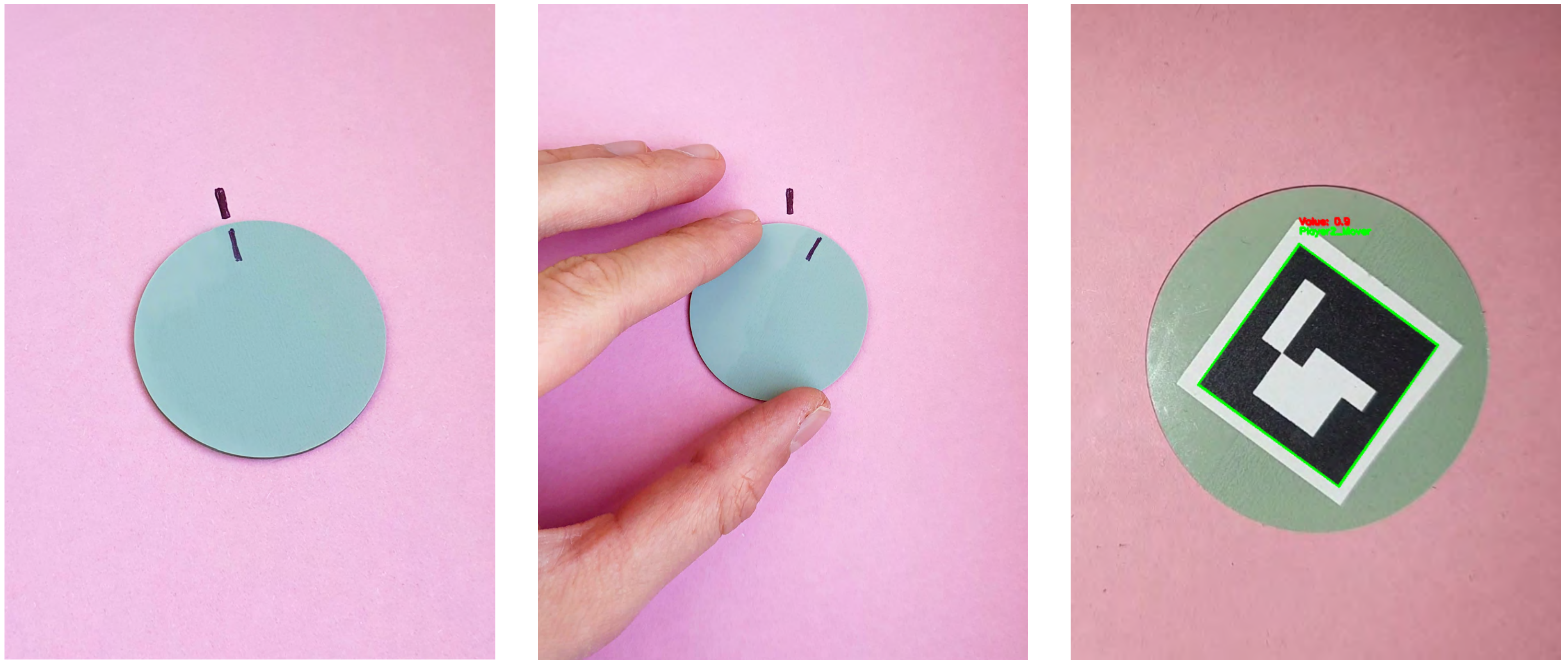
[Fig. 8] *Interaction with a crafted slider element and marker recognition with computer vision.*

Marker position

The marker's position can be tracked along movement in all three axis directions. Common interface elements in digital interfaces for reading continuous values include sliders, steppers, and range selectors. Establishing a range for the minimum and maximum values involves positioning two reference markers as endpoints and placing the value marker between them. This indicates the current relative position between the references, determining a value within the specified range.

To craft a component capable of reading positional values, it must incorporate constraints for moving the value marker between the two reference markers. This can be achieved by creating a channel, allowing the element with the value marker to move along a linear path. The two reference markers are then positioned at each end of the channel to define the range.

In order to enhance the tangibility, bigger objects can be used as slider. We experienced a better haptic experience with LEGO bricks and the value marker glued on the bottom. Indicators that show what the action of the slider is were added by drawing on the paper or cardboard.



[Fig. 9] Interaction with a crafted knob element and marker recognition with computer vision.

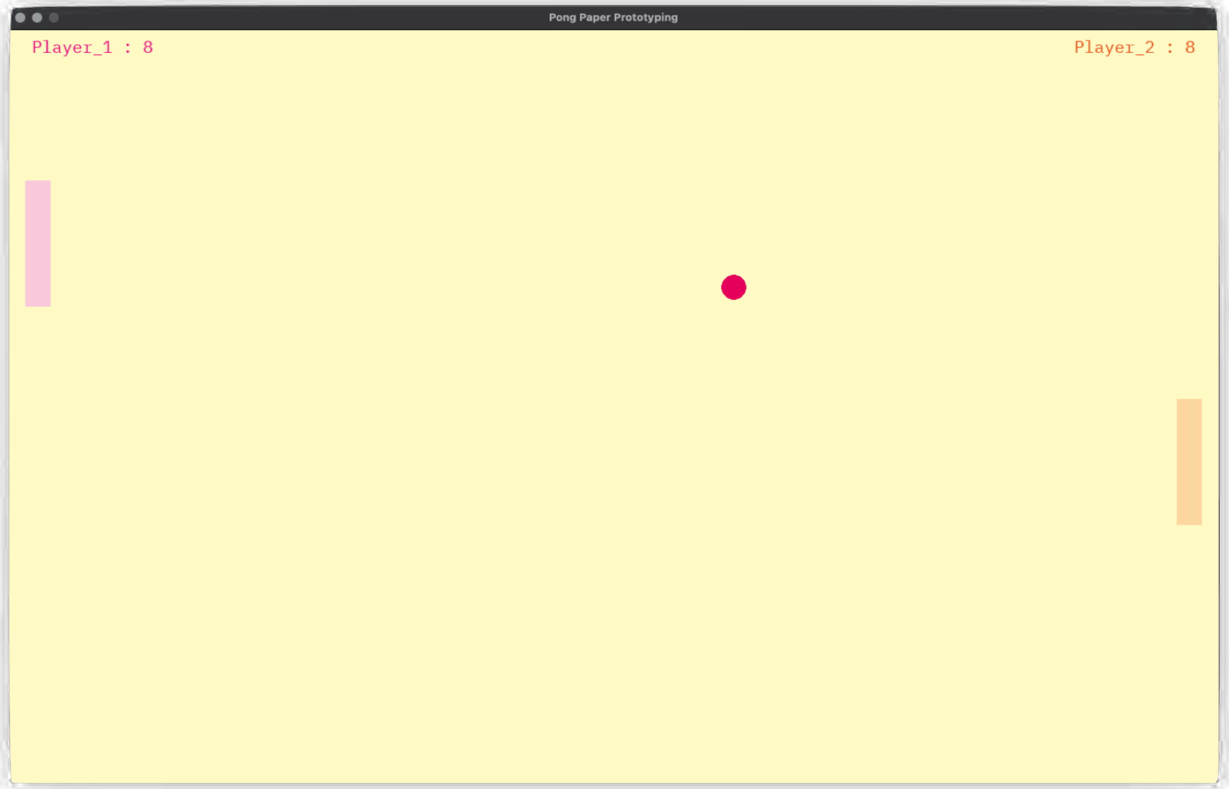
Marker rotation

The final implemented interaction method detects the rotational value of markers. However, while orientation data for all three axes can be obtained, we are only capturing the value of the axis aligned vertically to the camera. Equivalent interaction elements that are commonly found in digital interfaces that derive continuous values from rotation are knobs and jog wheels.

Constructing this component follows a similar approach to sliders, requiring a cut-out to expose the marker to the camera. To indicate the initial position, a line was drawn on both the frame and the rotatable object. For the rotary part, we opted for a taller object, such as a coffee pod or a stack of sticky notes.

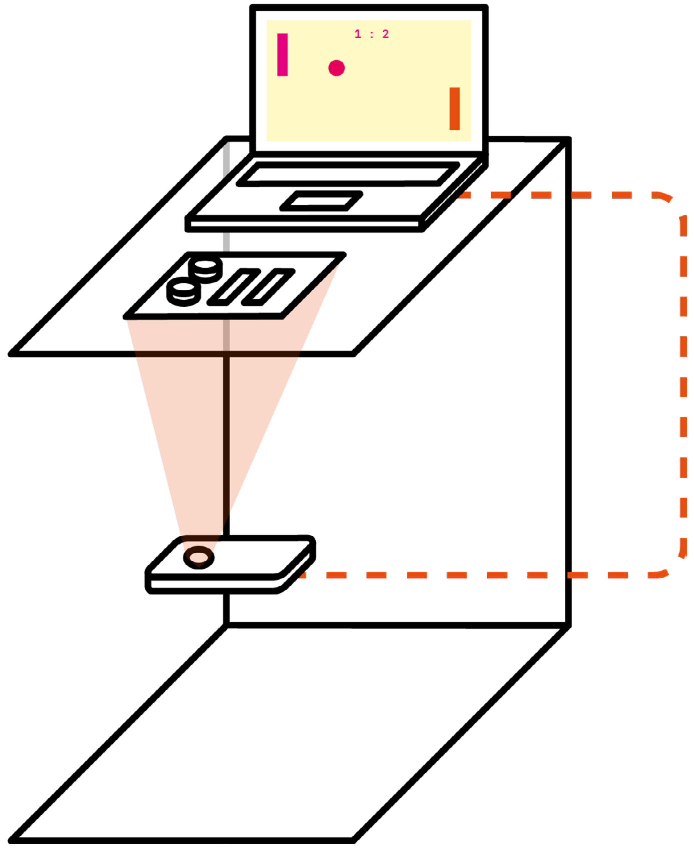
APPLICATION

The overall objective and vision of this project are to integrate the built prototype interfaces with various applications. To ensure timely progress in this project, we chose the game “Pong” as the application for testing interfaces. “Pong” is a video game, originally released by Atari, that simulates tennis. The game includes two striker paddles and a ball, with the goal being to outscore the opponent by hitting the ball and directing it into their side.



[Fig. 10] *Implemented Pong game.*

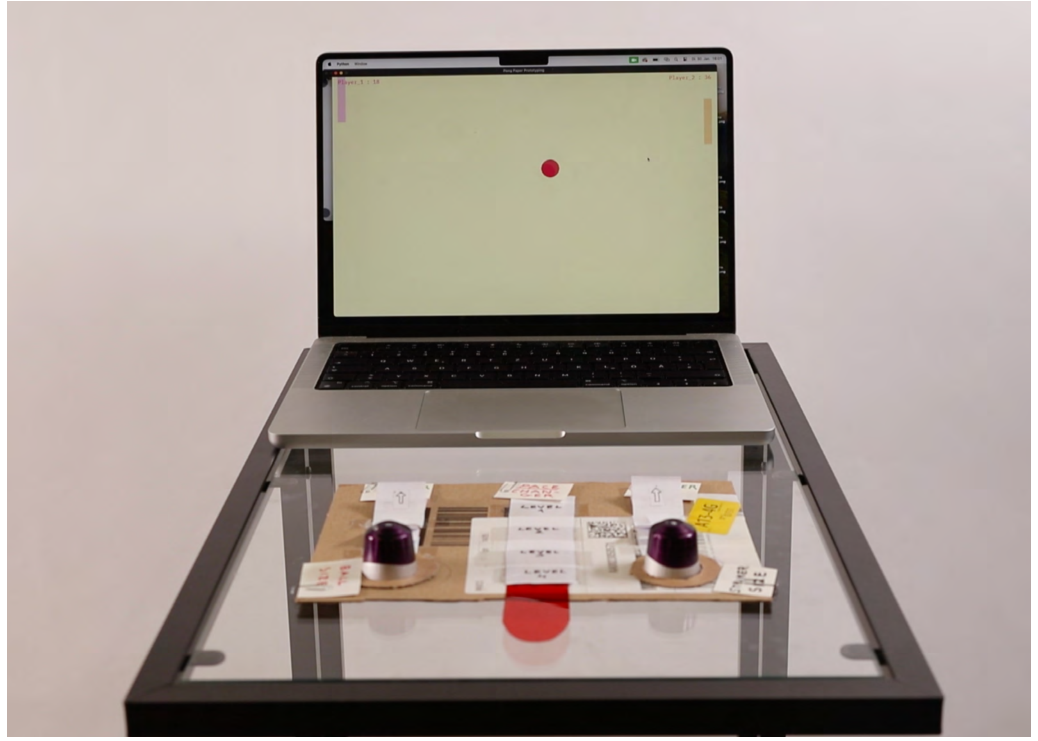
4_IMPLEMENTATION



[Fig. 11] *Sketch of the pyhsical setup including a laptop, webcam and the crafted interface.*

INTERFACE SETUP

The setup required a tabletop for placing both the interface and the laptop, while still ensuring marker recognition by the camera for markers positioned beneath the interface. These requirements were met through a glass tabletop. The recognisability of the markers was significantly influenced by the lighting conditions, which made it necessary to adapt the lighting on the spot depending on the situation in the room.



[Fig. 12] *Setup of the interface and laptop.*

[Fig. 13] *Setup of the interface and laptop (top view).*

TECHNICAL SETUP

As mentioned earlier, this project heavily relies on OpenCV [3], a real-time computer vision library and toolbox. Python was chosen as the programming language due to its compatibility with OpenCV and the Pygame library, which was utilized for the “Pong” game application.

We utilized the ArUco Original dictionary within the ArUco Marker Library and generated markers using an online ArUco marker generator by Oleg Kalachev [4]. OpenCV facilitated marker identification on webcam frames, while additional Python scripts were developed for interaction mapping and interface element creation. For the “Pong” game application, we adapted an existing script from GeeksForGeeks [5] utilizing the Pygame module.

To improve code visibility and adaptability, we established two areas for code adjustment. One script contains a list of available UI elements and their corresponding marker IDs *[Marker_Elements.py]*, simplifying interaction element addition for designers only requiring minimal coding skills.

```
# EXAMPLE
UI_ELEMENTS = {
    "buttons": [{"name": "Click_Button", "id": 2}, {"name": "Click_Button2", "id": 4}],
    "sliders": [
        {"name": "Slide me", "ids": [3, 4, 5]},
    ],
    "knobs": [{"name": "Turn me", "id": 1}],
}
```

[Fig. 14] Excerpt of the script [Marker_Elements.py] for marker ID editing and UI element addition.

The second area for script adjustment is the mapping-function, linking each interface element to specific game actions *[main.py > fn:handle_aruco_detection()]*. Although this currently requires coding skills, we plan to streamline this process in the future by providing a GUI for interaction mapping.

To only utilize available household objects, we chose to use the smartphone as a webcam for marker detection. We achieved this by employing the software Iriun [6], which can be installed on both the device running the application and a smartphone. This approach enabled us to wirelessly connect the phone’s camera as a webcam.

[3] OpenCV
[4] Oleg Kalachev
[5] GeeksForGeeks
[6] Iriun

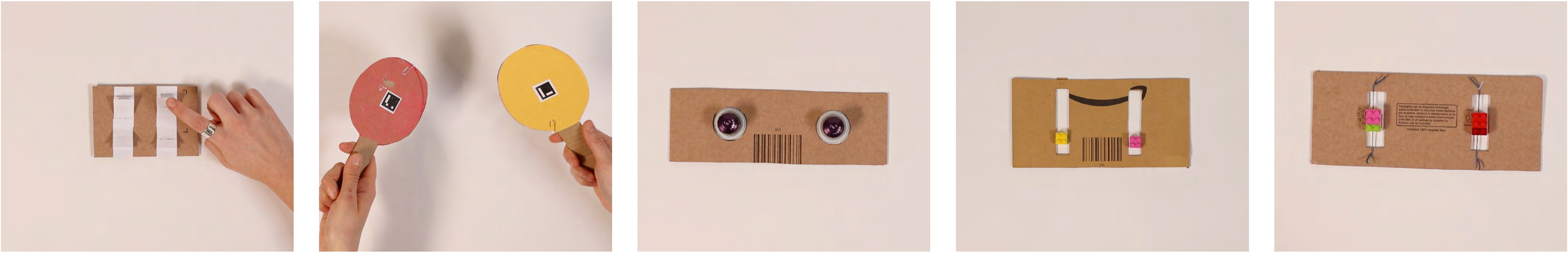
A user study was conducted to explore the usability and gameplay enjoyment associated with the provided interfaces. The study employed a within-subject design, in which each participant engaged in every task.

5_EVALUATION

STUDY SETUP

During the study, participants engaged in pairs, playing Pong against each other using different interfaces. Minimal instructions were provided on how to operate the assigned interface. The following five interfaces in the given order were presented for evaluation:

[Fig. 15] *Crafted interfaces for the evaluation.*



1. Buttons: Two button pairs to move the striker paddle up and down for each player.

2. Ping pong rackets: Two handheld representations of Ping Pong Rackets which need to be turned to reveal the markers mapped to moving the striker paddle up and down.

3. Knobs: One knob for each player which moves the striker paddle up and down. The absolute position of the striker paddle is mapped to the knob's minimum and maximum value.

4. Sliders: One slider for each player to move the paddle up and down. The position of the slider element represents the absolute position of the striker paddle.

5. Velocity shifters: One shifter for each player. A sliding element is attached to a rubber band for haptic force feedback. This interface controls the velocity of the striker paddle in each direction.

Following each task, participants completed a questionnaire adapted from Lund's (2001) USE questionnaire. A simplified version was used to collect insights on four categories: usefulness, satisfaction, ease of use, and ease of learning. A 5-point Likert scale was utilized to streamline testing time. The questionnaire for each task contained the following statements:

1. Usefulness
- *a. It is useful.*
- *b. It meets my needs.*
- *c. It does everything I would expect it to do.*

2. Ease of Use
- *a. It is easy to use.*
- *b. Using it is effortless.*
- *c. I can use it without written instructions*

3. Ease of Learning
- *a. I learned to use it quickly.*
- *b. I quickly became skillful with it.*

4. Satisfaction
- *a. I am satisfied with it.*

Upon completion of all tasks, participants were asked to rank the five interfaces hierarchically from least to most favorite.

Finally, participants were asked to provide feedback to clarify their preferences for the least and most favorite interfaces.

The evaluation study was conducted with 9 participants with ages ranging between 21 and 60, and varying technical affinity.

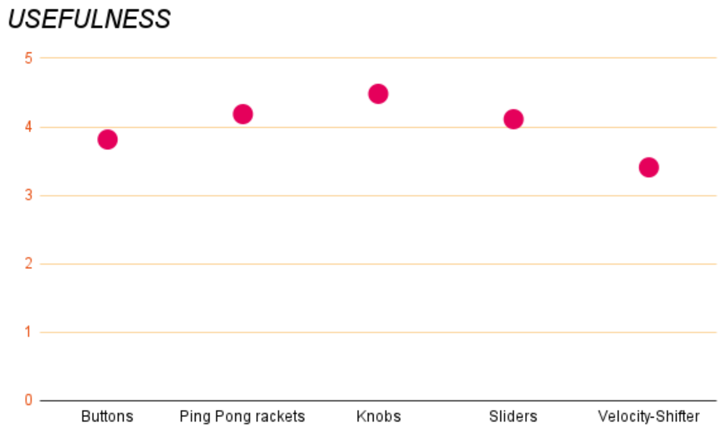
RESULTS

Questionnaire

The results of the questionnaire for each interface is summarized in each category of the USE questionnaire. The full result of each statement is located in the Appendix.

Usefulness

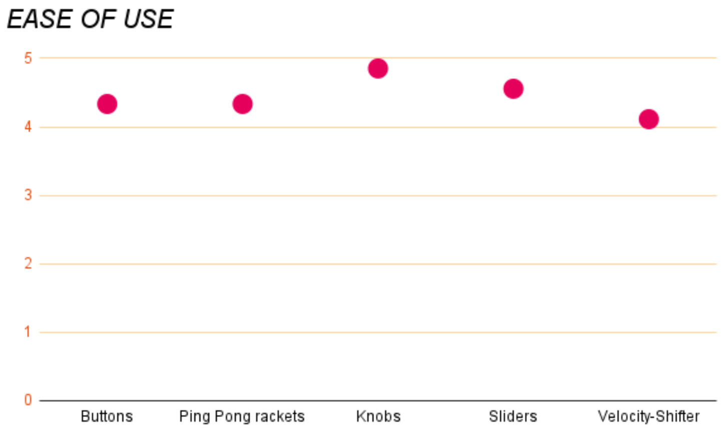
Overall, participants found each interface to be useful, with the knob interface standing out as the most useful option.



[Fig. 16] Evaluation of the category "Usefulness".

Ease of Use

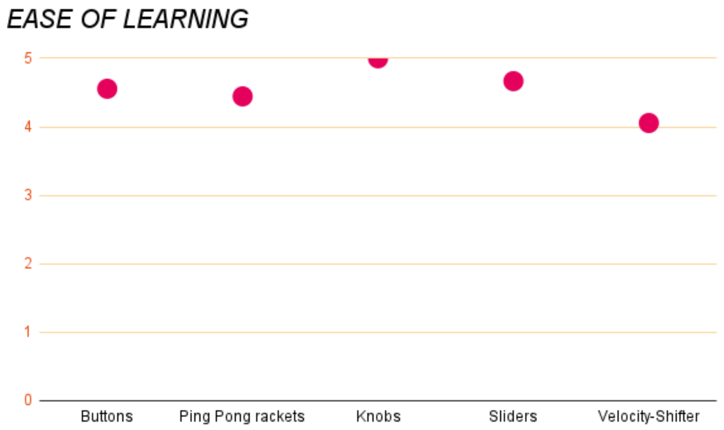
Similar to the category of usefulness, participants also regarded the Knob interface as the easiest to use.



[Fig. 17] Evaluation of the category "Ease of Use".

Ease of Learning

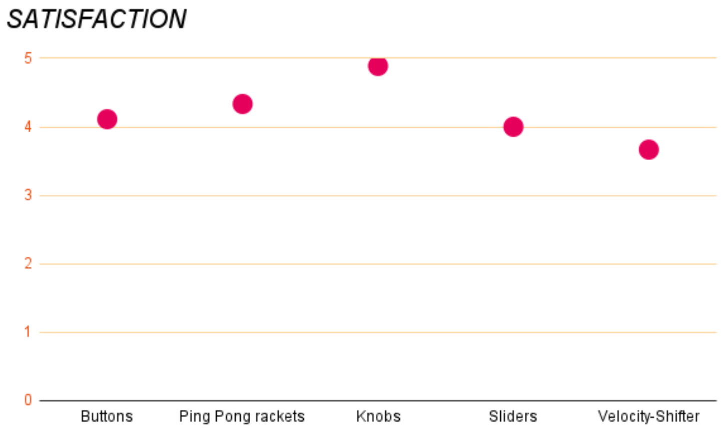
Overall, participants found all interfaces to be easy to learn, with the Knob interface being the easiest to learn.



[Fig. 18] Evaluation of the category "Ease of Learning".

Satisfaction

The satisfaction rating shows the participants preference for the knob interface.



[Fig. 19] Evaluation of the category "Satisfaction".

Rating

The average rating position of each interface from most to least favorite is the following:

1. *Knobs*
2. *Slider*
3. *Ping pong rackets*
4. *Buttons*
5. *Velocity shifter*

Open questions

Here are the user study results presented in response to open-ended questions:

Participants consistently favored the knob interface for adjusting the striker paddles' positions, often expressing surprise at its functionality. However, some initially struggled to understand the need to rotate the knobs.

Concerns regarding the least favored interface, the velocity shifter, primarily revolved around the slow movement of the striker paddle and the irregular marker recognition, leading to gameplay disruptions. To improve this issue, the acceleration factor can be adjusted in the script.

Positive feedback regarding the ping pong racket interface focused on its visually appealing design, particularly the use of "fun" colors. However, some participants found it challenging to stop the striker paddle movement, especially because at least one marker always remained visible.

Participants encountered difficulties with the button interface, as they were not immediately recognized as pressable buttons. Furthermore, marker recognition was disrupted when buttons were pressed too firmly in the heat of battle. This issue can be prevented by altering the marker position on the cardboard itself. In this version, the buttons were slightly hovering above the glass plate when pressed. They need to be positioned to the bottom of the interface instead.

Summary

The questionnaire results and ratings shows a consistent connection between participants' assessments of usefulness, satisfaction, and ease of learning in relation to the interfaces. The most preferred interface aligns with being perceived as the most useful, easiest to learn, and the most satisfying to use.

However, insights from the open-ended questions indicate that opinions, both in favor of and against an interface, often revolve around its physical appearance or the material used. Participants frequently expressed confusion when the interaction element did not work, primarily attributing it to issues related to marker recognition rather than a lack of a corresponding action mapping. For future iterations, providing a brief orientation emphasizing that these interfaces are prototypes with potential limitations could prove beneficial. Participants could be advised to prioritize the interactive element and gameplay experience over the functionality of the interface.

CONCLUSION

When developing physical products with interactive functions, only a few usability test methods are generally used. Creating physical representations with materials such as Styrofoam or modelling clay is popular, but often requires a significant amount of time, is not repeatable and, most importantly, cannot test interactions because the interface is not connected to an output source. This project emphasises the quick and effortless creation of numerous interfaces.

Future directions will definitely be to simplify the process of seamlessly mapping interaction elements to software actions through a graphical user interface. Such integration would streamline the process by linking each interaction technique to specific outputs in the program.

As the application presented was limited to the game Pong, future enhancements aim to extend compatibility with different types of programmes for interaction testing.

Although the evaluation interfaces focused primarily on simple controls for positioning the striker paddle, numerous other attributes remain open for customisation. This project also explored interfaces with advanced features and different materials to explore the possibilities of marker recognition [Fig. 20]. These interfaces include buttons for paddle control, buttons to change ball and striker sizes, and a level changer to increase ball speed.

[Fig. 20] *Interface with advanced features.*

REFERENCES

Page 5: Related work

[1] Clement Zheng, Peter Gyory, and Ellen Yi-Luen Do. 2020. Tangible Interfaces with Printed Paper Markers. In Proceedings of the 2020 ACM Designing Interactive Systems Conference (DIS '20). Association for Computing Machinery, New York, NY, USA, 909–923. https://doi.org/10.1145/3357236.3395578.

[2] Eric Akaoka, Tim Ginn, and Roel Vertegaal. 2010. DisplayObjects: prototyping functional physical interfaces on 3d styrofoam, paper or cardboard models. In Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction (TEI '10). Association for Computing Machinery, New York, NY, USA, 49–56. https://doi.org/10.1145/1709886.1709897.

Page 15: Technical setup

[3] OpenCV. Documentation. URL: https://opencv.org/ [3.02.24].

[4] Oleg Kalachev. ArUco Markers Generator. URL: https://chev.me/arucogen/ [3.02.24].

[5] GeeksForGeeks. Create a Pong Game in Python – Pygame. URL: https://www.geeksforgeeks.org/create-a-pong-game-in-python-pygame/ [3.02.24].

LIST OF FIGURES

[Fig. 1] Paper Prototype by Norman Nielsen Group. URL: https://www.nngroup.com/articles/paper-prototyping-cutout-kit/ [3.02.24].

[Fig. 2] Zheng et al. Tangible Interfaces with Printed Paper Markers. URL: https://dl.acm.org/doi/10.1145/3357236.3395578 [3.02.24].

[Fig. 3] Akaoka et al. DisplayObjects. URL: https://www.humanmedialab.org/blog/displayobjects [3.02.24].

[Fig. 4] ArUco Marker with the ID 27. Generated with OpenCV.

[Fig. 5] Collection of used materials. Own photograph.

[Fig. 6] Cutting out the desired shapes of the ArUco Marker. Own photograph.

[Fig. 7] Interaction with a crafted button and marker recognition with computer vision. Own photograph.

[Fig. 8] Interaction with a crafted slider element and marker recognition with computer vision. Own photograph.

[Fig. 9] Interaction with a crafted knob element and marker recognition with computer vision. Own photograph.

[Fig. 10] Implemented Pong game. Own photograph.

[Fig. 11] Sketch of the pyhsical setup including a laptop, webcam and the crafted interface. Own photograph.

[Fig. 12] Setup of the interface and laptop. Own photograph.

[Fig. 13] Setup of the interface and laptop (top view). Own photograph.

[Fig. 14] Excerpt of the script [Marker_Elements.py] for marker ID editing and UI element addition. Own screenshot.

[Fig. 15] Crafted interfaces for the evaluation. Own photograph.

[Fig. 16] Evaluation of the category "Usefulness". Own graph.

[Fig. 17] Evaluation of the category "Ease of Use". Own graph.

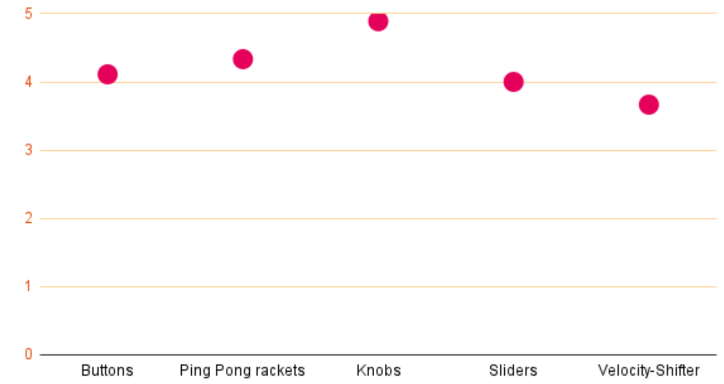
[Fig. 18] Evaluation of the category "Ease of Learning". Own graph.

[Fig. 19] Evaluation of the category "Satisfaction". Own graph.

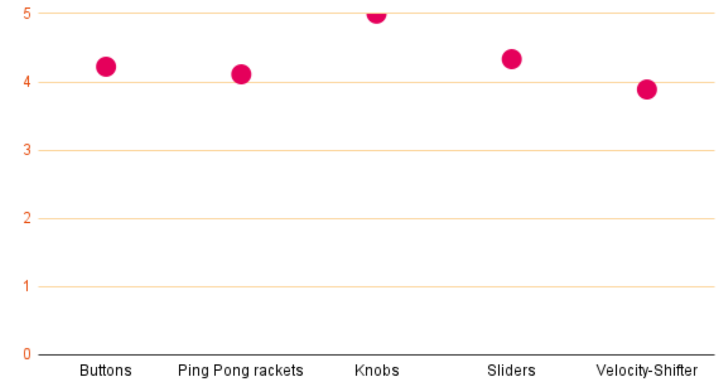
[Fig. 20] Interface with advanced features. Own photograph.

APPENDIX

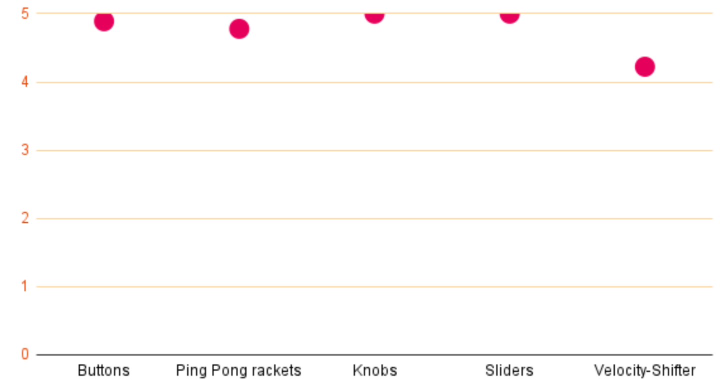
I am satisfied with it.



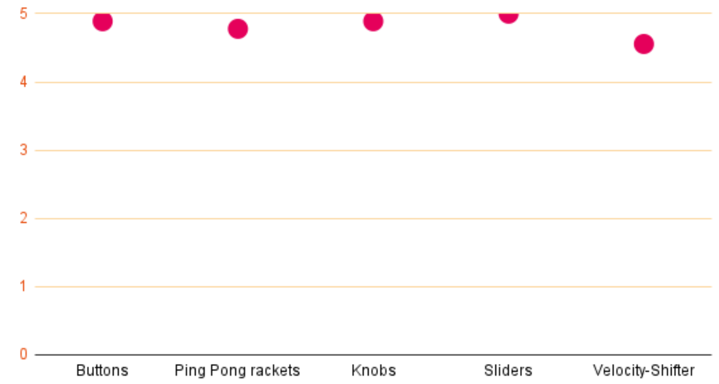
I quickly became skillful with it.



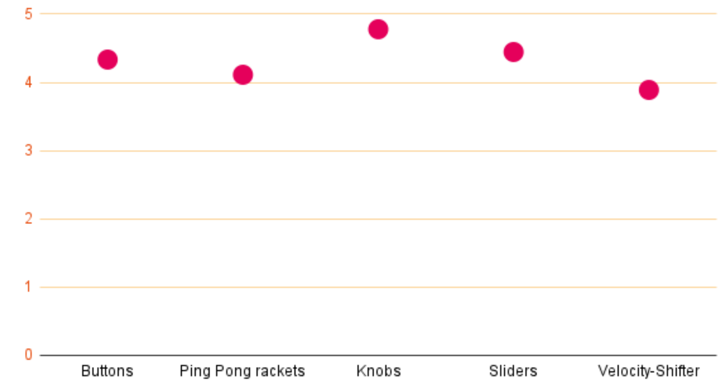
I learned to use it quickly.



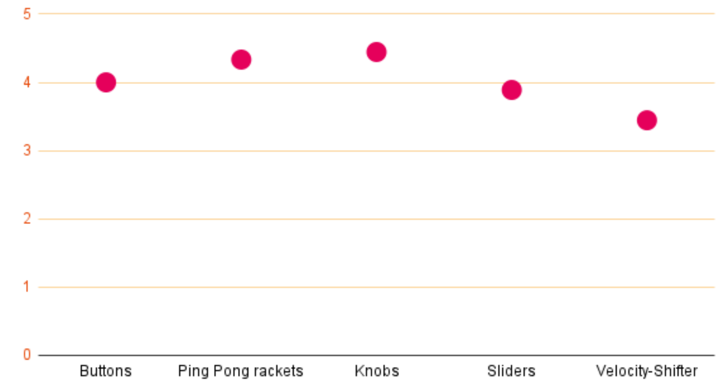
I can use it without written instructions.



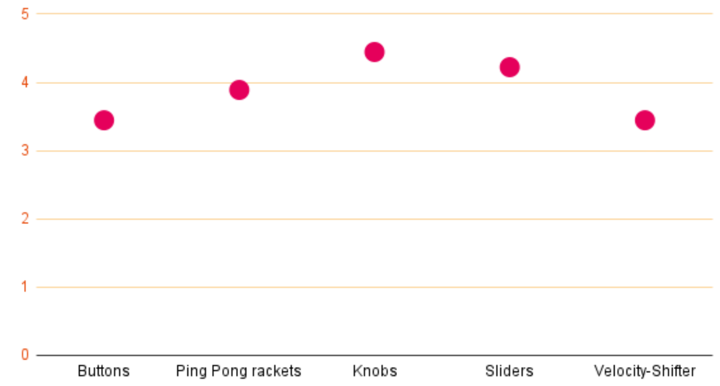
Using it is effortless.



It does everything I would expect it to do.



It meets my needs.



It is useful.