ingenieur
wissenschaften
htw saar

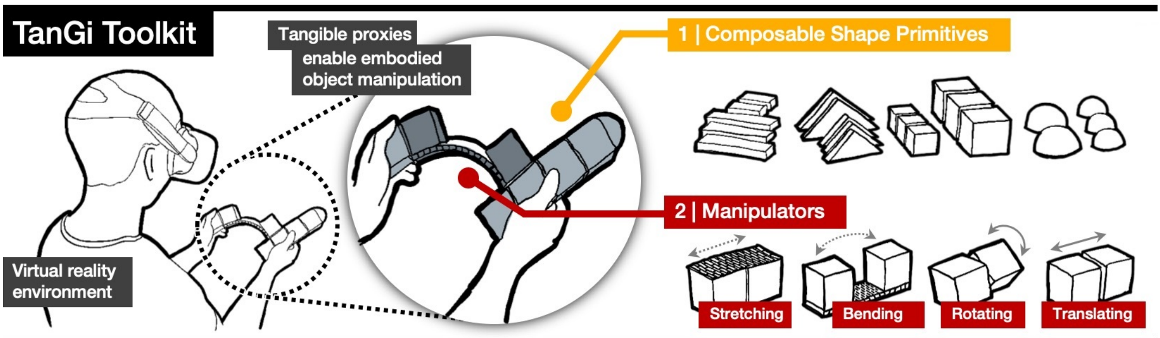
Hochschule für
Technik und Wirtschaft
des Saarlandes
University of
Applied Sciences

STL SYSTEMTECHNIKLABOR

TanGi: A Toolkit for Rapid Creation of Tangible Proxies for Virtual Reality

Martin Feick

Technical Report – STL-TR-2019-03 – ISSN 2364-7167



stl.htwsaar.de

Technische Berichte des Systemtechniklabors (STL) der htw saar
Technical Reports of the System Technology Lab (STL) at htw saar
ISSN 2364-7167

Martin Feick: TanGi: A Toolkit for Rapid Creation of Tangible Proxies for Virtual Reality
Technical report id: STL-TR-2019-03

First published: December 2019
Last revision: October 2019
Internal review: Nicolai Marquardt, André Miede, Scott Bateman, Anthony Tang

For the most recent version of this report see: https://stl.htwsaar.de/

Title image source: Martin Feick

htw saar – Hochschule für Technik und Wirtschaft des Saarlandes (University of Applied Sciences)
Fakultät für Ingenieurwissenschaften (School of Engineering)
STL – Systemtechniklabor (System Technology Lab)
Prof. Dr.-Ing. André Miede (andre.miede@htwsaar.de)
Goebenstraße 40
66117 Saarbrücken, Germany
https://stl.htwsaar.de

TANGI: A TOOLKIT FOR RAPID CREATION OF TANGIBLE PROXIES FOR VIRTUAL REALITY

MARTIN FEICK

Master's Thesis

Applied Computer Science
Faculty of engineering
University of Applied Sciences, Saarland
Hochschule für Technik und Wirtschaft des Saarlandes

Submitted: 31th October 2019

SUPERVISORS:
Prof. Dr. Nicolai Marquardt (University College London)
Prof. Dr. André Miede (Saarland University of Applied Sciences)

CO-SUPERVISORS:
Prof. Dr. Scott Bateman (University of New Brunswick)
Prof. Dr. Anthony Tang (University of Calgary)

LOCATION:
University College London, United Kingdom

TIME FRAME:
April 2019 - September 2019

ABSTRACT

Exploring and manipulating complex virtual objects is challenging due to the existing limitations of conventional controllers and free-hand interaction techniques. We present the TanGi toolkit which enables novices to rapidly build physical proxy objects using *Composable Shape Primitives*. TanGi also provides *Manipulators* allowing users to build objects including movable parts, making them suitable for rich object exploration and manipulation in VR. With a set of different use cases and applications from a wide spectrum, we show the capabilities of the TanGi toolkit, and evaluate its use. In a study with 16 participants, we demonstrate that novices can quickly build physical proxy objects using the *Composable Shape Primitives*, and explore how different levels of object embodiment affect virtual object exploration. In a second study with 12 participants we evaluate TanGi's *Manipulators*, and investigate the effectiveness of embodied interaction. Findings from this study show that TanGi's proxies outperform traditional controllers, and were generally favored by participants.

PUBLICATIONS

Some materials, ideas and figures from this thesis also appear in the following publications:

Martin Feick, Scott Bateman, Anthony Tang, André Miede, and Nicolai Marquardt. TanGi: Tangible Proxies for Embodied Object Exploration and Manipulation in Virtual Reality. (in submission to ACM CHI'20. Hawaii, USA)

„Allow yourself to dream"

Anthony Tang

ACKNOWLEDGMENTS

This thesis was written at the University College London (United Kingdom), which would not have been possible without the support of many people. Therefore, it is my pleasure to thank Professor Nicolai Marquardt and Professor André Miede who made this incredible experience possible. I am grateful for their continuing support during my thesis. I also owe my deepest gratitude to Professor Anthony Tang and Professor Scott Bateman for their intellectual support, their encouragement and their time.

It was an honor for me to be a part of UCLIC with all its amazing people. Working with you made it a unique experience. A special thanks goes to the basement crew, especially to Tu Dinh Duong, Justas Brazauskas, Warren Park and Ethan Wood for their great ideas and helping me with both studies. I would also like to thank Jessi Stark and Karthik Mahadevan for their time, help and insightful comments on this thesis.

Further, I would like to thank my housemates for giving me an introduction to the British culture. Another big thanks goes to Tom Price for being such a great climbing partner, and becoming a good friend.

At my home institute, I would very much like to thank Niko Kleer for his encouraging support during our Master studies. It made it so much easier to have somehow sharing the same goals.

Furthermore, a special thanks goes to Professor Klaus Berberich, Professor Juergen Steimle and Daniel Groeger for giving me the chance continuing HCI research during my Master studies.

Finally, I would like to thank my whole family. Without their support the last five years would not have been possible!

CONTENTS

LIST OF FIGURES

LISTINGS

1 INTRODUCTION

In this chapter we highlight the motivation behind this thesis, and articulate our research questions. Next, we outline the four key contributions of this work. Finally, we explain the structure and we show how the chapters link back to our contributions.

1.1 MOTIVATION & RESEARCH QUESTION

Scope of this thesis

Virtual Reality interfaces will fundamentally change how we design and work with physical objects. VR-based 3D content creation systems allow rapid prototyping of 3D models by using head worn displays and employing 6-DOF controllers. These controllers give designers a type of embodiment in the virtual space, allowing them to move, place and rotate the models using 3D controls. Furthermore, they enable new, intuitive ways to create and engage with 3D objects compared to completing these tasks with traditional 2D interfaces [2, 3, 29, 30, 46].

Problem that we are facing

Despite the improvements offered by 6-DOF controllers to facilitate rapid creation, exploration and manipulation of 3D objects, working with virtual 3D models can still be challenging, because current interfaces are disembodied. For example, a designer creating a new toy relies on controller-based manipulations to move parts of the virtual toy around, and this sort of control-display remapping is cumbersome. The designer cannot feel and easily test out the object through the controllers, and studying how different parts of the toy will behave and react when they are physically manipulated relies on imagination, since controls are not a direct analog for how the toy would really feel.

In this work, we deepen research into how we can give an embodiment to virtual objects, by giving them tangible form and moveable parts that match their virtual counterparts. Recent work has highlighted that providing a physical proxy for virtual objects can facilitate interactions [21, 22, 40, 47, 56]. Our work extends these findings, enabling embodiments to be created for virtual objects by providing a toolkit that allows the creation of tangible proxies – rapidly built physical stand-ins that approximate key elements of both form and function of a virtual object. Our toolkit, called TanGi, enables users to create representations that allow proxy object manipulations, such as bending, stretching, and rotating. The TanGi toolkit provides both composable shape primitives (to approximate the size and shape of the virtual objects), and a representative set of manipulators (which allow multi-part objects to move in relation to one another through rotating, stretching and bending operators). Figure 1 illustrates a proxy object which is assembled using TanGi primitives, and allows for manipulations.

1.2 THESIS OBJECTIVES

Execution of the research project

First, we examined the existing literature to identify the challenges people face when interacting with virtual objects in an environment. Following this, we designed, fabricated and implemented the TanGi toolkit which allows people to quickly build tangible proxy objects that match virtual models. TanGi consists of *Composable Shape Primitives* and *Manipulators*. The Composable Shape Primitives can be combined to rapidly build a variety of objects. The Manipulators allow users to include rotatable, translatable, stretchable and bendable parts in the tangible proxies enabling functional objects that closer match to the virtual counterparts. Through showing different use cases and applications, we demonstrate TanGi's capabilities and expressive power.

Next, we evaluated how object embodiments created with TanGi can affect interactions. To this end, we conducted two lab studies that explored embodied object exploration and manipulation. Our analysis of the first study showed that for reorientation and finding tasks, embodied proxies offered quicker completion times and physical operations that aligned more closely to people's expectations. The second study showed that participants could use the proxies to more quickly and accurately complete matching tasks required manipulating different parts of a proxy. Finally, we discuss findings from our studies, and provide implications for future work.

This work makes four major contributions:

Contributions of this thesis

1. A synthesis of past work on tangible proxies in virtual reality as it relates to embodied interaction.
2. The conceptual design of TanGi, a toolkit that enables embodied object exploration and manipulation in VR.
3. Two user studies showing that physically embodied virtual objects enable richer exploration and manipulation of virtual objects in VR.
4. Different use cases and applications showing the capabilities and expressive power of the TanGi toolkit.

1.3 OVERVIEW

This work is divided into ten chapters which are structured as followed:

Chapter 2 directly addresses contribution (1) providing a view into the related work on tangible interfaces, haptics in virtual reality, toolkit research as well as rapid fabrication. Here, we position our work in the literature, and we articulate the existing problems which have not been addressed in previous work.

Chapter 3 directly addresses contribution (2). Based on the previous chapter, we explain the design of the TanGi toolkit consisting of *Composable Shape Primitives* and *Manipulators*.

Chapter 4 outlines the design and fabrication of the *Composable Shape Primitives*. This chapter further provides information about our prototyping phase and the implementation required to use TanGi proxies in virtual reality.

Chapter 5 addresses the first part of contribution (3). We conducted a user study to understand how embodied exploration using TanGi affects user interaction. We outline the design of the study, provide background information of participants as well as data analysis/collection. Finally, we also elucidate the findings from the study.

Chapter 6 shows the design, fabrication and implementation of the *Manipulators*. This chapter also provides information about prototyping and circuit design of the *Manipulators*.

Chapter 7 addresses the second part of contribution (3). We conducted a second user study to understand how embodied manipulation using TanGi's *Manipulators* affects user interaction. We describe the design of the study, provide background information and we also show the findings from this study.

Chapter 8 addresses contribution (4). Here, we present different use cases and applications to demonstrate TanGi's capabilities, and evaluate its use.

Chapter 9 discusses TanGi for embodied object exploration and manipulation and we show implications for future systems. Furthermore, we outline how TanGi's design and functionality can be improved.

Chapter 10 summarizes the thesis.

2

RELATED WORK

The HCI community uses the term "embodiment" in a number of ways. In this paper, we refer to embodiment in two ways: first, the proxy object gives physical embodiment to the virtual object; second, how people interact with the virtual object thus becomes embodied since interactions with the object are more direct—manipulations on the physical object are mirrored in the virtual world. Therefore, we situate our work within the context of tangible and embodied interaction research, where research has long focused on the cognitive benefits of using tangibles to interact with computation.

2.1 TANGIBLE INTERFACES & EMBODIED INTERACTION

Both free-hand and tangible interfaces for manipulating VR objects have seen considerable popularity in the research literature [29]. Ishii et al.'s conception of tangible interfaces [26] underscored the basic premise of embodied interaction [13]: interfaces that allow people to interact cognitively and physically with information to more fluidly understand the information being manipulated. This concept is echoed by early childhood education psychologists, who promote the use of tangible objects to learn abstract concepts (e.g. math, geometry, etc.), since the theory is that the learning is double encoded through both thought and physical operations [41]. We have seen, for instance, that tangible interfaces promote natural interaction [45], are faster and more intuitive to use [8], because they benefit from human's spatial memory [12].

Why tangible interfaces?

Considerable research in the VR and AR space has also explored the how we might use tangibles objects as physical proxies for what would otherwise be strictly virtual objects [4, 18, 21, 55]. For instance, Hettiarachchi et al. [21] show how an AR system can automatically identify nearby real-world objects that offer the best physical approximation of a virtual object, to be used as a proxy object. They overlay the virtual model onto the physical object providing the best haptic sensation possible, given the available real-world objects. The downside of this approach is that multiple objects with various features need to be nearby, and real-world objects may only roughly match the shape of the virtual counterpart. Simeone et al. [47] investigate the effect of this mismatch between physical proxy and virtual model in their work on substitutional reality. They found that greater mismatches generally hinder the interaction, and particularly pointed

Using real-world objects as proxies

Limitations of real-world objects

out that mismatches are most significant for tactile feedback, temperature and weight differences. Addressing tactile feedback, Cheng and his colleges [9] used a general passive prop and haptic illusion. They redirect the user's hand to match a virtual objects with a location at the physical prop. Hence, it results in haptic feedback when touching the virtual model. Although, this works well for simple one-finger touching an object, but this may not be possible for complex shapes. On the other hand, Muender et al. [40] recently presented results from their study on how different levels of proxy fidelity affect immersion, performance, and intuitive interaction. They compared equal disc-shaped, Lego-built, and 3D-printed tangibles. The results show that higher fidelity leads to better results across all three measurements. However, they also pointed out that Lego offered a good trade-off given its rapid fabrication speed with respect to its overall performance.

Different free-hand interaction techniques

When using free-hand interaction techniques users often perform a grasping gesture to select virtual objects in space [48], and hence can translate and rotate them using their bare hand. However, it is challenging to precisely select and move virtual objects to the desired position and orientation [38]. Therefore, Song et al. [50] proposed the Handle Bar metaphor. Here, users make use of both hands, grasping a virtual handle bar at the opposite ends. The handle bar goes through the virtual object, and thus represents the rotation axis of the object. A downside of their approach is that users always need both hands to manipulate a virtual object. In contrast, Mendes et al. [38] allowed users to customize the rotation axis by using a single controller or their hand. They separated the degrees of freedom introducing different modes. For instance, grabbing inside the object enabled 3DoF translation, whereas grabbing outside entered the 3DoF rotation mode. The different approaches for virtual object manipulation can be classified into one- and two-handed techniques, which both have treads-offs with respect to the task [14].

2.2 HAPTICS IN VR

Different kinds of haptic devices

"Physicalizing" the VR experience has long been a central premise of haptics research, where the goal has been to make virtual objects touchable by applying forces, vibration or motions to the user. Providing haptic sensation for virtual models frequently requires unwieldy or bulky hardware . Various devices create different haptic sensations including rendering the shape of physical objects [7, 36, 37], providing force-feedback [20], or dynamic weight-shifting [54]. Some haptic devices overcome this with wearables that simulate weight and grasping [10] using electrical muscle stimulation [33, 34]. Robots can provide physical props for a virtual environment [52], and drones have been used to provide haptic feedback when interacting with virtual models [1, 25]. Similarly, shape-changing interfaces are promising, but can be bulky [16]. Haptic feedback has also been shown to improve immersion in virtual reality-based navigation tasks, since the navigation becomes an embodied task [27].

TanGi builds on the idea of embodiments by providing real-world physical proxies for manipulating virtual models, and extends this idea beyond composable primitives (e.g., Muender et al. [40] use Lego blocks) by adding manipulators that allow the proxies to be multi-part objects that move in relation to one another.

2.3 TOOLKIT RESEARCH & RAPID FABRICATION

Different approaches to create tangible proxy objects

Tangible proxies have been demonstrated to be powerful way to interact with virtual [22, 40, 45, 56]. Contrary, having a physical proxy objects around that perfectly match virtual objects is likely not possible. Therefore, different approaches have been explored allowing users to quickly design and build tangible proxies. 3D printing opened-up many possibilities, however it is time consuming to fully print objects. Mueller et al. [39] proposed faBrickation to significantly reduce the time needed to fabricate a physical object. Their approach utilizes a combination of 3D printing and Lego bricks to build functional objects over two times faster than traditional 3D printing. To further improve the fabrication speed, re-usability, flexibility and modularity toolkits have been proposed. For instance, the HapTwist [56] toolkit features a Rubik's Twist which consists of unified parts connected via twistable joints. It offers great robustness; however, it does not allow to replicate moveable object parts. Real-world objects have a vast complexity i.e. rotatable, bendable, stretchable and translatable parts. Toolkits ought to minimize these difference [31, 47] allowing users to build richer tangible proxy objects. Recently, the VirtualBricks [5] toolkit has been presented enabling users to integrate e.g. translatable and rotatable parts into proxy objects. They presented a variety of applications demonstrating the capabilities of objects with moveable parts, but did not evaluate it through user studies.

What do we contribute?

Our toolkit share similarities with HapTwist [56] and VirtualBricks [5], but extends the idea of manipulable parts by introducing two new types of manipulations (i.e., variable linear stretching and unidirectional bending) and various composable shape primitives. Further, we provide a first evaluation providing clear results that proxies better support exploration and manipulation interactions, when compared to conventional controllers.

3 DESIGN AND CONCEPT OF THE TANGI TOOLKIT

Why another toolkit?

Similar to earlier work, we are motivated by the need of providing designers with the ability to rapidly prototype physical proxies that can enable embodied exploration and manipulation. Our approach relies on Composable Shape Primitives, which allow rough tangible proxies to be constructed quickly, and Manipulators, which allow multi-part objects to be composed with moving parts. Together, these enable embodied exploration by matching the tangible proxy to the virtual object, and embodied manipulation by allowing the tangible proxy to control the virtual object. In our vision TanGi is extendable. Thus, designers can create customized shape primitives which meet their own requirements, and subsequently they can re-use them.

TanGi's design goals

The TanGi toolkit philosophy was driven by three philosophical goals. First, the toolkit should enable rapid iterative prototyping with very quick turnaround (<5 mins). Second, the proxies made with the toolkit should enable exploration of corresponding virtual objects. Third, the proxies should allow people to manipulate the virtual objects.

3.1 COMPOSABLE SHAPE PRIMITIVES

Whereas others try to solve the exploration problem by either repurposing real-world objects [21], 3D printing techniques [39] or through robot assemblies [19, 52, 55] our approach relies on Composable Shape Primitives, which allow people to create proxies that approximate the virtual object.

In the first version of this toolkit we provide four primitive shapes at three different sizes: cubes, triangles, half-spheres and sticks (see

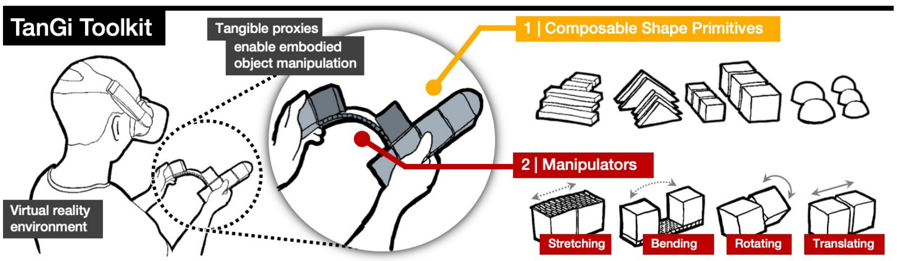


Figure 1: TanGi's design space: Composable Shape Primitives and Manipulators. A user interacts with an assembled TanGi proxy consisting of Composable Shape Primitives and a bending Manipulator.

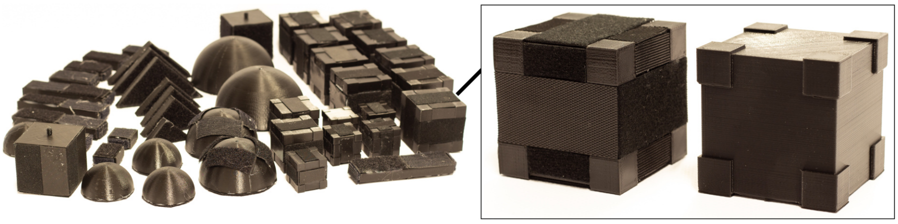


Figure 2: TanGi's Composable Shape Primitives (left) and 50mm base cubes showing ablated areas and Velcro tape pattern (right).

Design of the Composable Shape Primitives

Figure 2 left). We decided on these primitive shapes after a formative prototyping phase with foam board. These shapes can be composed into larger composite objects using heavy-duty Velcro tape with a specific pattern (see Figure 2 right). As illustrated in Figure 2, the primitives allow us to replicate a variety of basketball-sized objects. Our implementation relies on 3D printing to fabricate the shapes, and a specific Velcro-pattern (see Figure 2 right) on the cubes that provide a stable base atop which additional shapes can be applied. These proxies can thus be composed of reusable primitives that can be built up and taken apart to represent various virtual objects as necessary. This approach is similar to the often-used LEGO blocks [5, 40]. Going beyond using traditional brick structures, TanGi can provide a richer set of shapes primitives and can be easily extended with by adding new 3D-printed primitives when necessary. When combined with a 3D tracker (in our current version, a Vive Tacker) objects composed with TanGi can function as a tangible proxy that can be used to control the movement and orientation of a corresponding virtual object. This allows people to engage in embodied exploration, moving, feeling, reorienting and grabbing approximation of different parts of the virtual object.

3.2 MANIPULATORS

Why do we need the Manipulators?

Physical objects have vast complexity such as rotating parts, can be stretched, folded, deformed, bended etc. TanGi provides a representative set of Manipulators that allow multi-part objects to move in relation to one another, in an effort to minimized the difference between physical proxies and their virtual counterparts (as suggested by [47]). While the entire range of manipulations that are possible with a physical/virtual object is beyond the scope of this work, we developed TanGi with the goal of incorporating a larger set of representatives of manipulations than has been done in previous work. TanGi Manipulators replace the previously described Velcro connectors between shape primitives with new manipulable blocks. Manipulators allow for a movement relationship (i.e. rotation, translation, stretching, bending) between shape primitive to be tracked. These

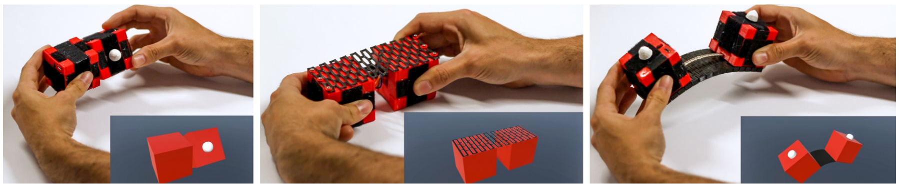


Figure 3: TanGi's Manipulators: Enable rich object manipulations such as rotations (left), translations and stretching (middle) as well as bending (right).

movements can then mapped to the virtual object, allowing parts of the virtual object to be controlled (see Figure 3).

In this first version of the toolkit, we focused on four movement primitives, which we describe below. We expand on variable linear stretching and unidirectional bending, since these are new contributions of our work.

Single-axis rotation
Enables objects to have rotational parts (e.g. bottle lid) through using a rotary potentiometer (see Figure 3 left).

Linear translation
Parts of an object can be moved back and forth in one direction (e.g. linear sliders). This manipulator utilizes a linear potentiometer (see Figure 3 middle without stretching pattern on top).

Variable linear stretching
Extends linear translation by providing a better sense of how much parts of the object can be translated in order to communicate min/-max states. As a result of the increasing amount of force needed to stretch the object (e.g. to cock a crossbow). It uses the same hardware as the linear translation manipulator; however, it uses a variable 3D printed stretchable material on top, which provides force-feedback. Following TanGi's modular approach the stretching patterns can be replaced. Thus, users can choose between less stretchable (more force required) or more stretchable (less force required) pattern to create different haptic sensations (see Figure 3 middle).

Unidirectional bending
Enables objects that have bendable parts, such as a fishing rod. It also naturally communicates min/max states. To achieve this, we use an Adafruit[1] bend sensor between two distant cubes. Similar to the stretching pattern we utilize a bending pattern between the cubes. Depending on the 3D printed pattern users can create a less/ more bendable object. In the default position the bend manipulator is straight

1 https://www.adafruit.com/

(see Figure 3 right).

In the next chapter, we discuss the development and fabrication process of the TanGi's Composable Shape Primitives.

4 TANGI'S COMPOSABLE SHAPE PRIMITIVES

This chapter covers the background behind TanGi's *Composable Shape Primitives*. It also briefly reports on our prototyping phase providing insights into the development of TanGi. Next, we report on how we fabricated the Composable Shape Primitives. Furthermore, we outline the required steps for re-implementing the virtual reality prototype. Finally, we summarize design, fabrication, and implementation of TanGi's Composable Shape Primitives.

4.1 PROTOTYPING

Here, we briefly describe our prototyping phase as well as how we developed the toolkit shapes.

To determine a initial set of shape primitives which is suitable for re-constructing a variety of different hand-held objects, we started a prototyping phase. Here, we made use of foam board allowing us to quickly prototype and evaluate different shapes. We also included our lab fellows in this process in order to receive early feedback. With their help we quickly generated many ideas, and after a first selection we built the initial prototypes. Figure 4 shows the foam board prototypes that we created.

The essential ideas of toolkits are modularity and re-usability. However, this requires a way to dis-/assemble objects multiple times. Our first approach to tackle this problem was to utilize magnets. We experimented with magnets in different sizes, shapes and magnitudes. Neither embedding them into the primitive shapes nor attaching them on the surface elicited the desired outcome. In fact, they make the object heavier, and moreover limit the user due to the magnetic north and

Why no magnets?

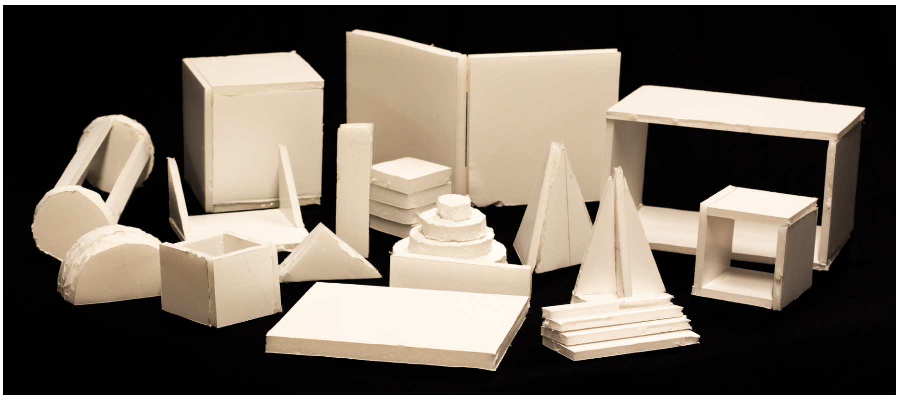


Figure 4: Foam board prototypes.

south pole. Thus, they can only be used contrarily. Further findings from this phase are: (1) Shapes with strong embedded magnets are challenging to assemble, (2) they may interfere with hardware components, and (3) the primitives can not be placed in close proximity, because they attract each other.

In the next iteration we decided to use heavy-duty Velcro tape. To allow more freedom in connecting the primitive shapes we arranged the Velcro tape in a specific pattern allowing users to connect multiple cubes as they wish. Additional strips of the counterpart are placed alongside the main lane (see Figure 2 right) in order to stabilize connected parts.

4.2 FABRICATION

How to 3D print the shape primitives

After the prototyping phase with foam board we designed a basic set of primitive shapes in Rhino3D[1] (Version 6 SR14). Rhino3D is a commercial 3D computer-aided design (CAD) and computer graphics software which is widely used to design and create 3D models. The models were exported as stereolithography (*.stl*) files, and eventually printed on a fused deposition modeling (FDM) 3D printer using the Cura 4.0.0 software with an Ultimaker 2+ as well as an Ultimaker 3. We used black 2.85 millimetre (mm) diameter Polylactic Acid (PLA) filament with a 0.4 mm nozzle and the default 0.1 mm layer height profile on both printers. Using gradual infill results in light-weight objects, increases printing speed and uses less material. Neither adhesion nor support material was used to fabricate the objects. As visible in Figure 2 right the design offers ablated areas of 1 mm to accommodate heavy-duty Velcro tape. To attach the Velcro tape to the primitive shapes we simply used super glue. All four primitive shapes were fabricated in three different sizes e.g. the cube in 50 mm, 40 mm and 30 mm. For example, the biggest 50 mm cube required 31.5 cm of Velcro tape, it took 5 hours and 24 minutes to print, required 3.48 meters of filament and weights 27 grams according to the 3D printer we used. Overall, we fabricated 56 objects.

All created Rhino3D files for this toolkit are open-source and can be downloaded from our GitHub repository[2].

4.3 IMPLEMENTATION

In this section we provide a detailed look into the implementation of our virtual reality prototype. First, we give a general overview of the tracking system and its architecture. Subsequently, we elucidate and discuss the implementation of the testbed which we used in both

1 https://www.rhino3d.com/
2 https://github.com/MartinFk/TanGi

studies. We show the study execution program and its functionality, as well as the internal data logging which also offers a data export.

System Overview

We designed and implemented the prototype in Unity3D[3] (version 2018.3.11f1) using an HTC Vive[4] virtual reality system (model 2PR8100) with SteamVR[5] (version 1.5.15) and the OpenVR SDK[6] (version 1.4.18). For the hand tracking we used a Leap Motion sensor[7] (Orion SDK version 2.3.1) attached to the HTC Vive with the help of a 3D printed mount (see Figure 5). The system was running on a Dell Notebook with an Intel Core i7 – 7700 HQ CPU, 16 GB of RAM and an NVIDIDA GeForce GTX 1060. We used a standard Vive tracker (model 2PYV200) attached to the toolkit object in order to track its position and orientation in three dimensional space. The HTC Vive tracking system does not allow to pair more than two controllers with the headset. To overcome this limitation we used two USB Dongles enabling us to pair and subsequently track two conventional controllers as well as two additional objects simultaneously. Following HTC's website, we placed the Vive tracking cameras so that they faced each other in order to ensure optimal and robust tracking results.

Hardware platform we used

We used the default virtual reality environment in Unity3D and C# scripting to program *GameObject's* behaviour. A Unity scene is composed of gameobjects which again can include multiple gameobjects or different components such as scripts. Here, we explain the required steps to set up a virtual reality scene using HTC Vive, and Leap Motion for hand tracking. To do so, import the Leap Motion Unity Core Assets (v. 4.4.0) and the SteamVR asset package into the project. Place the [SteamVR], [CameraRig], and [Leap Rig] Prefabs in the world origin. *Note:* Prefabs are templates which represent gameobjects with all their components and property values[8]. For instance, the [CameraRig] provides all the basic scripts for tracking the headset as well as the Vive controller. To allow users to interact with virtual objects we have to modify the [LeapRig] Prefab. First of all, this requires the Leap Motion Interaction Engine (v. 1.2.0) asset which can be downloaded from the Leap Motion website[9]. Next, create an InteractionManager gameobject and attach the InteractionManager script to it. Furthermore, it must derive from the [LeapRig] Prefab. The Manager implements the InteractionHand (Left) and InteractionHand (Right) Prefabs. Virtual objects must include a Rigidbody component, implement the InteractionBehaviour script and derive from the [LeapRig]

How to set up the virtual reality system

3 https://unity.com/
4 https://www.vive.com/uk/
5 https://store.steampowered.com/steamvr
6 https://github.com/ValveSoftware/openvr
7 https://www.leapmotion.com/
8 https://docs.unity3d.com/Manual/Prefabs.html
9 https://developer.leapmotion.com/releases/interaction-engine-120

Prefab. Thus, users can interact with virtual models enabling them to throw, catch, rotate and translate these virtual models using their bare hands.

Study modification

In our first study, described in Chapter 5, participants were required to translate and orient a 3D object so that it matches the position and orientation of a second object. They used four different interaction techniques to do that; a Free-hand interaction technique, a Vive controller to manipulate the virtual object and two physical objects that we tracked. In order to ensure equal conditions across the different interaction techniques we modified our system. Following, we use a concrete example case to illustrate how Free-hand and Vive controller virtual object interaction work. Furthermore, we explain how we track real-world objects in space, and map them to their virtual counterpart. Let us assume that we only want to interact with a simple base cube (50x50 mm) in virtual reality using a Free-hand interaction technique, a Vive controller and a tracked real-world object. However, the steps described below work for any arbitrary virtual 3D model (e.g. *.stl* or *.obj* model).

Required modifications for our user study

Free-hand

When interacting with virtual models they aim to behave similar to real-world objects. This means that people can drop them, for example. This often happens by unintentionally touching them. To avoid this we restricted possible object interactions. For instance, people could only manipulate the object when grasping it. Since the study task only required 6DoF (degrees of freedom) manipulations, collisions with other objects were disabled. Moreover, we used zero gravity and hence people could neither accidentally drop the object nor throw it away (due to applying force).

Controller and real-world objects

From an implementation perspective both conditions work similarly. By creating a new gameobject, placing it in the world origin and attaching the SteamVRTrackedObject script, the gameobject receives coordinate updates from paired Vive controller or Vive tracker. The 3D model derives from this gameobject and thus also receives updates.

Study testbed

Here, we describe the testbed we implemented in order to execute both studies. It also performs the data tracking and automatically exports it as a *.csv* file for further analysis. Once the testbed is started the program executes the study and runs through the conditions successively. The testbed expects an input file (.txt) providing all necessary information for executing the study such as condition order, num-

Study testbed utility

ber of participants, number of tasks etc. Based on this information it enables the relevant gameobjects and components using Unity's tag referencing system. All other components are disabled until they are needed.

Taking part in a study can be overwhelming for several reasons. Some people might have never used virtual reality before or it is the first time that they participate in an experiment in general. Therefore, the testbed offers a practise mode, where participants can familiarize themselves with the study task and virtual reality. Furthermore, it provides visual cues guiding people through the study. The messages and pictures were also dis-/enabled when necessary using Unity's tag system. To display a dwell timer (progress bar) as well as study instructions we used different canvases. They derive from the Camera (head) gameobject in the [CameraRig] prefab. Displaying text messages can easily be achieved by utilizing a rectangular transform and adding a Text component to it. To show a progress bar we attached an Image component, and thus the virtual reality system can display a non-interactive image to the user. The image was a white circle on a transparent background. By default, the Image component offers various settings that can be used to manipulate the an image. The configurations for creating a clockwise progress bar, as visible in Figure 7 are the following. Image type needs to be set to filled, fill method to radial 360, fill origin to bottom and we set colour to yellow. The fill amount can be manually set to a fixed value, however our progress bar was dynamically and therefore an additional script was used to adjust the fill amount with respect to the remaining time. Essentially, it is a timer which maps a time interval (two seconds) to the progress bar. Once, the user matches the object position or pointed at the right location the script triggers the progress bar. Leaving the threshold led to resetting the timer, and hence also reset the progress bar.

Displaying instructions

Next, we explain the *CheckAlignment()* method which determines object matching. This method is called once every frame and checks whether the user's object matches the goal object in position and orientation within a threshold. As a result, the *ObjectsMatch()* method is called which starts a timer (including progress bar and changing object colour). The participant is required to hold the object for two seconds, after that the *ObjectsMatch()* method processes the tracked data. Hence, the task is finished, and the main program continues with the next task. The testbed offers data tracking methods for various parameters, including a function to export tracked data as a .csv file for further analysis. To do so, we used the CsvHelper .NET library [11] which is an open-source project allowing users to read and write in csv files. It offers great flexibility and is easy to use. By implementing a mapper class, the data structure and the headers for the .csv file can be determined.

Detect object matching

4.4 SUMMARY

In this chapter, we described the prototyping phase of TanGi's Composable Shape Primitives. We reported on our lessons learned, and provided all necessary details to fabricate the shape primitives. Further, we described our tracking and virtual environment setup using HTC Vive, enabling users to use their own assembled tangible objects in virtual reality. In the next chapter we evaluate TanGi's Composable Shape Primitives. Therefore, we described the study testbed as well as the study modifications in the last section.

5 STUDY 1 - EMBODIED OBJECT EXPLORATION

This chapter elaborates on the first lab study. We conducted this study to understand how embodied exploration using TanGi affects user interaction. Here, we describe the design of the study, provide background information about participants as well as we explain our data analysis and collection process. Finally, we report findings from this study.

5.1 DESIGN STUDY 1

Why did we design the study?

Our first study explores how different control types, demonstrating a range of different levels of embodiedness, affect virtual object exploration. Our examples (described above) demonstrate that TanGi does allow building a wide range of proxies for virtual objects, but we wanted to understand the impact of proxies on basic interactions with virtual object (such as reorienting them to get a different view, or interacting with them through natural gestures). To do this, we conducted a controlled lab study where participants re-oriented a virtual object to a pre-specified target orientation and pointed at a target on the virtual object (to represent a simple interaction). Participants compared four different control mechanisms, each with a progressing level of embodiment: (1) Free-hand control that approximates natural gesture-based control using a Leap Motion; (2) 6 DoF-controller using a Vive controller; (3) TanGi proxy, which functions as an approximation of the virtual object; and, (4) a high-fidelity 3D print that acts as an exact replica of the virtual object.

Participants

Participant information

We recruited 16 participants (seven reported as females; eight reported as males; one preferred not to answer), aged 20-38 (avg: 25.75; sd: 4.5) from the general public and the local university. Participants had a range of different educational and professional backgrounds including engineering, computer science, psychology, chemistry, robotics, music composition, law and modern languages. Participants were given a small 3D printed model and a sweet as a token for their participation. The experiment took about 45 minutes. Two participants had never used VR before, twelve had used it a few times (one to five times a year), one person used it often (6 - 10 times a year), and one other person on a regular basis (more than 10 times a year). The study has been approved by the University College London's Ethics Committee.

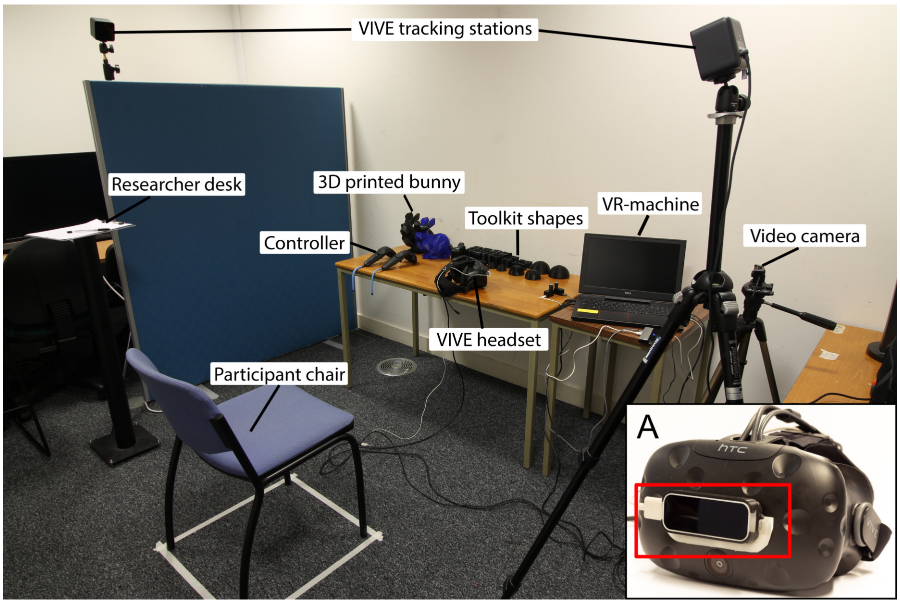


Figure 5: Study space: Showing the tools we used in our study. (A) show the HTC Vive with a 3D printed mount hosting the leap motion devices.

Procedure

Our study used a within-subjects design, allowing participants to explore and compare the different control types. A Latin-square design was used in order to counterbalance the four condition. The study was conducted in a quiet room to avoid distraction and ensuring the same testing conditions for all participants (see Figure 5).

After a study introduction and informed consent, participants performed a practice round in VR, giving them an opportunity to familiarize themselves with VR, the study task and the system. When participants felt comfortable, the study began. In the first part of the study, participants were asked to reconstruct the Stanford bunny[1] using the shape primitives available in TanGi. As a reference, a physical 3D printed version of the bunny was provided.

After completing the first part of the study, participants were provided a demographic questionnaire regarding their prior experience and background. Next, they performed a test, to collect data regarding their mental rotation abilities. Finally, they executed the matching task using four different techniques, followed by a final questionnaire as well as a semi-structured interview to better understand their experience.

1 http://graphics.stanford.edu/data/3Dscanrep/

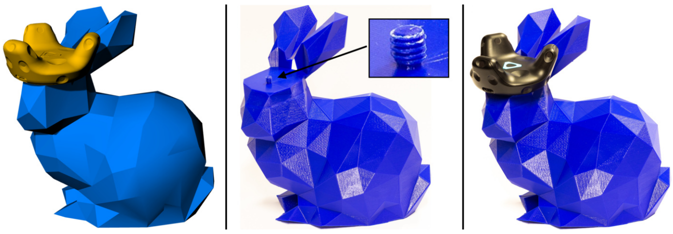


Figure 6: Study 1 object (left to right): Modified virtual 3D bunny with Vive tracker as a grown, 3D printed bunny with screw on top for hosting a Vive tracker, and final result showing 3D printed bunny with actual Vive tracker as used in study 1.

Task Design

We chose the 3D printed Stanford bunny for our study, because (1) it has a distinct unique shape, and (2) it has many details such as ears, tail, nose, etc. Following we describe the two study parts.

Participants assembled an object using TanGi

Part 1: The first part of the study aimed to evaluate the capabilities of our toolkit to approximate and relatively detailed object, and to help us to understand how novices', with no previous experience in this type of proxy creation, approach such tasks. We asked participant to assemble the bunny using our toolkit. There were no constraints given except that the cube with the tracker was required to be the head of the bunny, and therefore was 3D printed with a ¼ inch screw on top. We only offered two different primitive shapes (cubes and half-sphere), each in three different sizes. In our pilot study, we found these shapes were surprisingly sufficient for creating an approximation of the bunny, and put a reasonable cap on the task complexity.

Object matching task

Part 2: The task in part 2 models a common operation in a VR world: reorienting an object to locate a particular view and to interact with the object. Our experimental system generates pseudo-random locations on the bunny (red spheres) that indicated where participant needed to find and interact with (through pointing). Subjects were required to alternate between position matching and pointing interactions, and hold a particular position or pointing position for two seconds to complete the task. Figure 7 provides an illustration of the

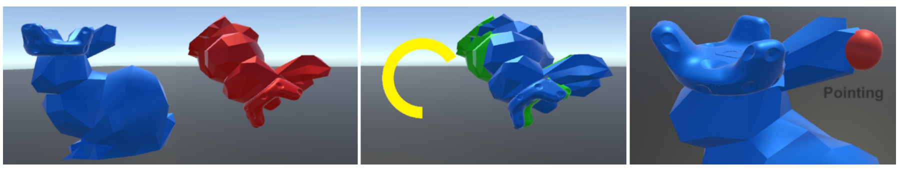


Figure 7: Study 1 task: Object-matching task. The blue bunny (left) is required to match the red's orientation and position. Yellow progress bar and green object color indicate matching. Next, participants point at locations - red sphere on the object.

Figure 8: Study 1 conditions: Increasing levels of proxy embodiment in study 1.

task. Each participant completed ten different orientations and ten pointing locations per condition. For this part of the study we used the testbed described in Section 4.3.

Apparatus
Participants were required to match ten different locations which were hardcoded to ensure that they are always reachable for sitting participants. The required end-locations covered a space of 100x30x25 cm (WxLxD) in front of the participant. The pointing locations were randomly selected from a set of five (nose, body, tail, paw and ear of the bunny). We used two standard Vive trackers (2PYV200) for the 3D printed bunny as well as the assembly. To provide support during the task, we displayed a dwell time indicator (during the two-second hold required to complete the task) using a yellow progress bar (see Figure 7 middle). After pilot testing, we chose a rotation threshold of 30 degrees across all three axes, and an overall threshold of 6cm for positioning. Once a participant entered that threshold, the goal bunny turned green and the progress bar started (see Figure 7).

Fabrication
For the study we fabricated one special cube which hosted the VIVE tracker, and therefore was 3D printed with a ¼ inch screw on top. We downloaded the *.stl* models for the screw and the low-poly Stanford bunny from Thingiverse[2], and modified them for our experiment. Thus, we could place the VIVE tracker on the bunny's head. Zhu et al. [56] found that not visualizing the VIVE tracker in virtual reality confused users and slowed them down. Therefore, we also created a CAD model including the VIVE tracker on the bunny's head to address their findings. Finally, all components were printed on an FDM 3D printer Ultimaker[3] 2+ and Ultimaker 3 using black and blue PLA.

Data Collection
We collected data from seven sources: a pre-study questionnaire for demographic information; a mental-rotation test using PsyToolkit [51];

2 https://www.thingiverse.com/
3 https://ultimaker.com/

Figure 9: Sample proxies study 1: Participant 1 built bunny proxy including many details (left), whereas P9 built a bunny that matches shape/-size (right).

video of the participant as they completed phase 1 and phase 2 of the experiment; internal logging (e.g. task completion times, accuracy, travelled position/orientation, head movement, head gaze etc.), field notes and observations; a post-study questionnaire and a short semi-structured interview to better understand participants' experiences in the different conditions.

Analysis

We conducted a statistical analysis (collected 7400 data points) following Scott MacKenzie's [35], and related this to the results from our thematic analysis where we identified recurring themes in participant behaviour as they engaged with the system. In addition, we conducted a modified interaction analysis (Jordan & Henderson [28]), where we looked at unusual incidents to provide further insights into how people used the different techniques. The statistical analysis was done using GoStats [35] and RStudio[4].

Pilot study

We ran a pilot study with four participants to clarify the instructions and to ensure that our method is viable [42].

5.2 RESULTS & FINDINGS STUDY 1

Here, we show the findings from our two-part experiment. We start with part one where participants were asked to build a rough approximation of the bunny using our toolkit.

Part 1: Building the proxy object.

All participants successfully built proxy objects

All participants successfully assembled a bunny using our toolkit. Two participants reported that it was *"...tricky to match the Velcro tape"* (P11), and suggested that *"...different colors might help"* (P11). However, generally subjects reported that it was easy to build the

4 https://www.rstudio.com/

object; (median: 6.0; sd: 1.15) on a 7-point Likert-type scale to the question: *"It was easy to assemble the object"* (1-strongly disagree; 7-strongly agree). On average it took participants 2:47 min. (sd: 44 sec.) to complete their assembly. Participants built 16 different unique bunnies; Figure 26 showing all 16 bunnies can be found in the appendix. Four participants pointed out that they would have liked additional shapes such as triangles (P2, P7, P11, P16); the rating on; *"All necessary shapes were provided for building the object"* was (median: 5.5; sd: 1.93). Generally, participants told us that they were satisfied with the result (median: 6.0; sd: 1.59). The instructions were always the same and the practice round provided participants with the context of why and for what purpose they needed to build the bunny. This also aligns with the interview data where we asked participants why they decided to build their proxy in that way. P7 responded: *"I just tried to roughly match the size"* whereas P16 stated: *"The bunny needs ears!"* showing they wanted to re-create this detail. The bunny broke towards the end of the second part of the study for two participants: *"My object was severely hampered by its head falling off"* (P10). Other participants also reported issues regarding the robustness of the bunny while manipulating it. This was mentioned in particular when more detailed features were included in the model (.e.g., a ear, nose or tail). But despite these concerns, having only two instances of a proxy breaking, did not prevent participants from completing the task. We discuss alternative construction techniques in the discussion section that would address some of the robustness issues, but in general the possibility of breaking a proxy object is a block-like construction kit (including Lego-based toolkits).

Part 2: Orienting and Interacting.
Participant used their own object that they built in Part 1 as the TanGi condition, for the second part of the study, and all participants completed the second part of the experiment. Our analysis of the mental rotation test did not show any outliers. First, we report on the general results before we dive into the different conditions. Overall, subjects found the different conditions *"easy to learn"*: 3Dprint (7.0), Controller (6.0), TanGi (6.0), Free-hand (6.0); and, "easy to use" 3Dprint (7.0), Controller (6.5), TanGi (5.5), Free-hand (5.5) (median scores). Generally, the 3D printed bunny was the fastest in terms of task completion time. Means for the four conditions were: 3D print (mean: 3.9 sec.; sd: 1.2 sec.), Controller (mean: 4.2 sec.; sd: 1.4 sec.), TanGi (mean: 5.8 sec.; sd: 1.7 sec.), and Free-hand (mean: 10.7 sec.; sd: 2.3 sec.).

Findings from the matching task

To further investigate our data, we ran One-Way repeated-measures ANOVAs. The collected data sets hold the homogeneity assumption, because they are normally distributed verified through Lilliefors normality tests. Main effects revealed by the ANOVA were tested for significance using post-hoc Bonferroni-Dunn tests.

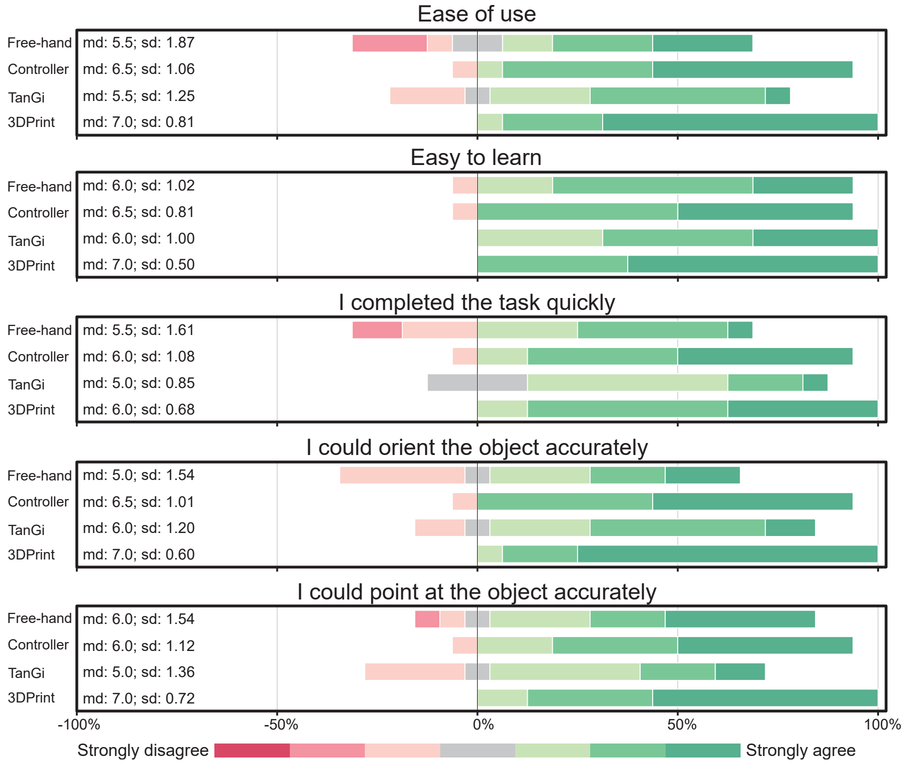


Figure 10: Post-study questionnaire results on a 7-point Likert-type scale (1= Strongly disagree; 7=Strongly agree).

We found a main effect on task completion times ($F_{4, 45} = 130.9$, $p < .0001$). Following this, we found a significant difference between Virtual and the three other conditions as well as between Toolkit and the 3D print at $p < 0.05$. This is also supported by participants' ratings to *"I completed the task quickly"* (medians: 3Dprint (6.0), Controller (6.0), Toolkit (5.0), and Free-hand (5.0)).

In terms of accuracy we saw similar results. Average error values in degrees across the three rotation axes were: 3D print (mean: 12.7 degrees; sd: 3.1 degrees), Controller (mean: 12.0 degrees; sd: 2.5 degrees), TanGi (mean: 13.7 degrees; sd: 3.3 degrees), and Free-hand (mean: 16.4 degrees; sd: 2.6 degrees). Translation error values along

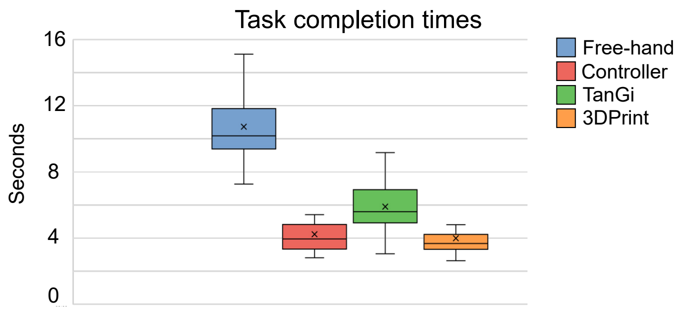


Figure 11: Task completion times study 1.

Figure 12: Twists/bends arm to match goal bunny.

x, y, and z in sum were 3D print (mean: 2.8 centimeter (cm); sd: 0.7 cm), Controller (mean: 2.7 cm; sd: 0.5 cm), TanGi (mean: 3.2 cm; sd: 0.8 cm), and Free-hand (mean: 3.5 cm; sd: 0.5 cm). We found a main effect for the orientation offsets ($F_{3,45} = 20.279$, $p < .0001$). Post hoc tests showed a significant difference between Free-hand and the three other conditions at $p < .05$. The ANOVA for translation difference indicated a main effect ($F_{3,45} = 7.865$, $p < .0005$); however, post hoc showed no significant differences after corrections. Participants' ratings align with these findings *"I could orient the object accurately"* (medians: 3Dprint (7.0), Controller (6.5), TanGi (6.0), and Free-hand (5.0)).

Observations

Free-hand.

Without tangible elements it was significantly harder to manipulate the object. We frequently observed that participants were not aware of their grasping point. As with real world objects the grasping point simultaneously represents the rotation axis. Grasping the bunny at the ear resulted in an unexpected large rotation for participants. Contrary, P11 and P16 favored the virtual condition. *"This is magical... I am not afraid to drop stuff"* (P11) or *"I can just arrange it how I want"* (P16).

Observations from the different conditions

Controller.

The Controller performed as we expected. It provides an easy tangible way to manipulate virtual objects. Subjects reported that it was comfortable to hold and allowed them to easily match the goal orientation. We often observed that rather than changing the grasping position, participants twisted and bended their wrist to rotate the object.

TanGi toolkit.

Participants were deliberately slower with the TanGi proxies, as they were worried the components might not stay together. In spite of this, participants performed well using their own proxy. Compared to the

Free-Hand and Controller conditions, it allowed them to *"... better understand the size/dimensions of the object"* (P1), *"... because it was closer to what I am holding"* (P10). Participants stated that they used physical parts of the object as landmarks being able to quickly determine the object's orientation: *"I used the tail and the ears so that I roughly know how it is oriented, and it helped me to find the correct pointing location"* (P9). These observations make it clear the proxy functions as an embodied stand-in for the virtual model. This kind of stand-in would be appropriate, opined P8, particularly for *"objects that are challenging to understand in VR, because of the environment, task, rendering, complexity etc. [The proxy] would [allow] my hands to better understand it"* (P8).
One challenge we observed with TanGi proxies was that mismatches between the virtual model and the TanGi proxies caused some confusion – it *"... slows me down, because I need [a mental] model of my physical object while working with a different virtual representation"* (P8). In some cases, we observed that participants overshot the pointing location (i.e. pointed into the model rather than on the surface), because they expected to receive tactile feedback about the edge of the virtual model. This would occur, for instance, when parts of the bunny were not replicated in the proxy (e.g. the ears), and tried to touch the tip of the ear. These mismatches slowed participants down, consistent with prior literature [31, 47].

3Dprint.
The 3Dprint performed best across all measurements, and was also most favored by our participants: *"The 3D print was definitely the best"* (P11) or *"It feels very natural"* (P5). It allowed participants to explore the object, use landmarks to better understand the object and help them especially with the pointing: *"I can just follow the object"* (P13) or *"It allows me to do fine-grain adjustments when I touch it"* (P7). However, four participants told us that they found it challenging to work with the 3D print, because of its size. Furthermore, two stated that

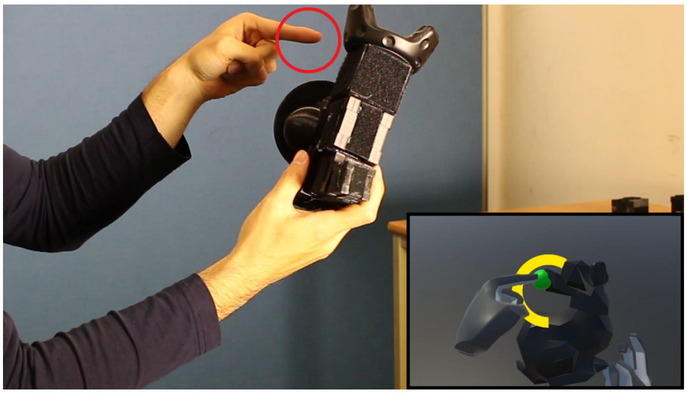


Figure 13: Points in the air (red circle), because the object is missing ears. Does not receive tactile feedback.

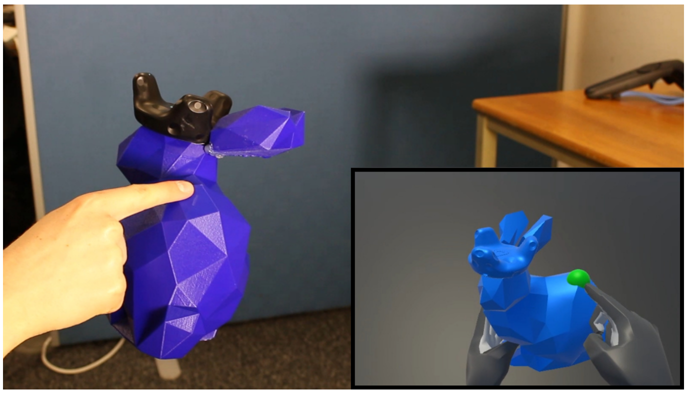


Figure 14: Points at the back of the bunny and receives tactile feedback while touching the physical 3D print due to high embodiment.

they found the weight distribution (center of gravity) confusing. This problem was created by using the HTC VIVE tracker on the head of the bunny, and has also been outlined in previous work [54, 56].

Study 1 Summary

This study demonstrates that TanGi allows people to build tangible proxy objects that can be used for object exploration in VR. TanGi's proxies helped participant's spatial understanding of virtual objects over the Controller condition, and generally increased their performances compared to free hand interactions. Up to this point, we only investigated how embodied exploration affects user interaction. Therefore, in our second study, we further investigate the use of the Manipulators for embodied object manipulation, which bring proxy objects closer to the rich manipulation possibilities of real-world objects. In the next chapter, we take a depper look at the Manipulators and their implementation.

6 TANGI'S MANIPULATORS

This chapter explains the concept and implementation of the *Manipulators*. First, we outline the underlying design of the Manipulators followed by reporting on lessons learned from our prototyping phase. Next, we show the components inside a Manipulator, and demonstrate how we fabricated and implemented them. Finally, we summarize the Manipulators.

All *.stl* models, circuit schematics and the processing code shown in this chapter are open-source and can be downloaded from our GitHub repository[1].

6.1 PROTOTYPING

This section provides further insights into our prototyping phase, and through which we report on our lessons learned. Since we expanded on TanGis' primitive cube design, we only needed a few design iterations modifying the cube's lid and base, so that they host all necessary hardware components. However, finding a pattern between cubes which allowed bending and stretching led to another experimental phase with different materials and fabrication methods. To ensure the reproducibility of our approach we report on our procedure, discuss their trade-offs and finally, present our prototype.

First, we clarify the design requirements for the stretching and bending patterns as follows:

Design requirements

1. The pattern can be stretched or bent without breaking it, and returns to its original state when no force is applied.
2. The force required to stretch/bend the pattern remains approximately the same, even after multiple repetitions.
3. Only off-the-shelf technology or hardware is needed to fabricate the pattern.
4. Designer should be able to replicate it with ease.
5. Following the toolkit design principles, modularity must be ensured.
6. The pattern enable different levels of stretchability and bendability depending on the use-case, leading to different haptic sensations.

1 https://github.com/MartinFk/TanGi

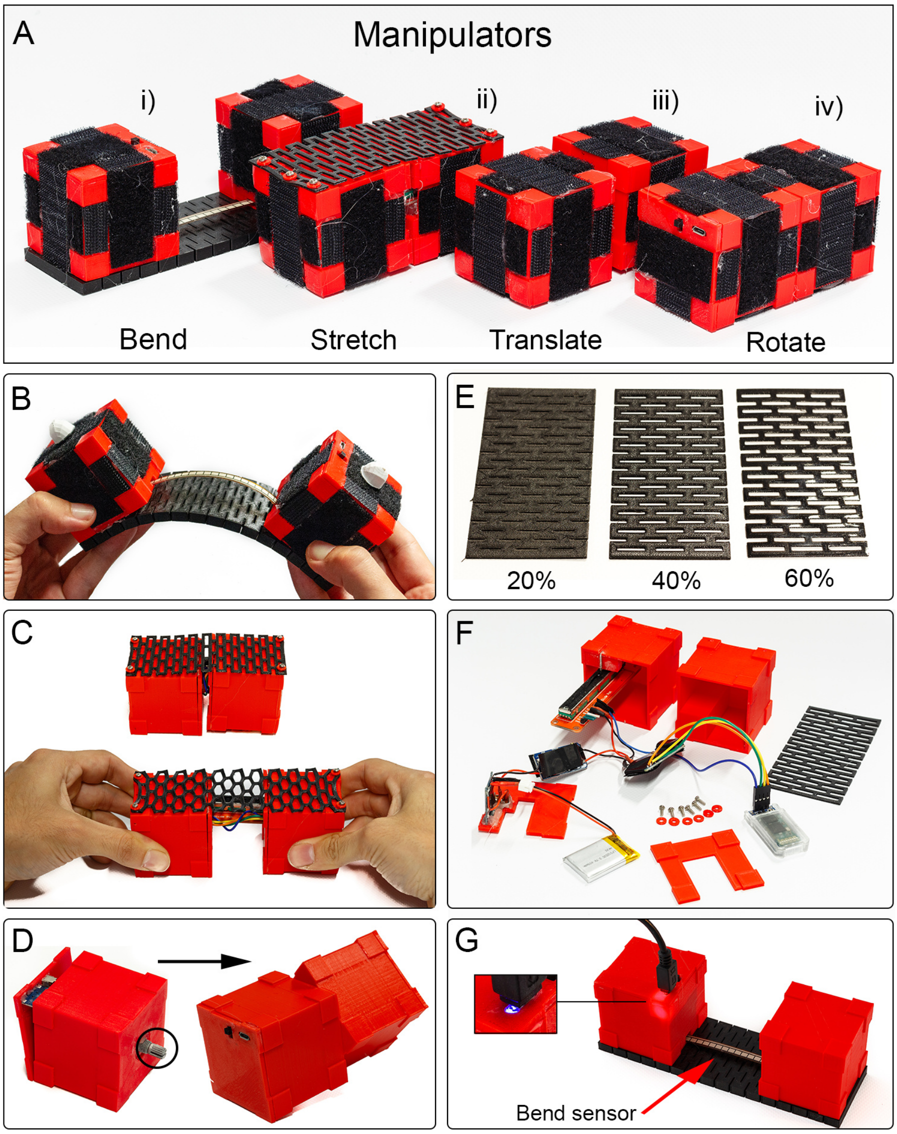


Figure 15: (A) The four different Manipulators: (i) unidirectional bending, (ii) variable linear stretching, (iii) linear translation and, (iv) single-axis rotation, all augmented with Velcro tape. (B) and (C) show stretching and bending. (D) shows a rotation Manipulator connected using a rotary potentiometer. (E) Variable stretching patterns. (F) All the components inside a stretching Manipulator including the modular stretching pattern. (G) Charging the battery; when charging, the LED is red - blue when done.

Ideally, the same method can be used for fabricating both bending and stretching pattern.

Wood and Laser cutting
In our first approach followed a similar process as Groeger and Steimle's [17] in their work on stretchable circuits. They laser cut a specific pattern in a non-flexible material. Due to the cutting pattern the material is hence stretchable. Hereby, the shape and spacing of the pattern

influence how stretchable the material becomes. However, they used very thin plastic sheets which were not suitable for our prototype. The cubes must stay in place when no force is applied. Driven by the laser cutting idea we decided to test different cutting patterns using plywood. The settings we used for cutting 4 mm plywood with our laser cutter (Universal Laser Systems 60 Watts) were *Mode*: vector cutting; *Power*: 75%; PPI: 250 and *Speed* 4.8%. For initial testing we downloaded different patterns from the Instructables website[2]. Early testing showed that laser-cut wood is not particularly suited for our prototype. Generally, it significantly increased the weight of the Manipulator. Moreover, it lacked robustness - after multiple bending and stretching cycles it started to loose tension. We were looking for a material that could be bent and stretched multiple times while retaining its original tension. Lastly, laser cutting also required additional hardware and expertise to fabricate the pattern. Therefore, we decided to move to a different approach describe below.

Laser cutting as a first approach

3D Printing using TPU

In the next iteration, we moved back to FDM 3D printing using a new material *Thermoplastic Polyurethane* (TPU). TPU is a flexible and very durable material that can withstand impact. Furthermore, it is abrasion, water and chemical resistant. Due to its flexibility it is challenging to print. We needed multiple iterations to determine the ideal settings for our FDM Ultimaker 3 printer. Here, we provide the settings and configuration we used to ensure reproducibility. Generally, flexible 3D printing materials are very sensitive to quick movements such as retraction of filament when moving to the next layer or when travelling to a different area on the printing bed. These are the well-known and most common issues when printing TPU. Reducing the retracting speed to 35 mm/s helped to avoid this. Also, making sure that the filament can easily run from the spool through the feeder without much resistance further reduced quick and unexpected movements. Furthermore, the printing speed was reduced to 25 mm/s while printing layer height was set to 0.1 mm using a 0.4 mm nozzle. The printing temperature was set to 220 degrees Celsius, and zero degrees for the build plate (no adhesion was used). We printed the bending and stretching patterns with 10% infill density with the cross 3D infill pattern in order to improve robustness.

Final prototype makes use of 3D printing with TPU

The final design patterns shown in Figure 15 (E) are the result of many iterations using different spacings and experimenting with material thickness. The final stretching patterns are 1 mm thick; 6 mm for bending. Depending on the printing pattern more or less force is required to bend and/or stretch them giving users the flexibility to create different haptic sensations.

2 https://www.instructables.com/workshop/laser-cutting/

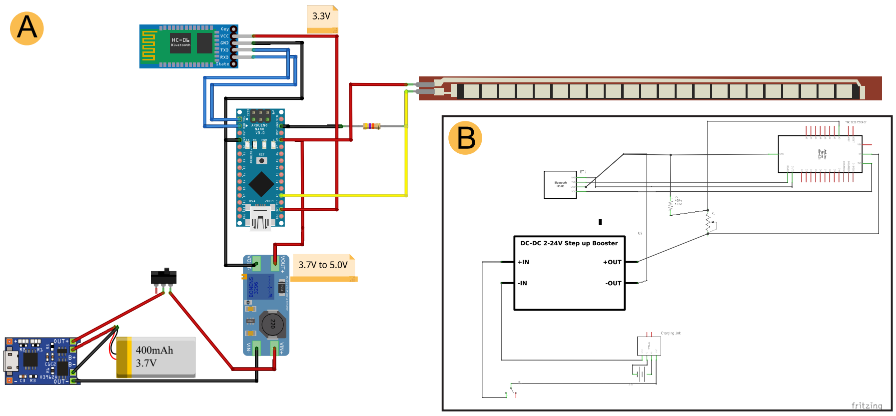


Figure 16: Manipulators circuit design: (A) shows an overview of all components and how they are connected inside a unidirectional bending Manipulator. (B) provides an in depth view of the circuit schematics to ensure reproducibility.

6.2 CIRCUIT DESIGN

This section outlines the inside-out tracking of the Manipulators. First, we unpack the Manipulators and describe each component as well as the final circuit. Next, we discuss the different sensors used to track object manipulations. Finally, we show the existing mechanism to ensure the user's safety.

Components & Circuit

Manipulator's hardware components

Each Manipulator uses low-cost off-the-shelf hardware components; an hc-06 Bluetooth module, an Arduino Nano 3.x (ATmega328), a voltage converter, a charging unit (chip TP4056), a 3.7 Volts 400 milliamp Hour(mAh) Lithium Polymer battery (LiPo), a 2-pin JST-PH connector, a switch, wires, connectors, resistors and different sensors such as an Adafruit bend sensor or potentiometers. Thus, Manipulators are self-contained and do not require external power or connection cables to transmit data. The circuit schematics Figure 16 shows how the different components are connected. The 3.7 Volts LiPo is connected to a charging unit enabling users to charge the battery using a micro USB cable. The maximum charging current is 1C. As shown in Figure 15 (G), Manipulators also indicate their charging status with a red led - currently charging, and a blue led - fully charged. Following the Arduino Nano specifications the microcontroller requires a minimum voltage of 5 Volts. Therefore, we utilized a voltage converter. We manually set the output voltage to 5 Volts using a DC adjustable power supply (model CSI3005SM) monitoring and ensuring a consistently low current. A switch is used to interrupt the current flow cutting-off the power supply. Switches are always connected to the positive terminal. *Note:* Since we have a direct circuit (DC) current only flows is one direction (negative to positive).

Figure 17: Different sensors: (A) a linear and (B) rotary potentiometer. (C) shows a bend (flex) sensor.

We power the Arduino Nano by connecting the power source to the 5 Volts input pin. This ensures that we do not bypass the Nano's internal overcurrent protection. The hc-06 Bluetooth module is powered with 3.3 Volts, and connected to the Arduino's RX and TX pins for wireless data transmission. The sensors are connected to the 5 Volts power source, because testing showed that it results in better signal quality. All components are connected to a common ground. The above illustrated circuit essentially is a voltage divider. A voltage divider takes a larger input voltage and by using resistors it outputs a smaller voltage. The sensors we used (describe below) act as variable resistors which, depending on the sensor state output a higher or lower voltage. Later in this section we describe our sensing firmware which detects this change in voltage.

Sensors

Different sensors for tracking Manipulator states

The Manipulators utilize different sensors in order to recognize changes in e.g. rotation and translation. We use resistive sensors, rotary and linear potentiometer, and a bend (sometimes also called flex) sensor which include conductors. As described in Chapter 3, the rotation Manipulator uses a rotary potentiometer, translation and stretching Manipulator use a linear potentiometer, and the bending Manipulator uses a bend sensor. Their underlying working principle is that the conductor length is directly proportional to its resistance and vice versa. For instance, the bend sensor has a one-sided print with a polymer ink including conductive particles. By bending the sensor the conductive particles move further apart, and thus result in an increased resistance. Moving the bend sensor back to its default (straight) state decreases the resistance, because the particles are closer together. We detect this difference in resistance using the sensing firmware describe below, and process it accordingly.

Generally, we found that connecting the sensor to 5 Volts significantly improves the signal quality. Furthermore, for stretching we also tested conductive rubber band that we planned to integrate in the stretching pattern. However, we found-out that it took several seconds to see changes in resistance when stretching and releasing it. Therefore, we replaced this sensor with a linear potentiometer which offered a much higher resolution.

Safety

Since we developed and implemented a new prototype system which makes use of a custom-designed circuit powered by a LiPo battery, we were also concerned with user's safety. First of all, the circuit has three build-in safety mechanisms: (1) we ordered high quality LiPo batteries which include a overcurrent protection cutting of the connection (e.g. during a short circuit). It also has a overcharge/discharge protection, (2) the charger implements the same safety mechanisms as the LiPo battery resulting in a double-layer of protection, and (3) the Arduino Nano's current limiting component, which mainly regulates voltage[3]. In the latter, it is important to connect the battery to the input pins. Otherwise, we would bypass the Nano's on-board current control. Moreover, in the development process of the circuit we used a breadboard and jumper wires to connect all components. This allowed us to check different parameters such as current and voltage for all hardware components by using a multimeter. Furthermore, we included a 47 kiloohms resistor to prevent short circuits. The maximum current flow in the circuit is 45 milliampere, and the overall resistance of the circuit that provides functionality is 4.7 Ohm.

Manipulators implement different safety mechanisms

This demonstrates that the Manipulators are safe to use. Since the circuit and the human would be connected in parallel they effectively form a current divider (two resistors connected in parallel). Due to the low resistance users could even directly touch wires and hardware components (in the unlikely scenario that a Manipulator falls apart), because human's skin resistance is significantly higher following [44].

6.3 fabrication

Here, we describe the fabrication process of the Manipulators. We modified our cube primitive design in Rhino3D to accommodate all components and parts inside. Furthermore, as visible in Figure 15 (D - right) Manipulators offer a micro USB port for charging as well as an on/off switch. Additional components such as washers and stabilizers for the rotation Manipulator were custom designed and 3D printed on a FDM printer using standard PLA. As a result, besides the off-the-shelf hardware components only 3D printing is necessary to fabricate Manipulators. In order to pack all components into the cube case we had to optimize (1) the placement of the components and (2) reduce the space needed for wires. Since jumper wires occupied too much space, we decided to solder all components and cables. To prevent short-circuits, we used shrinking tube and insulated the

Adjusting the Manipulator design

3 http://www.ti.com/lit/ds/symlink/lm1117.pdf

electrical components with PVC (polyvinyl chloride) electrical tape as visible in Figure 15 (F).

An alternative solution was to design a custom Printed Circuit Board (PCB) with embedded hardware components. However, this turned-out to be fairly expensive (> 200 British Pounds Sterling) for a single PCB board excluding the battery and sensor, whereas buying the individual components and soldering them was significantly cheaper (between 20 and 35 British Pounds Sterling per Manipulator depending on the sensor). Another advantage is the maintenance; in case one of the components fails, it can simply be replaced.

Custom PCB as an alternative

To fabricate the patterns that allow stretching and bending, we used TPU as described above. These patterns are modular and can be exchanged depending on the user's needs. To demonstrate this, we printed three different patterns for stretching and bending. For instance, Figure 15 (E) shows the three different stretching patterns which are 20%, 40% and 60% stretchable. The stretching pattern is fixed atop of the stretching Manipulator using three screws and 3D printed washers on both sides.

Finally, we assembled the cubes, and fastened all components using super glue as well as hot glue, preventing them from moving around.

6.4 implementation

In this section we provide a detailed look into the implementation of the Manipulators. First, we give an overview of the system and its architecture. Next, we take a closer look at the Ardurino and the Unity3D program, and we explain how the communication works. For our second experiment we implemented a Free-hand and a controller version of the Manipulators which will be discussed at the end of this section.

Overview

We used Unity3D to design and implement the virtual environment including all components, SDK's, libraries, and design principles as mentioned in the previous Chapter 4. The Manipulators implementation consists of two programs: one running on the Ardurino and another one on Unity3D. The Ardurino program continuously senses sensor states and sends updates to the Unity3D program using wireless Bluetooth communication. On the Unity side, the system receives the data, processes it, and visualizes it in the user's virtual environment. Thus, sensing and data processing happen in two distinct programs offering great flexibility. As a result, all interfaces implement the same data structure and only differ in how they interpret, map and visualize the incoming data.

How do the Manipulators work in Unity?

```
void loop()
{
  int ADC = analogRead(A0);//(1)
  float Voltage = (ADC * 5.00) / 1023.0;//(2)
  float RESISTANCE = RESISTOR * (Voltage / 1023.0);//(3)
  Serial.println(RESISTANCE);//(4)
}
```

Listing 1: Resistive sensing code

Another advantage is the maintenance of the Manipulators. The sensing firmware running on the Ardurino inside the cubes can be updated via Bluetooth. Next, we look at the sensing firmware which is running on the Ardurino microcontroller.

Sensing firmware

The Ardurino firmware was developed using C++ and Ardurino IDE[4] (version 1.8.9). The firmware continuously executes a resistive sensing algorithm. Listing 1 shows the part of the code which computes the resistance, and writes it into the serial port (4). Next, we briefly run through the different steps required to create a signal that can be interpreted and processed. In (1) the program uses the analog-to-digital converter (ADC) of the microcontroller to convert the analog voltage readings from pin A0 into a digital signal. Here, the analogRead() function[5] reads the voltage and maps it to a number between zero (minimum voltage) and 1023 (maximum voltage). In order to determine the voltage and the resistance, we utilize the voltage divider that we described above.

Reading and processing sensor states

To do so, the program takes the ADC reading, multiplies this with the voltage in the system (5 Volts) and divides it by the maximum ADC value (2). Hence, we calculated the current voltage reading from pin A0. Following this, the system is able to compute the resistance (3). Furthermore, we set the baud rate to 9600 meaning that we send 9600 bits per second (maximum) over serial port.

Unity Processing

We developed a Plug& Play Unity3D asset/module allowing novices to quickly integrate the Manipulators into their VR environment, thus enabling them to build and use richer proxy objects. Importing our Manipulator module, a library including all components for serial communication into Unity3d provides the VR interface for the Manipulators. To do so, we used a slightly modified model-view-controller (MVC) design pattern in order to process and visualize incoming data with respect to their physical counterpart in VR. Following, we start with the View, next the Controller and finally the Model classes.

Software design pattern

4 https://www.arduino.cc/en/main/software
5 http://arduino.cc/en/Tutorial/ReadAnalogVoltage

Views include all necessary 3D CAD files (e.g., *.obj* or *.stl*) for representing the virtual object in the environment (e.g., the Stanford bunny as seen in study 1).

Controller enables us to establish the communication. First, we have to open the serial port. Note that matching the baudrate 9600 is essential for the communication, otherwise the system would suffer data loss.

Executing the processing in the *Update()* method using the main thread would block the program. Therefore, we utilize Coroutines and IEnumerators to continuously read data from the serial port. Through Coroutines and IEnumerators we can asynchronously read incoming data without interfering or blocking the main Unity thread. Updates are stored in a ConcurrentQueue before processing them. Each frame (30fps) the system selects the most recent entry in the queue. Subsequently, it runs through a *MovingAverage()* filter function. We use this common digital signal processing method to reduce the random noise in our system [49]. Finally, the system sends the received update to the corresponding model (described below) in order to visualize it in VR.

Models implement the logic and rules for rotation, translation, stretching, and bending Manipulators. Common methods and attributes are outsourced into a base class. The deriving classes capsule Manipulators in parametric C# classes. Thus, parts of any arbitrary 3D model in Unity can, for instance be rotated by tracking and dropping the desired Manipulator C# script to it. Parametric manipulator classes offer various settings to e.g. calibrate sensors, modify resolution or for limiting the DoF. To better understand this process, we use a concrete example case below.

Manipulator configuration in Unity

For instance, Sally wants to bend a virtual object using the bending Manipulator. She attaches the desired script to the bendable part of the virtual object. Required parameters are maximum bend angle (e.g. 90 degrees), selecting the bend axis (x,y, or z), maximum and minimum resistance of the sensor, and more advanced settings such as adjust the *movingAverage* buffer size for signal quality/noise optimization. Sally can visualize bend angles and bend axis beforehand using sliders and check boxes in the interface helping her to better understand and predict complex object manipulations. The manipulator class implements a *IntervalMapper()* function which maps the interval size (e.g. 90 degrees for bending) to the resistance values of this sensor; determined through minimum and maximum resistance input. Sally can either enter these values manually or once the connection is established, they can use the calibration function. The calibration function consists of two simple buttons (min and max). This works by pressing the min button, when the bend sensor is straight, and the max button when the bend sensor is fully bent (in our example 90 degrees). Subsequently, the system computes the intervals resulting in a direct mapping between the resistance from the sensor and the corresponding visualisation in VR. Finally, a *CheckInterval()* method maps the incoming resistance updates from the controller to the corresponding interval, which is visualized in VR. Depending on the Manipulator this either means linear translation/ stretching using *transform.localPosition* or single-axis rotation through *transform.localRotation*. In the latter, we use quaternions to avoid gimbal locks. The *transform.local* function is required, since the system still performs 6DoF tracking using Vive trackers. In fact, we provide 7DoF tracking for TanGi proxies that, for instance include one single-axis rotation Manipulator.

For bending simple 3D virtual objects we used a mesh deformer Unity3D asset[6] from the Unity asset store, because it is challenging to realize bending in Unity3D, and it was not in the scope of this thesis.

Finally, users can utilize multiple Manipulator simultaneously. Here, each Manipulator uses a different serial port for streaming data to the VR machine. In the current version we provide up to 10DoF tracking (four Manipulators).

Study modification

As in Chapter 5 we modified the system to run our second experiment described in the next Chapter 7.

Required modifications for our second study

First of all, we used the same testbed developed in Section 4.3 to execute the study, and to track various data such as task completion times. Again, the data was exported as .csv files for further analysis. Our pilot testing showed that allowing 6DoF manipulations on virtual object using a Vive controller or a Free-hand interaction technique was challenging for participants. Therefore, we decided to implement a Controller and Free-hand version of the Manipulators only allowing single-axis rotation, linear translation and stretching as well as unidirectional bending. For the Controller condition we simply did this by restricting the DoF using Unity's *RigidbodyConstraints.FreezePosition* attribute. However, in case of the Free-hand interaction technique it led to arbitrary object movements. Following this, we implemented an interaction layer which overrides the Leap Motion's interaction engine. Once users grasped the object, rotations and translations of their wrist got immediately displayed on the object, due to a direct mapping between hand and virtual object orientation.

6 https://assetstore.unity.com/packages/tools/animation/bend-deformer-38494

6.5 SUMMARY

In this chapter, we first described the prototyping and development phase of TanGi's Manipulators. We discussed alternative solutions such as different materials and fabrication methods, and presented our final prototype. Moreover, we provided a detailed look into the technology and implementation inside the Manipulators. We showed the circuit schematics and explained how the wireless communication works. Finally, we described the implementation of a Free-hand and Controller version of the Manipulators, which we used in our second experiment outlined in the next chapter.

7 STUDY 2 - EMBODIED OBJECT MANIPULATION

This chapter covers the second lab-study we conducted to understand how embodied manipulation using TanGi affects user interaction. We outline the design of the study, provide background information of participants as well as we explain our data analysis/collection procedure. Finally, we also elucidate the findings from the study.

7.1 DESIGN STUDY 2

Why did we design the study?

While our first study focused on how different control types affected exploration of a virtual object, our second study focused on how embodiment affects manipulations of virtual objects. Specifically, we wanted to understand the impact of TanGi proxies on manipulation tasks. To build this understanding, we conducted a controlled laboratory experiment where participants completed single dimension manipulation tasks. Participants completed trials where each of the three control types (Free-hand, Controller and TanGi) represented a different level of embodied interaction.

We were interested in comparing three conditions (Free-hand, Controller and Toolkit) to perform three different primitive object interactions; rotating, stretching and bending parts of an object. Since, linear translation and linear stretching is essentially the same for the Free-hand and Controller condition, we decided to only include one (linear stretching) in the study. Moreover, 3D printing does not easily allow people to fabricate objects including rotating, stretchable, and bendable parts. Therefore, it was not part of this experiment. We counterbalanced the three different conditions resulting in six permutations.

Participants

Participant background and information

We recruited a new set of 12 participants (6 reported as female; 6 reported as male), aged 19-35 (avg: 25.46; sd: 4.8) with a range of

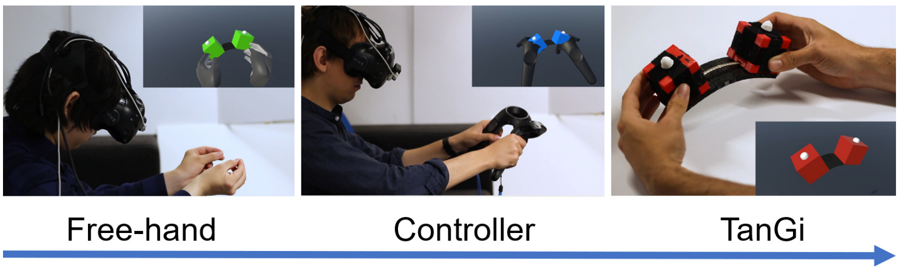


Figure 18: Study 2 conditions.

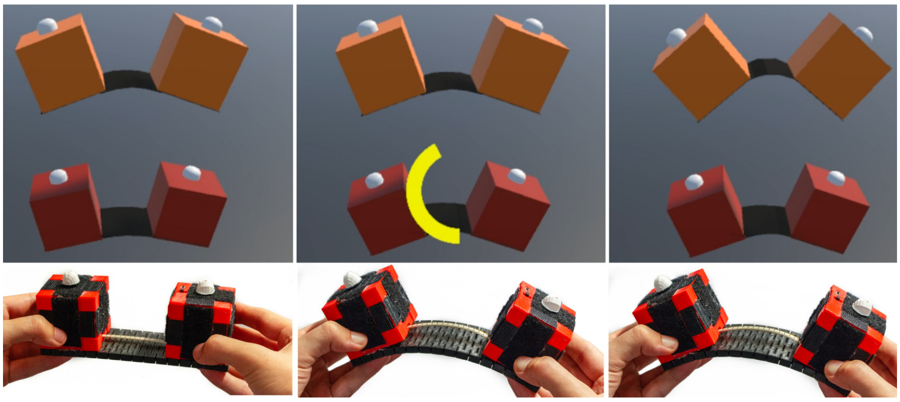


Figure 19: Study 2 task: Bending example for study 2. Goal bend degress is shown (orange); participant matches it (progress bar starts); next bending state is displayed (left to right).

professional and educational backgrounds including humanities education, geography, computer science, psychology, environmental science, linguistic, English literature, and civil service. This excludes one participant that was omitted before analysis, due to a problem with experimental system. Each participant was provided a £5 Amazon Voucher as remuneration. Five participants reported that they had never used VR before, five had used it a few times (one to five times a year), and two other subjects use it on a regular basis (more than 10 times a year). Participants from the first study were not permitted to take part in this experiment.

Task Design

The three tasks were modeled to help us compare three different levels of embodied manipulation: Free-hand, Controller and TanGi . For instance, participants were required to re-produce five different levels of stretch, match five different rotation and bend states within a threshold. Early pilot testing revealed that allowing 6DoF for the Controller and the Free-hand interaction technique was challenging. To ensure the equality of the different conditions we restricted the DoF for Free-hand and Controller. Thus, we essentially implemented a virtual version of the Manipulators by only allowing single-axis rotation, linear stretching and unidirectional bending. Following study 1, subjects were required to hold the object for two seconds (indicated through a yellow progress bar). A second object above showed how much rotation, stretch and bend was required.

Goal of the study task

Procedure

After giving participants a general introduction to the study, we explained the task, and showed them the first condition. Next, they performed practice rounds for rotating, bending and stretching, before they did the main experiment. This gave them the opportunity

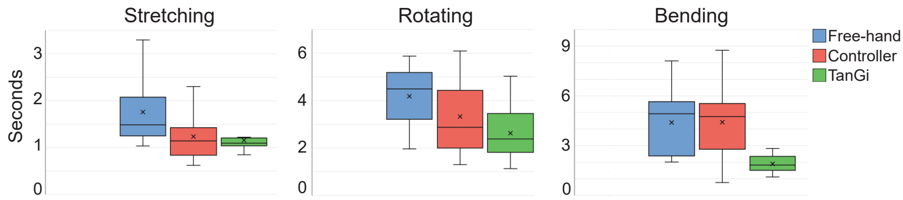


Figure 20: Task completion times study 2.

to familiarize themselves with VR, the study task and the condition. We fully counterbalanced the presentation of the three different conditions resulting in six permutations. The experiment took about 30 minutes. The study has been approved by University College London's Ethics Committee.

Analysis
We collected data from six sources: a pre-study questionnaire for demographic information; video of the participant as they completed the experiment; internal logging (e.g. task completion times, accuracy), field notes and observations; a post-study questionnaire and a short semi-structured interview to better understand participants' experiences in the different conditions.

Pilot study
In a first informative pilot study we found that offering 6DoF manipulations in the Controller and Free-hand condition was challenging for participants. Therefore, we decided to restrict the DoF as described in Section 6.4. We piloted these two interaction techniques in our lab before we included them in our actual experiment.

7.2 RESULTS & FINDINGS STUDY 2

Here, we focus on the findings from our second experiment. Particularly, on how people make use of the Manipulators, and we contrast their experiences with the Controller and Free-hand condition to explore embodied manipulations. In the analysis, one participant was omitted before data analysis, due to a problem with the Leap Motion sensor. Therefore, we recruited an additional participant.

Generally, the Manipulators outperformed the two other conditions across all measurements. First, we take a look at the task completion times (mean. for one trial) for the three tasks rotation, bending and stretching (also see Figure 20).

Completion Time.
To further investigate our data, we again ran One-Way repeated measures ANOVAs after verifying the normality distribution assumption

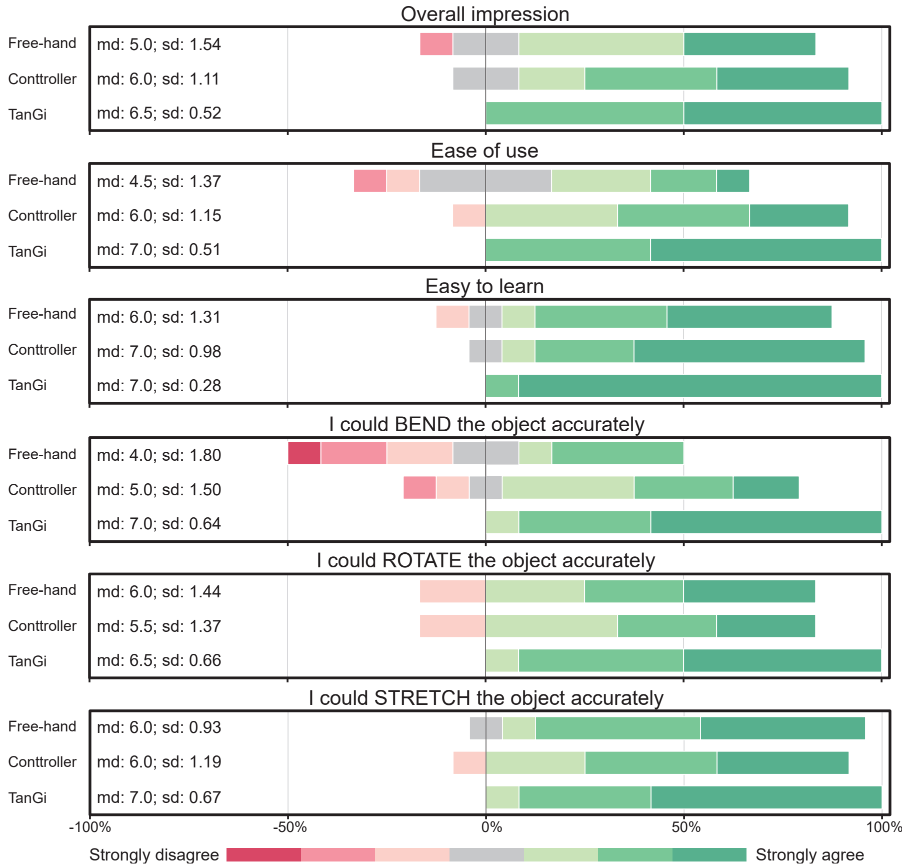


Figure 21: : Post-study questionnaire results on a 7-point Likert-type scale (1= Strongly disagree; 7=Strongly agree).

through Lilliefors tests. Main effects were analyzed for significance difference using post-hoc Bonferroni-Dunn tests.

The times for rotation were Free-hand (mean: 4.18 sec., sd: 1.34 sec.), Controller (mean: 3.32 sec., sd: 1.63 sec.), and Manipulators (mean: 2.62 sec., sd: 1.17 sec.). We found a main effect ($F_{2,\,22} = 6.408$, $p < .05$). Post-hoc tests revealed a significant difference between the Free-hand and the Manipulators condition.

Task completion times in the different conditions

Next, Stretching Free-hand (mean: 1.75 sec., sd: 0.72 sec.), Controller (mean: 1.23 sec., sd: 0.49 sec.), and Manipulators (mean: 1.15 sec., sd: 0.23 sec.). The ANOVA showed a main effect ($F_{2,\,22} = 4.429$, $p < .05$). We found a significant difference between the Free-hand and the Manipulators condition.

Finally, bending times were Free-hand (mean: 4.39 sec., sd: 2.06 sec.), Controller (mean: 4.41 sec., sd: 2.18 sec.), and Manipulators (mean: 1.90 sec., sd: 0.56 sec.). A main effect was found ($F_{2,\,22} = 11.969$, $p < .05$), and a significant difference between the Free-hand and the Manipulators condition as well as the Controller and the Manipulators condition were revealed.

Generally, bending was challenging for participants. Even though we constrained the DoF it still required to manipulate two virtual objects relative to one another. As our early pilot testing showed this confronts participants with challenges.

Subjective Responses.
This trend also aligns with participants' questionnaire responses on a 7-point Likert-type scale (1-Strongly disagree; 7-Strongly agree). For instance, medians for *"Overall impression of the system: I would use the system for virtual 3D object manipulation"* were: Manipulators (median: 6.5, sd: 0.52), Controller (median: 6.0, sd: 1.11) and Free-Hand (median: 5.0, sd: 1.54). Participants stated that Manipulators were *"easy to use"* Manipulators (median: 7.0, sd: 0.51), Controller (median: 6.0, sd: 1.15), Free-Hand (median: 4.5, sd: 1.37), and *"easy to learn"* Manipulators (median: 7.0, sd: 0.28), Controller (median: 7.0, sd: 0.98), Free-Hand (median: 6.0, sd: 1.31)). Findings regarding task completion times indicated that participants struggled with the bending task in the Controller and Free-hand condition. This was also visible in the questionnaire responses to *"I could BEND the object accurately"* Manipulators (median: 7.0, sd: 0.64), Controller (median: 5.0, sd: 1.50) and Free-Hand (median: 4.0, sd: 1.80). Whereas the tasks (stretching and rotating) that only required the direct manipulation of one virtual object seemed easier *"I could ROTATE the object accurately"*; Manipulators (median: 6.5, sd: 0.66), Controller (median: 5.5, sd: 1.37) and Free-Hand (median: 6.0, sd: 1.44) and *"I could STRETCH the object accurately"*; Manipulators (median: 7.0, sd: 0.67), Controller (median: 6.0, sd: 1.19) and Free-Hand (median: 6.0, sd: 0.93). Next, we provide further insights into how people used and experienced the different conditions.

Manipulators were favoured in interviews

Free-hand.
Participants had very mixed opinions about the Free-hand interaction regardless their prior experience with VR. Interview data showed that generally participants would use this technique *"..., if I want to have a less conscious or free interaction"* (P11) with the virtual model. Performing these very specific manipulations required a lot of active thinking and people were *"... very focused on my hand movements"* (P4), because subtle changes in hand orientation got immediately displayed on the object. Interestingly, participants frequently reported that they felt very tensed while interacting with the virtual object. *"It was very tiring for my arm grasping literally nothing"* (P9), and mentioned that *"it seems like the energy just goes somewhere when I have an object in my hand"* (P5).

Controller.
The Controller with its uniform shape was slightly preferred over the Free-hand condition in our study providing a tangible way to interact with a virtual model. *"Having an object to hold onto made it easier to keep*

position of the cubes relative to each other." (P6). However, it added an additional layer between the human and the object in order to interact with it pointed out by participant 11: *"Controller feels like a barrier to the object"*.

Manipulators.
Overall, the Manipulators performed best offering a *"direct way to interact with the virtual object"* (P11). Due to the direct mapping between the object interactions (stretch, bend, rotate) people *"... can apply the movement I [they] learned"* (P12). Moreover *"The cubes [Manipulators] appeared much easier to stretch, due to the physical feedback (i.e. actually holding two objects in your hand), whereas the other two methods were a little bit more difficult, as they appeared more 'abstract' "* (P7). The Manipulators allowed users to easily perform *"... subtle adjustments"* (P1) being very precise and furthermore supporting them to *"better understand the object its capabilities and limitations"* (P1).

Study 2 Summary
The study demonstrates that TanGi Manipulators enable people to perform complex object manipulations with ease due to a higher degree of embodiment. Furthermore, it provides interesting insights showing the trade-offs between the different levels of embodiment. In the next chapter, we present different applications and uses cases to further evaluate the TanGi toolkit.

8 USE CASES & APPLICATIONS

The TanGi toolkit is composed of Composable Shape Primitives and Manipulators which enable users to quickly build a variety of objects allowing complex manipulations. In this chapter, we illustrate how the toolkit can be used to build several different tangible proxies that represent and control virtual objects. Importing our Manipulator module (a library including all components for serial communication) into Unity3d provides the VR interface for the Manipulators. TanGi proxies are tracked (6DoF) by using conventional VIVE trackers.

Bunny Head

This application shows a modified Stanford bunny (toy) which can turn its head (see Figure 22). We build the bunny using a single-axis rotation Manipulator, five 50mm cubes, two 40mm sticks, three 40mm half-spheres and two 100mm half-spheres. Its virtual representation gets rendered accordantly to the physical proxy object. To do so, a user would simply replace two of the TanGi's base cubes with the rotation Manipulator. In Unity3d s/he imports 3D models for the head and the body of the bunny. By attaching the *RotationManipulator* script to the 3D model, the bunny can now receive updates. After selecting the local rotation axis through the script, the head rotation gets displayed.

Users can rotate the bunny toy's head

Catapult

In our second example Figure 23, a gameplay catapult utilizes a unidirectional bending Manipulator, six 50mm cubes and two 100mm sticks. Users can move the catapult in the desired position and load it by bending the Manipulator. To launch a virtual stone the user releases the cube on the end of the bend Manipulator which then

By bending the catapult users can launch a stone

Figure 22: A manipulable toy bunny on a virtual beach.

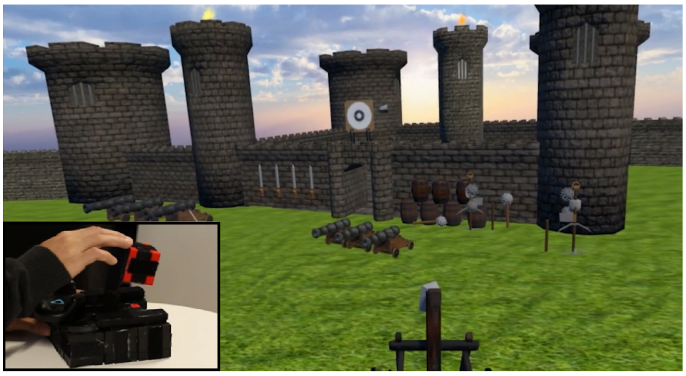


Figure 23: A medievel siege game, where the proxy corresponds directly to the in-game catapult.

accelerates. In the virtual scene the throwing mechanism also bends accordantly to the bend state of the Manipulator. The maximum state determines how much force is applied to the stone, and consequently affects its velocity and trajectory. This application implements the *BendingManipulator* script.

Crossbow

Cocking a crossbow in VR

This third application shown in Figure 24 makes use of the linear-stretching Manipulator with a 40% stretching pattern (see Figure 15), two 50mm cubes as well as two 30mm cubes. The user aims on the target and pulls back on the virtual arrow using the physical block; the virtual crossbow gets rendered with respect to the stretched Manipulator. Once, the user lets go of the cube, it snaps back and triggers the arrow. Based on how much the user cocked the crossbow, the arrow's trajectory and speed is determined. As in the previous application, simply attaching the *LinearTranslationManipulator* script to the crossbow model.

Robotic arm

This last example application demonstrates the use of TanGi for con-

Figure 24: A crossbow proxy used in a shooting-gallery style game.

Figure 25: Creating a proxy to explore and manipulate an existing industrial robotic arm model.

Using two Manipulators to move and control a robotic arm

trolling a robotic arm (see Figure 25). The model for this arm is similar to an existing robotic arm model used in industrial settings [15]. It uses two Manipulators, single-axis rotation and linear translation, simultaneously. Furthermore, we used five 50mm cubes, two 100mm sticks, two 50mm sticks as well as two 25mm sticks to assemble to robotic-arm. The user can move the object 6DoF in space, however she can also rotate the robot wrist independently to adjust and fine-tune the gripper orientation using the single-axis rotation Manipulator. To open and close the gripper the physical proxy robotic arm utilizes a linear translation Manipulator. The *RotationManipulator* script determines the orientation of the wrist/gripper. Based on this, the *LinearTranslationManipulator* script renders the virtual gripper with respect to its physical counterpart delivering a hands-on experience while controlling the virtual robotic arm.

These example applications act as a proof-by-example (as suggested by [32]), and illustrate a wide spectrum of possible use cases for TanGi, from toys and gameplay to industrial applications. Futhermore, through our two previous user studies we demonstrated that composable proxy objects and Manipulators provide advantages in terms of usability and naturalness when used for object exploration and manipulation.

9

DISCUSSION & FUTURE WORK

Based on our studies, we discuss TanGi proxies and their utility for embodied interaction in VR, identifying opportunities to improve the toolkit.

Embodied Exploration and Manipulation with TanGi

The TanGi toolkit gives people the capability to create tangible proxies linked to corresponding VR models. Study 1 showed that people can easily create tangible proxies using the TanGi toolkit. These were good enough for basic exploration tasks such that people's performance with them was on par with a 3D printed virtual object. As we showed in our design explorations and studies, the current prototype of the TanGi toolkit enables a wide range of proxy possibilities. The tangible proxies enable embodied exploration and embodied manipulation. For the participants in our studies, the proxies were used as if they were the virtual object. Exploring different sides of a virtual model and pointing at different parts of it was accomplished by turning the proxy, and pointing at it. Similarly, manipulating different aspects of the virtual model was done by manipulating the proxy. Many participants described developing an understanding of the capabilities and limitations of the virtual model through their handling and manipulation of the TanGi proxy. Instead, participants described the Free-hand and Controller conditions as introducing a "layer" between their interactions and the virtual model.

TanGi enables rapid proxy creation

TanGi proxies help understanding object capabilities

This embodied interaction presents problems when there are mismatches between proxy and virtual model. The tangible proxies are ultimately approximations of the virtual model; as described in Study 1, each participant approximated the bunny in different ways—some built details like ears while others focused on simply approximating size. The problems with the mismatches would manifest in some fairly obvious ways; for instance, participants would overshoot when trying to point/rest their hand on the virtual model's ear if the TanGi proxy did not have ears. Additionally, participants indicated that secondary characteristics of the proxy were also important; for example, the overall weight and the centre of gravity of the proxy. In Study 1, the TanGi proxy needed to be affixed with a relatively heavy tracker, which threw off how participants expected to be able to handle the proxy (based on how it looked in the VR world). The fact these limitations arose indicate that the TanGi proxies did very much embody the virtual models for participants.

Low fidelity trade-offs

Improving the Design of the TanGi Toolkit

TanGi's limitations

While TanGi worked as designed, our experiences provide some clear directions for improvement. TanGi allows people to rapidly build proxies that embody virtual objects by approximating size, shape and manipulations close to what is expected. However, currently TanGi composable blocks are limited in what types of proxies can be created. We believe this can be easily improved upon, for example with additional primitive shapes that few participants asked for. We could easily create a larger range of shapes (e.g. cylinders, pyramids, etc.) in various sizes. This increases the complexity of actually building proxies, but provides more flexibility in the range of models that can be represented.

Upgrading the hardware

Recently, Ardurino announced a new Nano IoT microcontroller offering on-board Wifi, Bluetooth and an accelerometer. In future work we want to replace the exciting hardware with this new device. As a result, Manipulators could have a much smaller form factor offering greater flexibility and new possibilities. We also plan to employ actuators inside Manipulators allowing proxy objects to be even closer to their virtual counterpart. For instance, a fully functional power drill that can be started by pressing a button.

Furthermore, while we used Velcro to affix blocks to one another, other well-engineered approaches could be leveraged. For example, 3D printed snaps or anchors can be incorporated directly into our 3D prints, providing robust and strong connections that are less likely to break. And while the standard Vive trackers added bulk and weight to the proxies built in Study 1, we could replace them with smaller and lighter emerging trackers (e.g., HiveTracker [24]). This would allow us to have trackers on each individual shape primitive. Hence, we could visualize primitives in VR enabling users to assemble TanGi objects. Finally, it might be possible to provide tactile feedback for parts of the proxies that do not have physical manifestation. For example, recent work has shown that worn devices such as temporary tattoos can be used to provide electro tactile feedback [23, 53]. Furthermore, it may be possible to use certain types of haptic retargeting to provide this tactile sensation [6].

Generalized Controllers with TanGi

Using TanGi's to build custom in- and output controller

Beyond interacting with VR objects, participants suggested that the TanGi concept could be used for building more generalized, custom input and output controllers. For example, the robotic arm in Figure 4d can be modelled with various manipulators (for steering, rotating and twisting different parts of the arm). In principle, a simple interface to the robot operating system ROS [43] would allow users to control an actual robot arm using TanGi proxies. Other application domains might include AR (e.g. [21]).

10 CONCLUSION

This thesis outlines the development of TanGi, an open-source toolkit that allows novice users to rapidly build tangible proxy objects in VR. TanGi enables virtual objects to be embodied by approximating their shape and moveable parts, enabling fast and easy virtual object exploration and manipulation. We demonstrated TanGi's flexibility and expressive power by presenting a variety of potential uses cases and applications. Moreover, we provided a detailed look into the design and implementation of the TanGi toolkit. Finally, through two lab studies we show that different levels of proxy embodiment affect fluidity of virtual object interaction, and that TanGi proxies offer clear advantages over conventional controller. Our work extends the state-of-the-art in virtual reality technology, by demonstrating a new way to build, richer more fully embodied proxy objects.

APPENDIX A

This chapter provides supplementary materials for this thesis. In Section A.1 we show an additional figure from our first study. Finally, Section A.2 provides the pre- and post-study questionnaires we used in both user studies.

A.1 ILLUSTRATIONS

Figure 26 shows the 16 bunnies participants built in the first part of study 1. All participants completed this task successfully, and thus used the assemblies in part 2 of the experiment.

Figure 26: Study 1 assembled bunnies.

A.2 STUDY MATERIALS

The following pages provide the questionnaire from the first and the second study, as well as the interview questions.

Tangible Proxies (Pre-study)

*Required

1. **Participant# (for experimenter use)** *

2. **Participant condition (for experimenter use)** *
   Mark only one oval.
   - ( ) 1
   - ( ) 2
   - ( ) 3
   - ( ) 4

3. **Gender** *
   Mark only one oval.
   - ( ) Male
   - ( ) Female
   - ( ) Other
   - ( ) Prefer not to say

4. **Age**

5. **Professional Background (e.g. trainings, Bachelor, Master, PhD studies)**

6. **Prior experience with VR? (devices such as HTC VIVE, Oculus Rift, Playstation VR)**
   Mark only one oval.
   - ( ) no experience – I have never used it or don't recall
   - ( ) sometimes or infrequently - I use it 1 to 5 times a year
   - ( ) often – I use it 6 - 10 times a year
   - ( ) Option 4

7. **Please briefly describe which system you have used, and for what?**

8. **Prior experience with virtual 3D object interaction (e.g., design programs, CAD, Unity3D etc.)**
Mark only one oval.

- no experience – I have never used it or don’t recall
- sometimes or infrequently - I use it 1 to 5 times a year
- often – I use it 6 - 10 times a year
- regular basis – I use it more than 10 times a year

9. **Please briefly describe which system you have used, and for what?**

10. **Prior experience with physical 3D modelling (arts, model making, Lego etc.)**
Mark only one oval.

- no experience – I have never used it or don’t recall
- sometimes or infrequently - I use it 1 to 5 times a year
- often – I use it 6 - 10 times a year
- regular basis – I use it more than 10 times a year

11. **Please briefly describe which system you have used, and for what?**

Tangible Proxies (Post-study)

*Required

1. **Participant# (for experimenter use)** *

2. **Participant condition (for experimenter use)** *

   Mark only one oval.

   - 1
   - 2
   - 3
   - 4

3. **Overall impression of the system: I would use this system for virtual 3D object interaction** *

   Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

4. **Overall ease of use: The system was easy to use** *

   Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

5. **Easy to learn: I found the system was easy to learn** *

   Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

6. **I completed the task quickly** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

7. **I could orient the object accurately** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

8. **I could point at the object accurately** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
My object							
Vive controller							
3D print							

9. **I am satisfied with my object after the study** *

Mark only one oval.

- Strongly agree
- Agree
- Neutral
- Disagree
- Strongly Disagree

10. **If not, please describe why (what would you change)**

Tangible Proxies Study 2

*Required

1. **Participant# (for experimenter use)** *

2. **Participant condition (for experimenter use)** *
   Mark only one oval.

   - 1
   - 2
   - 3
   - 4
   - 5
   - 6

3. **Gender** *
   Mark only one oval.

   - Male
   - Female
   - Other
   - Prefer not to say

4. **Age**

5. **Professional Background (e.g. Chemistry, Physics, etc.)** *

6. **Prior experience with VR? (devices such as HTC VIVE, Oculus Rift, Playstation VR)**
   Mark only one oval.

   - no experience – I have never used it or don't recall
   - sometimes or infrequently - I use it 1 to 5 times a year
   - often – I use it 6 - 10 times a year
   - regular basis – I use it more than 10 times a year

7. **Overall impression of the system: I would use this system for virtual 3D object manipulation** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
Controller							
Cubes							

8. **Overall ease of use: The system was easy to use** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
Controller							
Cubes							

9. **The system was easy to learn** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
Controller							
Cubes							

10. **I could ROTATE the object quickly** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
Controller							
Cubes							

11. **I could ROTATE the object accurately** *

Mark only one oval per row.

	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree
Free-hand							
Controller							
Cubes							

12. **Additional comments**

BIBLIOGRAPHY

[1] Parastoo Abtahi, Benoit Landry, Jackie (Junrui) Yang, Marco Pavone, Sean Follmer, and James A. Landay. "Beyond The Force: Using Quadcopters to Appropriate Objects and the Environment for Haptics in Virtual Reality." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: ACM, 2019, 359:1–359:13. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300589. URL: http://doi.acm.org/10.1145/3290605.3300589.

[2] Laurent Aguerreche, Thierry Duval, and Anatole Lécuyer. "Reconfigurable Tangible Devices for 3D Virtual Object Manipulation by Single or Multiple Users." In: *Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology*. VRST '10. Hong Kong: ACM, 2010, pp. 227–230. ISBN: 978-1-4503-0441-2. DOI: 10.1145/1889863.1889913. URL: http://doi.acm.org/10.1145/1889863.1889913.

[3] David Anderson, James L. Frankel, Joe Marks, Aseem Agarwala, Paul Beardsley, Jessica Hodgins, Darren Leigh, Kathy Ryall, Eddie Sullivan, and Jonathan S. Yedidia. "Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling." In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000, pp. 393–402. ISBN: 1-58113-208-5. DOI: 10.1145/344779.344960. URL: http://dx.doi.org/10.1145/344779.344960.

[4] Bruno Araujo, Ricardo Jota, Varun Perumal, Jia Xian Yao, Karan Singh, and Daniel Wigdor. "Snake Charmer: Physically Enabling Virtual Objects." In: *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*. TEI '16. Eindhoven, Netherlands: ACM, 2016, pp. 218–226. ISBN: 978-1-4503-3582-9. DOI: 10.1145/2839462.2839484. URL: http://doi.acm.org/10.1145/2839462.2839484.

[5] Jatin Arora, Aryan Saini, Nirmita Mehra, Varnit Jain, Shwetank Shrey, and Aman Parnami. "VirtualBricks: Exploring a Scalable, Modular Toolkit for Enabling Physical Manipulation in VR." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: ACM, 2019, 56:1–56:12. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300286. URL: http://doi.acm.org/10.1145/3290605.3300286.

[6] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. "Haptic Retargeting: Dynamic Repur-

posing of Passive Haptics for Enhanced Virtual Reality Experiences." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: ACM, 2016, pp. 1968–1979. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858226. URL: http://doi.acm.org/10.1145/2858036.2858226.

[7] Hrvoje Benko, Christian Holz, Mike Sinclair, and Eyal Ofek. "NormalTouch and TextureTouch: High-fidelity 3D Haptic Shape Rendering on Handheld Virtual Reality Controllers." In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. UIST '16. Tokyo, Japan: ACM, 2016, pp. 717–728. ISBN: 978-1-4503-4189-9. DOI: 10.1145/2984511.2984526. URL: http://doi.acm.org/10.1145/2984511.2984526.

[8] Lonni Besançon, Paul Issartel, Mehdi Ammi, and Tobias Isenberg. "Mouse, Tactile, and Tangible Input for 3D Manipulation." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: ACM, 2017, pp. 4727–4740. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025863. URL: http://doi.acm.org/10.1145/3025453.3025863.

[9] Lung-Pan Cheng, Eyal Ofek, Christian Holz, Hrvoje Benko, and Andrew D. Wilson. "Sparse Haptic Proxy: Touch Feedback in Virtual Environments Using a General Passive Prop." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: ACM, 2017, pp. 3718–3728. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025753. URL: http://doi.acm.org/10.1145/3025453.3025753.

[10] Inrak Choi, Heather Culbertson, Mark R. Miller, Alex Olwal, and Sean Follmer. "Grabity: A Wearable Haptic Interface for Simulating Weight and Grasping in Virtual Reality." In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. UIST '17. Québec City, QC, Canada: ACM, 2017, pp. 119–130. ISBN: 978-1-4503-4981-9. DOI: 10.1145/3126594.3126599. URL: http://doi.acm.org/10.1145/3126594.3126599.

[11] Josh Close. *CsvHelper*. URL: https://joshclose.github.io/CsvHelper/ (visited on 09/26/2019).

[12] Andy Cockburn and Bruce McKenzie. "Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '02. Minneapolis, Minnesota, USA: ACM, 2002, pp. 203–210. ISBN: 1-58113-453-3. DOI: 10.1145/503376.503413. URL: http://doi.acm.org/10.1145/503376.503413.

[13] Paul Dourish. "Embodied interaction: Exploring the foundations of a new approach to HCI." In: *Unpublished paper, online: http://www. ics. uci. edu/~ jpd/publications/misc/embodied. pdf* (1999).

[14] Jinbo Feng, Isaac Cho, and Zachary Wartell. "Comparison of Device-Based, One and Two-Handed 7DOF Manipulation Techniques." In: *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. SUI '15. Los Angeles, California, USA: ACM, 2015, pp. 2–9. ISBN: 978-1-4503-3703-8. DOI: 10.1145/2788940.2788942. URL: http://doi.acm.org/10.1145/2788940.2788942.

[15] Cliff Fitzgerald. "Developing baxter." In: *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*. IEEE. 2013, pp. 1–6.

[16] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. "inFORM: Dynamic Physical Affordances and Constraints Through Shape and Object Actuation." In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. UIST '13. St. Andrews, Scotland, United Kingdom: ACM, 2013, pp. 417–426. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2502032. URL: http://doi.acm.org/10.1145/2501988.2502032.

[17] Daniel Groeger and Jürgen Steimle. "LASEC: Instant Fabrication of Stretchable Circuits Using a Laser Cutter." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: ACM, 2019, 699:1–699:14. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300929. URL: http://doi.acm.org/10.1145/3290605.3300929.

[18] Daniel Harley, Aneesh P. Tarun, Daniel Germinario, and Ali Mazalek. "Tangible VR: Diegetic Tangible Objects for Virtual Reality Narratives." In: *Proceedings of the 2017 Conference on Designing Interactive Systems*. DIS '17. Edinburgh, United Kingdom: ACM, 2017, pp. 1253–1263. ISBN: 978-1-4503-4922-2. DOI: 10.1145/3064663.3064680. URL: http://doi.acm.org/10.1145/3064663.3064680.

[19] Zhenyi He, Fengyuan Zhu, Aaron Gaudette, and Ken Perlin. "Robotic haptic proxies for collaborative virtual reality." In: *arXiv preprint arXiv:1701.08879* (2017).

[20] Seongkook Heo, Christina Chung, Geehyuk Lee, and Daniel Wigdor. "Thor's Hammer: An Ungrounded Force Feedback Device Utilizing Propeller-Induced Propulsive Force." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 525:1–525:11. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3174099. URL: http://doi.acm.org/10.1145/3173574.3174099.

[21] Anuruddha Hettiarachchi and Daniel Wigdor. "Annexing Reality: Enabling Opportunistic Use of Everyday Objects As Tangible Proxies in Augmented Reality." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: ACM, 2016, pp. 1957–1967. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858134. URL: http://doi.acm.org/10.1145/2858036.2858134.

[22] Ken Hinckley, Randy Pausch, John C. Goble, and Neal F. Kassell. "Passive Real-world Interface Props for Neurosurgical Visualization." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '94. Boston, Massachusetts, USA: ACM, 1994, pp. 452–458. ISBN: 0-89791-650-6. DOI: 10.1145/191666.191821. URL: http://doi.acm.org/10.1145/191666.191821.

[23] H. G. Hoffman. "Physically Touching Virtual Objects Using Tactile Augmentation Enhances the Realism of Virtual Environments." In: *Proceedings of the Virtual Reality Annual International Symposium*. VRAIS '98. Washington, DC, USA: IEEE Computer Society, 1998, pp. 59–. ISBN: 0-8186-8362-7. URL: http://dl.acm.org/citation.cfm?id=522258.836098.

[24] Cédric Honnet and Gonçalo Lopes. "HiveTracker: 3D Positioning for Ubiquitous Embedded Systems." In: *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. UbiComp/ISWC '19 Adjunct. London, United Kingdom: ACM, 2019, pp. 288–291. ISBN: 978-1-4503-6869-8. DOI: 10.1145/3341162.3349295. URL: http://doi.acm.org/10.1145/3341162.3349295.

[25] Matthias Hoppe, Pascal Knierim, Thomas Kosch, Markus Funk, Lauren Futami, Stefan Schneegass, Niels Henze, Albrecht Schmidt, and Tonja Machulla. "VRHapticDrones: Providing Haptics in Virtual Reality Through Quadcopters." In: *Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia*. MUM 2018. Cairo, Egypt: ACM, 2018, pp. 7–18. ISBN: 978-1-4503-6594-9. DOI: 10.1145/3282894.3282898. URL: http://doi.acm.org/10.1145/3282894.3282898.

[26] Hiroshi Ishii and Brygg Ullmer. "Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms." In: *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*. CHI '97. Atlanta, Georgia, USA: ACM, 1997, pp. 234–241. ISBN: 0-89791-802-9. DOI: 10.1145/258549.258715. URL: http://doi.acm.org/10.1145/258549.258715.

[27] Ricky Jacob, Peter Mooney, and Adam C. Winstanley. "Guided by Touch: Tactile Pedestrian Navigation." In: *Proceedings of the 1st International Workshop on Mobile Location-based Service*. MLBS

'11. Beijing, China: ACM, 2011, pp. 11–20. ISBN: 978-1-4503-0928-8. DOI: 10.1145/2025876.2025881. URL: http://doi.acm.org/10.1145/2025876.2025881.

[28] Brigitte Jordan and Austin Henderson. "Interaction analysis: Foundations and practice." In: *The journal of the learning sciences* 4.1 (1995), pp. 39–103.

[29] Ernst Kruijff Ryan P. Mcmahan Doug A. Bowman Ivan Poupyrev Joseph J. Laviola Jr. *3D User Interfaces: Theory and Practice*. Addison Wesley; 2 edition, 30 Mar. 2017.

[30] Kiyoshi Kiyokawa, Haruo Takemura, Yoshiaki Katayama, Hidehiko Iwasa, and Naokazu Yokoya. "VLEGO: A Simple Two-handed Modeling Environment Based on Toy Blocks." In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. VRST '96. Hong Kong: ACM, 1996, pp. 27–34. ISBN: 0-89791-825-8. DOI: 10.1145/3304181.3304189. URL: http://doi.acm.org/10.1145/3304181.3304189.

[31] Eun Kwon, Gerard J. Kim, and Sangyoon Lee. "Effects of Sizes and Shapes of Props in Tangible Augmented Reality." In: *Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality*. ISMAR '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 201–202. ISBN: 978-1-4244-5390-0. DOI: 10.1109/ISMAR.2009.5336463. URL: http://dx.doi.org/10.1109/ISMAR.2009.5336463.

[32] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. "Evaluation Strategies for HCI Toolkit Research." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 36:1–36:17. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173610. URL: http://doi.acm.org/10.1145/3173574.3173610.

[33] Pedro Lopes, Alexandra Ion, and Patrick Baudisch. "Impacto: Simulating Physical Impact by Combining Tactile Stimulation with Electrical Muscle Stimulation." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15. Charlotte, NC, USA: ACM, 2015, pp. 11–19. ISBN: 978-1-4503-3779-3. DOI: 10.1145/2807442.2807443. URL: http://doi.acm.org/10.1145/2807442.2807443.

[34] Pedro Lopes, Sijing You, Lung-Pan Cheng, Sebastian Marwecki, and Patrick Baudisch. "Providing Haptics to Walls & Heavy Objects in Virtual Reality by Means of Electrical Muscle Stimulation." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: ACM, 2017, pp. 1471–1482. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025600. URL: http://doi.acm.org/10.1145/3025453.3025600.

[35] I Scott MacKenzie. *Human-computer interaction: An empirical research perspective*. Newnes, 2012.

[36] Thomas H Massie, J Kenneth Salisbury, et al. "The phantom haptic interface: A device for probing virtual objects." In: *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*. Vol. 55. 1. Citeseer. 1994, pp. 295–300.

[37] John C. McClelland, Robert J. Teather, and Audrey Girouard. "Haptobend: Shape-changing Passive Haptic Feedback in Virtual Reality." In: *Proceedings of the 5th Symposium on Spatial User Interaction*. SUI '17. Brighton, United Kingdom: ACM, 2017, pp. 82–90. ISBN: 978-1-4503-5486-8. DOI: 10.1145/3131277.3132179. URL: http://doi.acm.org/10.1145/3131277.3132179.

[38] Daniel Mendes, Maurício Sousa, Rodrigo Lorena, Alfredo Ferreira, and Joaquim Jorge. "Using Custom Transformation Axes for Mid-air Manipulation of 3D Virtual Objects." In: *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. VRST '17. Gothenburg, Sweden: ACM, 2017, 27:1–27:8. ISBN: 978-1-4503-5548-3. DOI: 10.1145/3139131.3139157. URL: http://doi.acm.org/10.1145/3139131.3139157.

[39] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. "faBrickation: Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks." In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 3827–3834. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557005. URL: http://doi.acm.org/10.1145/2556288.2557005.

[40] Thomas Muender, Anke V. Reinschluessel, Sean Drewes, Dirk Wenig, Tanja Döring, and Rainer Malaka. "Does It Feel Real?: Using Tangibles with Different Fidelities to Build and Explore Scenes in Virtual Reality." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: ACM, 2019, 673:1–673:12. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300903. URL: http://doi.acm.org/10.1145/3290605.3300903.

[41] Jean Piaget. *Six psychological studies*. Vol. 4.

[42] Jenny Preece, Yvonne Rogers, Helen Sharp, David Benyon, Simon Holland, and Tom Carey. *Human-computer interaction*. Addison-Wesley Longman Ltd., 1994.

[43] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. "ROS: an open-source Robot Operating System." In: *ICRA workshop on open source software*. Vol. 3. 3.2. Kobe, Japan. 2009, p. 5.

[44] Leslie A. Geddes Raymond M. Fish. "Conduction of Electrical Current to and Through the Human Body: A Review." In: *Eplasty* (2009; 9: e44). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2763825/.

[45] Daniel Reinhardt and Jörn Hurtienne. "The Impact of Tangible Props on Gaming Performance and Experience in Gestural Interaction." In: *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*. TEI '18. Stockholm, Sweden: ACM, 2018, pp. 638–646. ISBN: 978-1-4503-5568-1. DOI: 10.1145/3173225.3173258. URL: http://doi.acm.org/10.1145/3173225.3173258.

[46] Jia Sheng, Ravin Balakrishnan, and Karan Singh. "An Interface for Virtual 3D Sculpting via Physical Proxy." In: *Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*. GRAPHITE '06. Kuala Lumpur, Malaysia: ACM, 2006, pp. 213–220. ISBN: 1-59593-564-9. DOI: 10.1145/1174429.1174467. URL: http://doi.acm.org/10.1145/1174429.1174467.

[47] Adalberto L. Simeone, Eduardo Velloso, and Hans Gellersen. "Substitutional Reality: Using the Physical Environment to Design Virtual Reality Experiences." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 3307–3316. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702389. URL: http://doi.acm.org/10.1145/2702123.2702389.

[48] Dana Slambekova, Reynold Bailey, and Joe Geigel. "Gaze and Gesture Based Object Manipulation in Virtual Worlds." In: *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*. VRST '12. Toronto, Ontario, Canada: ACM, 2012, pp. 203–204. ISBN: 978-1-4503-1469-5. DOI: 10.1145/2407336.2407380. URL: http://doi.acm.org/10.1145/2407336.2407380.

[49] Steven W Smith et al. "The scientist and engineer's guide to digital signal processing." In: (1997).

[50] Peng Song, Wooi Boon Goh, William Hutama, Chi-Wing Fu, and Xiaopei Liu. "A Handle Bar Metaphor for Virtual Object Manipulation with Mid-air Interaction." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: ACM, 2012, pp. 1297–1306. ISBN: 978-1-4503-1015-4. DOI: 10.1145/2207676.2208585. URL: http://doi.acm.org/10.1145/2207676.2208585.

[51] Gijsbert Stoet. "PsyToolkit: A novel web-based method for running online questionnaires and reaction-time experiments." In: *Teaching of Psychology* 44.1 (2017), pp. 24–31.

[52] Emanuel Vonach, Clemens Gatterer, and Hannes Kaufmann. “VRRobot: Robot actuated props in an infinite virtual environment.” In: *2017 IEEE Virtual Reality (VR)*. IEEE. 2017, pp. 74–83.

[53] Anusha Withana, Daniel Groeger, and Jürgen Steimle. “Tacttoo: A Thin and Feel-Through Tattoo for On-Skin Tactile Output.” In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. UIST ’18. Berlin, Germany: ACM, 2018, pp. 365–378. ISBN: 978-1-4503-5948-1. DOI: 10.1145/3242587.3242645. URL: http://doi.acm.org/10.1145/3242587.3242645.

[54] Andre Zenner and Antonio Kruger. “Shifty: A Weight-Shifting Dynamic Passive Haptic Proxy to Enhance Object Perception in Virtual Reality.” In: *IEEE Transactions on Visualization and Computer Graphics* 23.4 (Apr. 2017), pp. 1285–1294. ISSN: 1077-2626. DOI: 10.1109/TVCG.2017.2656978. URL: https://doi.org/10.1109/TVCG.2017.2656978.

[55] Yiwei Zhao, Lawrence H. Kim, Ye Wang, Mathieu Le Goc, and Sean Follmer. “Robotic Assembly of Haptic Proxy Objects for Tangible Interaction and Virtual Reality.” In: *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. ISS ’17. Brighton, United Kingdom: ACM, 2017, pp. 82–91. ISBN: 978-1-4503-4691-7. DOI: 10.1145/3132272.3134143. URL: http://doi.acm.org/10.1145/3132272.3134143.

[56] Kening Zhu, Taizhou Chen, Feng Han, and Yi-Shiun Wu. “HapTwist: Creating Interactive Haptic Proxies in Virtual Reality Using Low-cost Twistable Artefacts.” In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI ’19. Glasgow, Scotland Uk: ACM, 2019, 693:1–693:13. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300923. URL: http://doi.acm.org/10.1145/3290605.3300923.

DECLARATION

Ich versichere, dass ich die vorliegende Arbeit (bei einer Gruppenarbeit: den entsprechend gekennzeichneten Anteil der Arbeit) selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ich erkläre hiermit weiterhin, dass die vorgelegte Arbeit zuvor weder von mir noch von einer anderen Person an dieser oder einer anderen Hochschule eingereicht wurde.

Darüber hinaus ist mir bekannt, dass die Unrichtigkeit dieser Erklärung eine Benotung der Arbeit mit der Note „nicht ausreichend"zur Folge hat und einen Ausschluss von der Erbringung weiterer Prüfungsleistungen zur Folge haben kann.

University College London, United Kingdom, April 2019 - September 2019

Martin Feick

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". `classicthesis` is available for both LaTeX and LyX:

https://bitbucket.org/amiede/classicthesis/

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured here:

http://postcards.miede.de/

Final Version as of October 31, 2019 (`classicthesis`).