# *Bridging the Virtual World and the Physical World with Optically Dynamic Interfaces*

vorgelegt von
M.Sc.
David Lindlbauer
geb. in Linz, Österreich

von der Fakultät IV – Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

| | |
|---|---|
| Vorsitzender: | Prof. Dr.-Ing. Olaf Hellwich |
| Gutachter: | Prof. Dr.-Ing. Marc Alexa |
| Gutachter: | Prof. Ravin Balakrishnan, PhD |
| Gutachter: | Prof. Dr. Jörg Müller |

Tag der wissenschaftlichen Aussprache: 31. Mai 2018

Berlin 2018

For alex.

For my parents.

# Abstract

In the virtual world, changing properties of objects such as their color, size or shape is one of the main means of communication. Objects are hidden or revealed when needed, or undergo changes in color or size to communicate importance. Users are in full control over how the virtual world looks and behaves. With augmented reality, virtual content is overlaid over the physical world to display information. Virtual reality enables users to completely dive into artificial environments. In the physical world, however, these features are not readily available. In this dissertation, we investigate how the dynamics of the virtual world can be brought into the real world. We are interested in modifying the optical properties of physical objects and devices to foster interaction and make the physical world more dynamic.

To achieve this, we propose the concept of *optically dynamic interfaces*. Those interfaces change their appearance dynamically either through embedded capabilities or external augmentation. We describe the model of optically dynamic interfaces with three interconnected levels: dynamic objects, augmented objects and augmented surroundings. Dynamic objects are devices that integrate the capabilities to undergo visual changes. Augmented objects exhibit optically dynamic behavior through external augmentation. With augmented surroundings, objects that are unsuited for external augmentation due to their surface properties gain the ability the change their appearance. We present implementations of each of the levels and characterize their design spaces to gain insights into capabilities and limitations. Finally, we connect optically dynamic interfaces back to the virtual world with a new type of mixed reality, termed *Remixed Reality*.

# Zusammenfassung


In der virtuellen Welt ist die Veränderung von Objekteigenschaften wie Farbe, Größe oder Form eines der Hauptmittel, um mit Nutzern zu interagieren. Objekte können jederzeit versteckt oder hervorgehoben werden, oder ihre Farbe und Größe kann verändert werden, um deren Wichtigkeit zu kommunizieren. Benutzer haben völlige Kontrolle über die virtuelle Welt und können diese mit Hilfe von Software beliebig verändern. Bei Anwendungen im Bereich der erweiterten Realität (Augmented Reality) werden beispielsweise neue, rein virtuelle, Inhalte und Informationen über die reale Welt gelegt. Bei Anwendungen im Bereich der virtuellen Realität können Nutzer komplett in eine künstliche Welt eintauchen. Die reale Welt ist jedoch nicht so einfach veränderbar. In dieser Dissertation beschäftigen wir uns mit Methoden, die diese Dynamik der virtuellen Welt für Nutzer in der realen Welt verfügbar zu machen. Wir erforschen Methoden um Objekte und Geräte visuell zu verändern, mit dem Ziel Interaktionen zu vereinfachen und die reale Welt dynamischer zu machen.

Um dies zu erreichen erstellen wir das Konzept von *optisch-dynamischen Interfaces*. Diese Geräte ändern ihr Aussehen entweder durch integrierte Funktionalität oder durch externe Augmentierung. In dieser Arbeit beschreiben wir ein Model für optisch-dynamische Interfaces und drei in sich verschachtelte Ebenen: dynamische Objekte, augmentierte Objekte und augmentierte Umgebungen. Bei dynamischen Objekten handelt es sich um Geräte, die durch eingebettete Funktionalität ihr Äußeres verändern können. Augmentierte Objekte erhalten ihre optisch-dynamischen Eigenschaften durch externe Augmentierung. Durch augmentierte Umgebungen können Objekte visuell verändert werden, die aufgrund ihrer Oberflächeneigenschaften nicht für direkte Augmentierung geeignet sind. Für jede der drei Ebenen des Models stellen wir Implementierungen vor und charakterisieren ihre Vorteile und Nachteile. Schlussendlich schließen wir den Kreis von optisch-dynamischen Interfaces zur virtuellen Welt und stellen einen neuen Typ von gemischter Realität vor, in einem Ansatz den wir Remixed Reality nennen.

# Acknowledgments

I am in the lucky—and maybe unique—position of being able to thank not one but three advisors who guided my way throughout the process of this dissertation. Prof. Marc Alexa as my advisor at the Computer Graphics lab at TU Berlin. He helped me to foster the technical side of my research and kept me grounded when my ideas got a bit too crazy. Prof. Jörg Müller as my advisor during my time at the QU Lab at TU Berlin. He helped me shape the big picture of my research and continued to share his insights and feedback even after I left the lab. Prof. Michael Haller was my advisor during my studies in Hagenberg and during the first year of my doctoral studies. He introduced me to the field of HCI, and opened the gate to the world I am now part of. I would like to thank all of you for your continuing support and the way you shaped me as a researcher and as a person.

Over the years I had the pleasure of working with many great people. Everything started at the Media Interaction Lab in Hagenberg. I would like to thank my colleagues and friends Kathrin Probst, Christian Rendl, Florian Perteneder, Jakob Leitner and Thomas Seifried for many interesting discussions and fruitful collaboration. I am grateful to Prof. Mark Hancock and Prof. Stacey Scott for their help during my time at the University of Waterloo and their wisdom in study design and analysis. Furthermore, I would like to thank Yuji Uema, Toru Aoki and Prof. Masahiko Inami from Keio University for their help with Tracs.

After moving to Berlin, I had the pleasure of working with another group of amazing people. Robert Walter, as my close collaborator and dear friend, helped shape much of the work in this dissertation. Thank you for all your ideas, many of which were conceived during our countless hours playing Kicker. I would like to thank all students who supported my work: Andreas Fender, Tiare Feuchtner, Ines Ben Said, Constantin Schmidt, Dieter Eberle, Martin Schüssler, Viktor Miruchna and Klemen Lilija. I would also like to thank the Quality and Usability Lab of the Telekom Innovation Laboratories at TU Berlin, chaired by Prof. Sebastian Möller, for their support.

Finally, I moved to the Computer Graphics Lab at TU Berlin. I would like to thank my close friend and collaborator Philipp Herholz. His insights and feedback were essential to my work, as were the many hours we spent (at night in Berlin) discussing everything apart from work. Thank you to Christian Lessig for reminding me of scientific rigor and for his role as merciless proofreader, at the lab and at Bruchberg.

Furthermore, I would like to thank Andy Wilson for giving me the opportunity to work with him at Microsoft Research, and Hrvoje Benko, Eyal Ofek and Christian Holz for their feedback and interesting discussions.

None of this would have been possible without my family. I will be forever grateful to my parents Marianne and Josef Lindlbauer and my sister Anna. Thank you for making me the person I am and never doubting me. And central to all of this, my best friend and wife alex. Your wisdom, greatness and love continues to shape everything I do and everything I am for the better. Thank you.

# Contents

# Acronyms

| | |
|---|---|
| **2D** | Two-dimensional |
| **2.5D** | Two and a half dimensional |
| **3D** | Three-dimensional |
| **AC** | Alternating current |
| **AR** | Augmented reality |
| **CSCW** | Computer-supported cooperative work |
| **DC** | Direct current |
| **DLP** | Digital light processing |
| **FFC** | Flat flexible cable |
| **GPU** | Graphics processing unit |
| **GUI** | Graphical user interface |
| **HCI** | Human–computer interaction |
| **HMD** | Head-mounted display |
| **ICP** | Iterative closest point |
| **ITO** | Indium tin oxide |
| **OLED** | Organic light-emitting diode |
| **LC** | Liquid crystal |
| **LCD** | Liquid crystal display |
| **MDF** | Mechanical distance field |
| **PET** | Polyethylene terephthalate |
| **PDLC** | Polymer-dispersed liquid crystal |
| **Tracs** | TRAnsparency Controlled Screens |
| **VLT** | Visible light transmission |
| **USB** | Universal serial bus |
| **VAC** | Volts alternating current |
| **VDC** | Volts direct current |
| **VR** | Virtual reality |
| **WIMP** | ”Window, icon, menu, pointer” paradigm |

# 1 Introduction

Virtual objects, interfaces and environments are naturally dynamic and were created to adapt and fulfill users' needs. On desktop computers, for example, windows are resized based on the user's current task, and notifications are hidden or revealed on-time to allow users to react when needed. Software helps users navigate this virtual space and allows them to create and modify nearly arbitrary contents, from 2D photos and illustrations to complex 3D models.

Similarly, in augmented reality (AR), digital content is overlaid over the physical world to enrich it with contextual information. Finally, in purely virtual environments, typically referred to as virtual reality (VR), everything—including user representation and every object—affords *dynamic appearance*, since changes within this world can be performed seamlessly and mostly effortlessly.

Bringing the same dynamic properties to the physical world, however, is challenging. We are mostly surrounded by static objects, or interfaces that communicate solely through displays, i. e. windows into the virtual world. Consequently, changing the physical world, e. g. changing the size of an object, its shape, color or transparency is no easy undertaking. Furthermore, these changes cannot be performed computationally because normally there exists no connection between the physical objects and the digital world. In this dissertation, I explore and develop techniques to *bridge the gap between the virtual and physical world*, from equipping traditional WIMP devices such as displays with novel capabilities to dynamically altering the appearance of existing, previously static, physical objects.

Research in human–computer interaction (HCI) and computer graphics has acknowledged and explored the potential advantages of bringing virtual and physical objects closer together, from in-situ augmentation with spatial augmented reality (e. g. [84, 147]) to shape-changing interfaces (e. g. [77]). The contexts and means of modification and augmentation between those works differ, ranging from projecting onto objects to embedding actuators into physical objects.

I argue, however, that they share the *common goal of changing the appearance of physical objects to bridge the virtual and physical world*. The goal of my research is to provide implementations and a general framework to achieve this for a wide range of objects and situations. To accomplish this, I explored various techniques to make physical objects dynamic, i. e. giving them *dynamic appearance*.

I argue that the challenge to bridge the virtual and physical world can be approached from two different directions, from *physical to virtual* or from *virtual to physical*. In the former, physical objects are equipped with digital capabilities. Physical objects gain the ability to change their appearance (i. e. become more virtual) by building them from optically dynamic material or through external augmentation. I categorized these techniques into *the three levels of optically dynamic interfaces*: dynamic objects, augmented objects, and augmented surroundings. This dissertation develops this model of optically dynamic interfaces as well as instantiations of each level. I create dynamic objects by building them from optically dynamic material, specifically material with controllable transparency. This allows changing various properties of objects such as their perceived shape or size purely optically without any mechanically moving parts. As an instantiation of augmented objects, I enrich physically dynamic interfaces (i. e. devices that change their shape mechanically) with projection to extend their space of possible appearances. I then turn to existing static physical objects with surface properties that typically prohibit augmentation through projection, e. g. very reflective or transparent objects. To overcome this challenge, I create the concept of augmented surroundings where objects are visually altered by changing their surrounding space, thus omitting having to augment them directly.

The second direction is to go from *virtual to physical*. Given a purely virtual environment, I infuse properties of the physical world. This involves giving users the ability to see their physical environment as virtual representation as well as reintroducing tactile sensations into virtual reality. I call this approach *Remixed Reality*. I present an implementation that utilizes external cameras and an immersive virtual reality headset. Users see a live representation of their environment that is a reconstruction of the physical world. This allows users to arbitrarily change the environment and prototype interactions and devices that they may wish to re-create using optically dynamic interfaces. This approach can be seen as an extension of the general model on optically dynamic interfaces.

## 1.1 Approach

This dissertation presents four different implementations of optically dynamic interfaces: (1) a transparency-controlled 2D see-through display system (*Tracs*), (2) transparency-controlled 3D physical interfaces, (3) a system combining shape-changing interfaces with spatial augmented reality, and (4) an approach to change the appearance of existing static objects by adding virtual surroundings. Extending this work, I present *Remixed Reality*, an approach for prototyping environments that allow performing arbitrary visual changes to the space a user is in. Furthermore, the dissertation contains an initial evaluation of dynamic objects in the context of transparent displays. All these works are framed in the three-level model of optically dynamic interfaces.

The different projects allowed me to extract guidelines for bridging the virtual and the physical world, and gather insights into the specific platforms. Those insights are articulated as design spaces and taxonomies of the capabilities, and challenges, of the individual platforms. The model as well as the individual guidelines inform future research in this area and should serve as starting point for researchers who want to explore the world beyond traditional user interfaces and devices.

## 1.2 Contributions

In this dissertation, I present novel concepts and methods in terms of hardware and software to make the connection between the virtual world and the physical world more seamless. My main contributions are:

- The *concept* of optically dynamic interfaces that enables researchers to bridge the virtual and the physical world.
- Four *instantiations* of optically dynamic interfaces with detailed descriptions on their hardware and software implementation. They provide insights into the capabilities, advantages and challenges of the proposed concept. These exemplary implementations also provide insights into the different types of optically dynamic interfaces and how they can be employed.
- An *evaluation* of optically dynamic interfaces in the context of transparent displays, which are an instantiation of dynamic objects in the context of this thesis.
- A *model* for optically dynamic interfaces that allows researchers to categorize and explore ways to bring digital properties to the physical world. The model consists of three levels, dynamic objects, augmented objects and augmented surroundings. The levels aim at covering a large spectrum of concepts and technologies.
- An *extension* of optically dynamic interfaces blending virtual reality and augmented reality by means of external capture and reconstruction.

## 1.3 Publications

This dissertation would not exist without my collaborators from TU Berlin and other international institutions. Marc Alexa, as my advisor, played an essential role in this research, as did Jörg Müller and Michael Haller, who advised me before (and continued to after) I joined the Computer Graphics lab at TU Berlin. Robert Walter, Philipp Herholz and many others made this research possible with their knowledge, insights and diverse contributions. This is manifested in their co-authorship in the publications. To reflect this in my dissertation, I will use the plural form "we" to emphasize the collaborative effort. Large parts of the work presented in this thesis are based on the following conference publications:

- Chapter 4 was published and presented as *Tracs: Transparency Control for See-through Displays* [107] at UIST 2014, Honolulu, Hawaii, USA. © Owner/Author | ACM 2014. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology, UIST '14*, https://doi.org/10.1145/2642918.2647350.

- Chapter 5 were published and presented as *Changing the Appearance of Physical Interfaces Through Controlled Transparency* [110] at UIST 2016 in Tokyo, Japan. © Owner/Author | ACM 2016. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published as in *Proceedings of the Annual ACM Symposium on User Interface Software and Technology, UIST '16*, https://doi.org/10.1145/2984511.2984556.

- Chapter 6 was presented and published as *Influence of Display Transparency on Background Awareness and Task Performance* [109] at CHI 2016 San Jose, CA, USA. © Owner/Author | ACM 2016. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16*, https://doi.org/10.1145/2858036.2858453.

- Chapter 7 was presented and published as *Combining Shape-Changing Interfaces and Spatial Augmented Reality Enables Extended Object Appearance* [108] at CHI 2016 in San Jose, CA, USA. © Owner/Author | ACM 2016. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16*, https://doi.org/10.1145/2858036.2858457.

- Chapter 8 was published and presented as *Changing the Appearance of Real-World Objects by Modifying Their Surroundings* [111] at CHI 2017 in Denver, CO, USA. © Owner/Author | ACM 2017. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17*, https://doi.org/10.1145/3025453.3025795.

- Chapter 9 was published and presented as *Remixed Reality: Manipulating Space and Time in Augmented Reality* [106] at CHI 2018 in Montreal, Canada. © Owner/Author | ACM 2018. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18*, https://doi.org/10.1145/3173574.3173703.

- An overview of the work presented in this thesis was given at the doctoral symposium at UIST 2017 in Quebec City, Canada. © Owner/Author | ACM 2017. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Adjunct Publication of the 30th Annual ACM Symposium on User Interface Software and Technology, UIST '17*, https://doi.org/10.1145/3131785.3131840.

# 2

# Optically dynamic interfaces: a systematic approach

There exists a large body of work from human–computer interaction, computer graphics, and related fields that aims at bridging the gap between the virtual and the physical world. Arguably, starting with Ivan Sutherland's work on the *Sword of Damocles* [167], numerous approaches have been developed to make the physical world more dynamic, from augmented reality with projectors and displays, to shape-changing interfaces. Most of these works have been developed with a specific technology in mind or were focusing on one particular aspect of building a bridge between the virtual and the physical world.

In this dissertation, I argue that all of them serve a common goal: *to change how the physical world looks and behaves in order to make it more dynamic and bring available properties of the virtual world to the physical world.* Optically dynamic interfaces, and their underlying model, look at those approaches to identify a common theme and outline gaps and possibilities.

## 2.1 Previous conceptual frameworks

Over the past decades, researchers created various conceptual frameworks to group emerging technologies and ideas into unifying frameworks. They serve as guiding vehicles to drive further developments. This section briefly introduces frameworks that are most relevant to this dissertation. For extended reading, I refer readers to surveys by Billinghurst et al. [18] and earlier work by Azuma [7], which provide in-depth discussion on technologies, applications and research directions.

### Reality-virtuality continuum

While development on augmented reality and virtual reality had started in the 1960s [167], Milgram and Kishino [123] were among the first to conceptually link the technologies in their seminal work on the *Reality-Virtuality (RV) continuum*, illustrated in Figure 2.1. Their conceptual framework covers display technologies within a spectrum of *Mixed Reality*, where the notion of physicality governs the position of a technology on the spectrum. If a technology allows users to see a virtually enriched or extended physical world, it is located closer to the real environment (further left in Figure 2.1). Conversely, if a technology occludes the real world, e. g. through an immersive head-mounted display, it is located closer to virtuality, i. e. further right in Figure 2.1. The spectrum can also be seen with respect to how much of the world is recreated, from completely unmodeled (real world) to completely modeled (virtual environment), as noted by the authors.

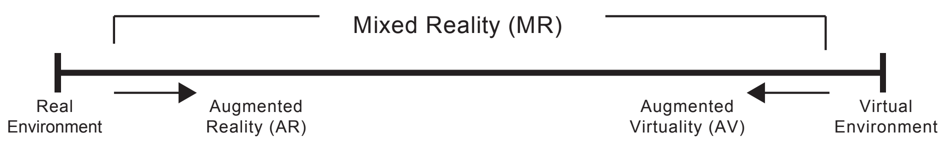


**Figure 2.1:** The Reality-Virtuality continuum introduced by Milgram and Kishino (reproduced from [123].)

### Mediated reality

Mann [115, 116] introduced the concept and multiple implementations of *Mediated Reality*. He focused on perceptual extensions and modifications of the physical world. The notion of mediation stems from the idea that any technology can alter how humans perceive the physical world. As an example, he envisioned giving users superhuman vision by wearing different types of lenses. Like Milgram et al., he sees the mediation with respect to how much of the physical world is presented to users. The relation between the different usages can be seen with respect to how the different types of technologies (e. g. augmented reality, virtual reality) overlap (see Figure 2.2, *right*) or as a matrix with reality, virtuality and mediality (see Figure 2.2, *left*). In the former, augmented reality and virtual reality are seen as special cases of mediated reality, whereas the latter specifies how much the physical world is (perceptually) altered.

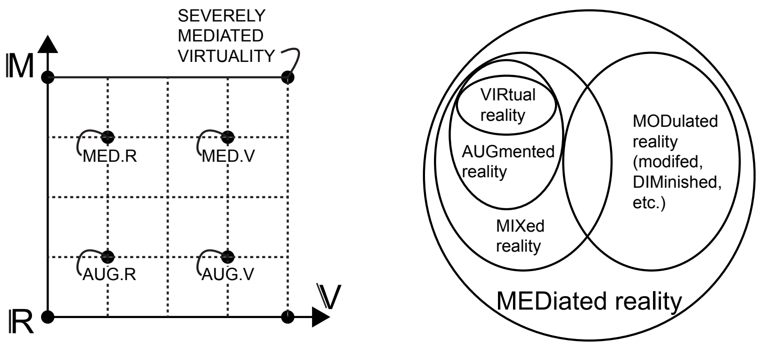


**Figure 2.2:** Two representations of mediated reality by Mann (reproduced from [115, 116]).

### Augmented reality: Linking real and virtual worlds

Mackay [113] frames the connection between virtual and physical space with respect to sensing technologies. She identified three approaches where augmentation and sensing takes place: at the users, at the physical objects, or in the environment. For example, users wear data gloves to control projection, objects are augmented with sensors for ubiquitous computing [187], or the environment is equipped with microphones to enable speech input.

### One reality

Among the more recent examples of conceptual frameworks for augmented reality falls the work of Roo et al. [154] on *One Reality*. They focus on the *transition* between different levels of augmentation. On the one hand, users can leverage the physical world with tangible input devices. On the other hand, users can choose to perform actions in virtual reality. They identify various levels between those two extrema, e. g. by augmenting surfaces with projection mapping. The notion of transition is crucial for a seamless and effortless user experience.

### Radical atoms

Besides work in the field of *visual* augmentation, there exists a large body of work on physically dynamic interfaces, or *shape-changing interfaces*. With their early work on Tangible Bits [76] and Radical Atoms [77], Ishii et al. strive at bringing the virtual and physical world closer together. They focus on equipping devices with actuators such as motors and pneumatics to make them dynamic and allow them to be controlled computationally by users or systems. By moving beyond traditional graphical user interfaces (GUIs), they believe a new type of human–computer interaction will manifest itself that allows for *seamless* interfaces. To achieve this, Ishii et al. envision advanced materials that allow objects to deform, merge, and split freely, and allow for any kind of input and output [26].

### Discussion

All aforementioned models opened way for research in the fields of augmented reality, mixed reality and tangible user interfaces under the common theme of bridging the gap between the digital and the physical world. Milgram et al. [123] focused on displays and general visual output technologies. Their horizontal model allows researchers to identify where specific technologies lie on the spectrum. The model, however, is not concerned with *how* the virtual and the physical world can be bridged. Mann [115–117] seeks to modulate human visual perception through display technologies. By developing different novel psychophysical phenomena, they wanted to explore how users react to dramatic changes in visual sensation. This helped them to gain insights into how a virtually distorted physical world might look like and how it would influence users. Their model mostly looks at what I call *inside-out approaches*, i. e. users are wearing displays to induce changes. They did not investigate *outside-in approaches*, i. e. changing the physical world through integrated displays or projection, for example. Mackay [113] took a holistic approach creating a model with levels that are conceptually related to my model of optically dynamic interfaces. Her work focused on interaction with objects and equipping users, objects or environments with input and output devices. Conversely, the focus of this dissertation lies on how visual properties can be altered dynamically. Roo et al. [154] looked at the *transition* between reality and virtuality. They focused on existing technologies and less on how objects can be altered. Ishii et al. [77] aimed at unifying input and output of interactive systems. They are concerned with embedding digital capabilities by means of novel materials and fabrication methods. Radical atoms aims at *tangible* output to go beyond purely visual output. I argue that in many situations, visual output is sufficient—sometimes even superior—to induce the illusion of completely dynamic physical objects. However, my model is also applicable to physically dynamic interfaces, as shown in Chapter 7. I strongly believe that the concepts are mutually beneficial.

All these models have helped to advance the state of the art. I believe, however, that they do not cover the complete space when aiming at bridging the gap between virtual and physical, and leave several key questions unanswered:

- Which objects can be manipulated?
- How can objects be manipulated?
- Can objects be manipulated *without* instrumentation of the user?
- Is it necessary to create *physically dynamic* objects in order to achieve a change in how an object is perceived?

These are the main the questions I seek to answer in this dissertation. Furthermore, I argue that any model that is concerned with this topic should contain *interconnected* levels. Most types of augmentations are not distinct but can actually build on top of each other. To reflect this, I created the following model for optically dynamic interfaces.

## 2.2 A model for optically dynamic interfaces

The goal of optically dynamic interfaces is to bring readily available (dynamic) properties of the virtual world to the physical world. Such properties include the ability to change size, visibility, color, or shape. These changes can be performed with a variety of technologies. The type of manipulation that can be achieved is defined by the object that is altered and the technology that is used. Conversely, the combination of desired manipulation and type of object defines the technology. This connection is illustrated in Figure 2.3.

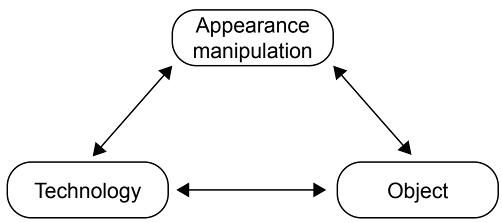


**Figure 2.3:** Optically dynamic interfaces rely on the interconnection between perceptual manipulation, technology and object (i. e. the manipulation target) .

### 2.2.1 Definition and categorization

Inspired by Azuma's survey on augmented reality [7], I define optically dynamic interfaces based on the following criteria:

- **Criterion 1.** An optically dynamic interface *changes its appearance to manipulate how it is perceived by users* with respect to geometric properties such as size or shape and/or surface properties such as texture or transparency.
- **Criterion 2.** An optically dynamic interface exists in the physical world, therefore it is not a purely virtual object.
- **Criterion 3.** The perceptual manipulation that an optically dynamic interface undergoes can happen through embedded capabilities, external augmentation of the object, or manipulation of the environment.
- **Criterion 4.** Optically dynamic interfaces can be either physically static or dynamic, i. e. there exist combinations of optically dynamic interfaces and physically dynamic interfaces.

In order to categorize the technologies and their ability to perform changes, I created a model of optically dynamic interfaces, illustrated in Figure 2.4. The model is composed of three levels: *dynamic objects*, *augmented objects*, and *augmented surroundings*. This dissertation focuses on these three aspects of how to adapt and manipulate physical interfaces to make them optically dynamic. I built several prototypes of such interfaces with dynamic appearance, from transparency-controlled displays to objects that undergo perceptual changes through modifying their surrounding space. The common theme of all these devices is that they exhibit *dynamic appearance*. The levels differ with respect to several aspects such as manufacturing process, suitability for projection, or integrated output capabilities.

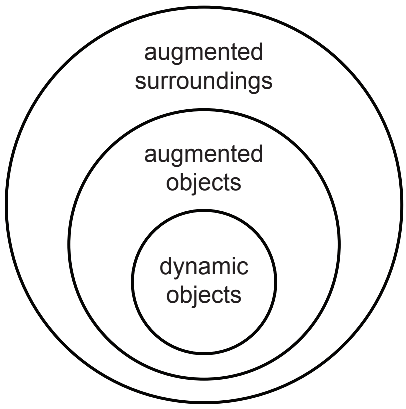


**Figure 2.4:** The different levels of optically dynamic interfaces, i. e. how to create objects and interfaces that exhibit *dynamic appearance*.

Note that augmentation for each level can be achieved with a wide range of technologies. The levels serve as guidelines for *where and how* augmentation can take place: if an object is intrinsically altered (i. e. dynamic objects), if augmentation happens *directly* on objects (i. e. as visual overlay for augmented objects) or if objects are augmented *indirectly* by changing their surrounding space. In this dissertation, I use material with controllable transparency for dynamic objects, projection for augmented objects, and displays for augmented surroundings. The alterations can be achieved with different technologies, e. g. integrated displays, smart materials, or augmentation through head-mounted displays.

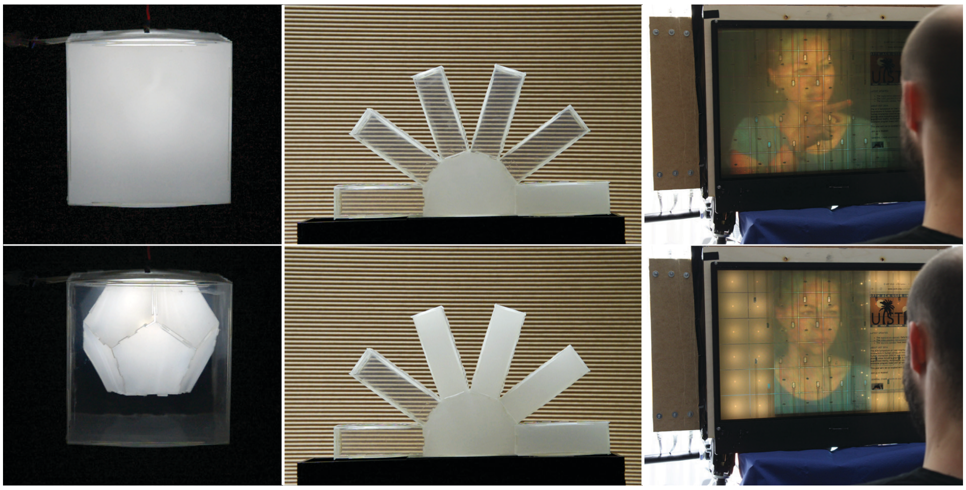

**Figure 2.5:** Examples of dynamic objects. *Left* shows a dynamic lamp shade, *center* a transparency-controlled progress indicator and *right* shows the transparency-controlled display. All those examples are built from optically dynamic material that allows them to change their transparency.

### 2.2.2 Dynamic objects

I define dynamic objects as interfaces that incorporate the ability to change their (optical) properties by creating them from optically dynamic material, such as the examples in Figure 2.5. They are manufactured with the desired functionality in mind. This gives them the ability to change their transparency or perceived size, for example. Objects can also include displays or, as in the field of shape-changing interfaces, include motors or pneumatic actuators. This means that dynamic objects can also be combined with physically dynamic interfaces. Dynamic objects require no instrumentation of users or the environment. This means, however, that changing an object's range of possible alterations requires modifying its underlying hardware.

In this dissertation, I manufacture objects from smart materials, i. e. the transparency-controlled physical interfaces introduced in Chapter 5 or the transparency-controlled see-through display system from Chapter 4.

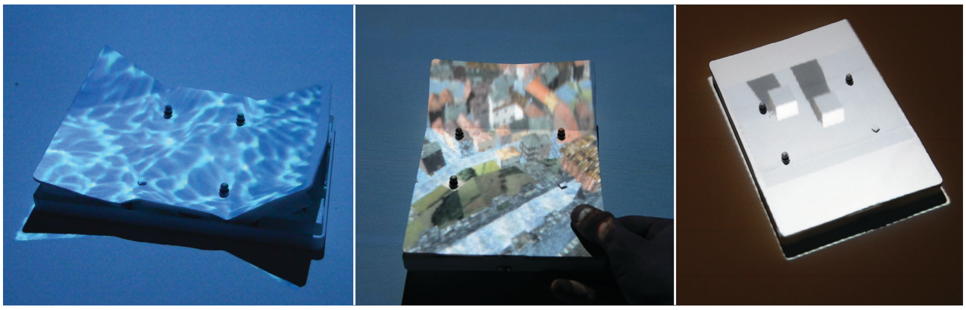

**Figure 2.6:** Example of an augmented object. The physically dynamic tablet is visually extended with projection mapping.

### 2.2.3 Augmented objects

I define augmented objects as interfaces that are visually altered through *external augmentation*. They do not include any ability to undergo visual manipulations, thus rely on technologies such as projection. This enables changing the color or decreasing the size of existing physical objects, for example. Typical examples include work from the field of spatial augmented reality (e. g. [8, 16, 83, 84, 108, 147]). In contrast to dynamic objects, the range of possible manipulations is not defined by integrated materials. This makes augmented objects very flexible and enables a broad range of manipulations that can be varied over time. The range, however, is constrained by the properties of objects such as geometry, transparency or reflectivity. Objects with diffuse surfaces can easily be augmented due to the capabilities of current projectors. Transparent or very reflective and dark objects on the other hand pose significant challenges. Furthermore, since augmented objects are usually altered by projecting light onto them, any augmentation is constrained by the physical bounds of the object since current projection technologies cannot easily display high-fidelity visual content in mid-air.

In this dissertation, I explore augmented objects in the context of physically dynamic interfaces, detailed in Chapter 7, shown in Figure 2.6. While a large body of work exists that augments static objects, I am interested if already dynamic objects can benefit from augmentation.

**Figure 2.7:** Examples of augmented surroundings. By changing the surrounding space, the objects are visually altered. The cup, glass, book and the cube are physical objects that are placed on a display. They undergo perceptual changes with respect to their contour, color and size, respectively. Note that also the shadow is rendered to increase realism.

### 2.2.4 Augmented surroundings

I define augment surroundings as the concept of performing visual alterations without modifying the target objects themselves, i. e. they are altered *indirectly*. Similar to augmented objects, augmented surroundings do not rely on integrated capabilities to alter an object's appearance. In contrast to augmented objects, however, augmented surroundings do not require any projection onto target objects directly. Instead, the environment that surrounds objects is equipped with displays (e. g. objects are placed on a display) or is visually altered through projection. By displaying graphics that are edge-aligned with a target object, for example, users can perceive a range of changes such as increase in size, change in contour, or change in color, shown in Figure 2.7. Augmented surroundings are very suitable for objects with surface properties that are challenging to augment with projectors, e. g. transparent or highly reflective objects.

This dissertation presents the first work that utilizes this approach to change properties such as color and size of a unmodified target object by placing it on a display, as detailed in Chapter 8.

### 2.2.5 Discussion

The three levels of optically dynamic interfaces enable researchers to explore ways to integrate digital properties and capabilities into the physical world as well as categorize prior work. In the next chapter, I discuss existing related work from various fields with respect to this proposed model.

It is noteworthy that an interface does not necessarily needs to fall into one distinct category. Dynamic objects can be enriched with external augmentation if the built-in functionality is not sufficient to cover a desired scenario, i. e. an object can be dynamic and augmented at the same time. Similarly, objects can be directly and indirectly enriched, i. e. they are augmented objects within augmented surroundings.

Technologies such as displays, projectors or smart materials can be used as enablers for all levels. This means that the levels are not tailored to one particular technology. Projection can be used to create augmented objects as well as to create augmented surroundings. Chapter 7 introduces an approach where I create augmented objects with projection. The same technology is used for an extension of augmented objects where also the surrounding space is altered to hide an object. Furthermore, interfaces can transition from one level to another. An augmented tablet is introduced in Chapter 7. By replacing the external augmentation with internal displays, the tablet essentially becomes a dynamic object. This emphasizes that the transition between the three levels can be fluid, which arguably makes the model more universal than existing frameworks.

# 3

# State of the art

This chapter introduces related work that is most relevant to the general topic of this dissertation, as well as to the individual technologies and concepts. We discuss the work with respect to the previously introduced model of optically dynamic interfaces with its three levels: dynamic objects, augmented objects, and augmented surroundings.

## 3.1 Dynamic objects

Dynamic objects have the embedded ability to change their appearance, e. g. by building them from smart materials. Therefore, they can change properties like their transparency, color or texture, which in turn leads to changes in how they are perceived by users. We take work on general physical interfaces into account, as well as work on optically dynamic WIMP interfaces, specifically transparent displays, which is relevant to the transparency-controlled see-through display system introduced in Chapter 4. We furthermore discuss relevant related work from the field of shape-changing interfaces, which share properties with optically dynamic interfaces, although they are built with dynamic haptics in mind.

There exists only a sparse set of works that focus on changing the appearance of objects by embedding optically dynamic components. Most of these works focus on reproducing complex surface properties by means of fabrication. Alexa and Matusik [2] fabricate objects with different microstructures that yield appearance changes under different viewing angles. Schüller et al. [157] create viewport-dependent bas-reliefs. Furthermore, there is a large body of work on controlled surface reflectance (bidirectional reflectance distribution function, BRDF, e. g. [43, 71, 118]). This allows creating objects with dynamic surface properties. In contrast to work on changing the reflectance functions on objects, Olberding et al. [134] create PrintScreen, which is used for augmenting objects with printed displays. While the work focuses on producing objects that communicate with users through graphics like conventional displays, their approach could be used to create objects with dynamic appearance by wrapping an object with PrintScreen to change the color of the whole object, for example. The PrintScreen technology is based on electroluminescent material, which has also been employed for interactive paper (e. g. [47, 92]) and on-skin interfaces (e. g. [186]). With printed optics, Willis et al. [190] integrated optical elements in their fabricated objects through 3D printing which, among other things, allow to dynamically change the appearance of object.

We introduce an approach to alter the appearance of objects by building them from material with controllable transparency. None of the above approaches is able to render an object transparent. Simulating transparent surfaces with projection mapping requires techniques such as projecting a video of the environment onto the object. This would then fall in the category of augmented objects. Even then, though, image quality (e. g. contrast, latency) would potentially be too low for creating a high quality illusion of transparency. Our approach does not require any instrumentation or tracking, since it allows altering an object's transparency independent of viewing position, and without any knowledge of the background.

### 3.1.1 Physically dynamic interfaces

Besides altering objects through optical manipulation, there exists a broad body of work on physically dynamic interfaces, usually referred to as shape-changing interfaces [77, 148], organic user interfaces [69, 183] or actuated user interfaces [143]. They are designed and created to alter their physical shape for fulfilling a variety of aims, such as extended functionality (e. g. communication [70], dynamic affordances [45]), hedonic aims (e. g. aesthetics [195]), conveying emotions (e. g. Thrifty Faucet [175]), or for exploration (cf. [77, 148]) and data physicalization (e. g. [82]). We briefly review the most important concepts and technologies in this field. For a broader review we refer readers to the surveys by Rasmussen et al. [148], Coelho et al. [33], and Sturdee and Alexander [166], which provide taxonomies of shape-changing interfaces and discuss different types of change, interactions, and goals.

Different strategies and technologies are used to integrate computational capabilities into everyday objects. Shape displays, as a prominent example of shape-changing interfaces, are used as input and output devices, e. g. for remote collaboration and telepresence (e. g. [45, 100, 142]). They are created with a multitude of individual (mostly linear) actuators, for example Relief [99], InFORM [45] or Transform [78]. Alternatively, shape displays have been implemented with technologies such as shape-memory alloys [142]. While these displays enable rendering dynamic material properties [132], for example, their resolution and speed are limited by physical constraints such as speed and density of the actuators.

Other shape-changing interfaces based on technologies such as pneumatics, particle jamming or shape-memory alloys exist. In contrast to the linear actuators of shape-displays, these technologies are more suited for making general objects dynamic since their shape can oftentimes be designed freely. With PneUI [195], the shape of general objects is altered with pneumatics to achieve functional (e. g. game controller) or hedonic aims (e. g. shape-changing lamp). Olberding et al. [135] created shape-changing devices with printed electronics through folding. BioLogic [196] uses natto cells, which can be described as a naturally occurring shape-memory alloys, to create dynamic wearable devices.

Roudaut et al. [155] quantify the resolution of shape-changing interfaces based on non-rational uniform basis splines (NURBS). As discussed by Rasmussen et al. [148] and Roudaut et al. [155], most research focused on topologically equivalent changes of shape, such as change in volume (e. g. [70]), or texture (e. g. [100, 195]). Other, non-topologically equivalent changes (e. g. adding holes to a surface) are less explored, arguably because they are hard to achieve with current technology (cf. [155]). Coelho et al. [32], as an example, changed the permeability of a surface for architectural purposes.

In this dissertation, we provide two different takes on shape-changing interfaces. In Chapter 5, we create interfaces that change their perceived shape purely optically, without any mechanically moving parts. This allows us to overcome limitations such as actuation speed and create interfaces that perform changes in permeability. Our work essentially creates the *illusion* of altering an object's shape. This makes our approach suitable e. g. for distant interaction and complementary to existing work on shape-changing interfaces. In Chapter 7, we propose an approach combining physical and optical changes. We enrich shape-changing interfaces with spatial augmented reality for extending the space of appearances of actuated interfaces. This work also aims at creating interfaces that can dynamically alter their appearance.

### 3.1.2 Optically dynamic WIMP interfaces

In this dissertation, we are concerned with the creation of general objects that exhibit dynamic optic behavior as well as how to apply those properties to traditional WIMP interfaces. Specifically, we introduce a novel type of transparency-controlled display. In the following, we review relevant related work from research on transparent displays as well as the transparent and transparency-controlled WIMP interfaces in general.

#### Transparent displays

Transparent displays, i. e. displays where users can see through to look at the environment or other users behind the displays, are generally implemented as video see-though displays or optical see-through displays. They have been identified as one way in co-located collaboration to overcome challenges with mutual gaze, specifically they allow users to perform an on-screen tasks while *simultaneously* retaining gaze awareness (cf. [75]). This way of face-to-face communication has been exploited by Tang and Minnemann with VideoDraw [173] and VideoWhiteboard [172] as well as by Ishii and Kobayashi with Clearboard-1 [75]. They emphasized the importance of integrating content directly into the communication channel to allow users to switch focus and task smoothly, which was enabled by using video see-through technology. Hirakawa et al. [68] used a see-through display as collaborative work environment in combination with augmented reality. Olwal et al. [136, 137] used an interactive dual-sided FogScreen for multi-user face-to-face collaboration, giving users the ability to collaborate closely while being able to see and interact with screen content. With MUSTARD [88] and PiVOT [87], Karnik et al. explored the possibility to target contents at specific users, giving them the ability to see contents collaboratively or exclusively. HoloDesk [67] and SpaceTop [98] use an optical see-through display for 2D and 3D spatial interactions behind the display. In Transwall, Heo et al. [64] display equal content on both sides of a transparent display for facilitating interpersonal communication, mostly through gaming. A similar setup has later been used by Kim et al. [91] for simple collaborative tasks such as drawing. Li et al. [103, 104] used a dual-sided projected transparent display for collaborative work. They discuss the affordances and possibility of such setups, with enhancements in workspace awareness being fundamental advantages of transparent displays.

While transparent displays offer advantages in terms of gaze awareness, challenges of distraction and visual rivalry arise. Insights from research on interface transparency (e. g. [11, 57, 61]) show decreased performance of task execution (e. g. selection or menu interaction) with transparent user interfaces. For transparent displays, Laramee and Ware [96] showed that visual rivalry between screen content and background leads to decreased performance and readability.

In this dissertation, we make two main contributions to the area of transparent displays. We first contribute Tracs, a two-sided display system with locally controllable transparency. Secondly, we perform a controlled experiment showing that transparent displays offer comparable performance to horizontal displays for a dual-task scenario while outperforming conventional displays, even when the conventional displays are rotated and moved to provide users with an unoccluded view on the environment they are required to observe.

### Transparency-controlled interfaces

There exists a small body of work that created transparency-controlled interfaces, mostly in the context of architecture. Rekimoto [152] developed programmable physical architecture, where areas of a wall can be toggled between transparent and opaque using a large-scale grid of switchable diffuser. They made use of the interplay between transparency and opacity, also by conveying information with switchable diffuser (essentially using it as pixels) without additional devices or displays (except camera tracking), and also used their system for dynamic shadow generation. In contrast to Rekimoto's work, our work on Tracs focuses on creating a dual-sided see-through display system which uses the property of adjustable transparency. Danninger et al. [39] created Attentive Office Cubicles, which change from opaque to transparent when users on both sides of a cubicle choose to interact with each other. Kakehi [86] created "transmart minispaces", an installation that resembles a multi-layer display. In art, Chen and Oliviera create x.pose [194], a dress made of patches of switchable diffuser. The number of patches that turn transparent increases with user's increased social media activity.

All these works rely on switchable diffuser as optically dynamic material, which we use in this dissertation for building the transparency-controlled physical interfaces and the transparency-controlled see-through display. Apart from transparent displays and general transparency-controlled interfaces, the material has also been used for camera see-through systems (e. g. [53, 93]) or for projecting through surfaces [80]. Apart from switchable diffuser some materials exist that undergo changes in transparency when actuated, e. g. thermoresponsive hydrogel (cf. [185]), which continuously turns from transparent to opaque when heated. While hydrogels are well suited for example for tactile feedback in interactive devices as employed by Miruchna et al. [124], they have prohibitively long activation times and are harder to manufacture than switchable diffuser.

## 3.2 Augmented objects

Augmented objects overlap with and include classical augmented reality applications and technologies. Research in HCI, computer graphics, computer vision and others have long been striving to overlay virtual information on top of the physical world or modify our perception of reality by means of visual manipulation. As discussed on Chapter 2, there exist multiple models and taxonomies that group existing work with respect to technological and conceptual features. In this section, we discuss relevant related work to this dissertation. An overview of augmented reality and adjacent concepts, which exist since the mid 1960s (e. g. Sutherland et al. [167]) and have evolved ever since, is beyond the scope of this thesis. For an overview we refer readers to the summary of augmented reality applications and concepts by Billinghurst et al. [18] and an earlier survey by Azuma [7].

Among the technologies closest to our work is projector-based augmentation, or spatial augmented reality (cf. [19]). Projecting light onto a known physical objects enables changing its appearance, for example adding shadows for virtual depth or changing its color. The virtual content is thereby *registered* to the real objects, which allows for 3D perception while exploiting natural depth cues. Projection mapping requires calibration and background compensation (e. g. [54]). Furthermore, to create 3D effects that work for more than one viewpoint, the position of a user has to be tracked. This, however, usually only works for a single user. We use an interpretation of the term spatial augmented reality where also displays mounted on a surface can serve as 3D virtual extension of the object (referred to as fish tank VR, e. g. pCubee [164]). No other instrumentation such as see-through displays is needed, since head tracking is used. Alternatively, the projection closely resembles the physical object, also eliminating the need for head tracking. Note that while we focus on projected augmented reality in this dissertation, many of the effects could also be achieved with head-mounted optical and video see-through devices such as the Microsoft HoloLens [122] or commodity smartphones (e. g. work by Möhring et al. [126]).

Raskar et al. [147], as an early example, developed Shader Lamps, which virtually enhances 3D objects through projected textures. Techniques for projection mapping (e. g. calibration, compensation for background [54]) are also applied for spatial augmented reality (cf. [20]). These techniques are used for example to support communication (e. g. [37]), urban planning, and media architecture design (e. g. [5]). With IllumniRoom, Jones et al. [84] used projection mapping to create room-scale experiences, in this case using the projection as an extension of a high-resolution screen. In RoomAlive, Jones et al. [83] used room-scale projection mapping for immersive gaming experiences that can take place on any surface that is suitable for projection. Bonanni et al. [27] augmented a kitchen with projection and framed their work in the context of hyper-reality [25]. By exploiting knowledge of the geometry, physical effects such as gravity can be virtually reproduced, as by Benko et al. [14] with MirageTable. Furthermore, since geometry is known, projection mapping is suitable for equipping nearly arbitrary objects with interactive capabilities, as shown by Benko et al. [13] with their work on spherical displays. Using movable projectors as in Beamatron [191] expands the space of interactions and reduces the need for multiple projectors to cover a whole room. With Illuminating Clay, Piper et al. [140] augmented a *manually* deformable surface with volume data through projection mapping. Hettiarachchi and Wigdor [66], while focusing on haptic feedback, overlaid physical objects with graphics through a head-mounted display. Tangible 3D Tabletop [38, 58] combines tangible tabletop interaction with 3D projection to augment tangible objects with visual content corresponding to their physical shape and position. This means that the flat tabletop surface together with the tangible objects becomes a deformable interface.

Among the applications of projection mapping in art we find the works of Valbuena. His installations use projection mapping to create optical illusions, e. g. Augmented Sculptures [181], typically with clear-cut geometrical shapes . AntiVJ's installation Enghien [5] employed projection mapping to create the illusions of the transformation of a building façade. The non-interactive installation first mimics moving light sources to emphasize the 3D effect; then, starts copying and apparently moving the physical, architectural features, such as windows and balconies; finally, it deconstructs these features of the building (cf. [37]).

### 3.2.1 Diminished reality

In contrast to typical augmented reality scenarios where the physical world is enriched with virtual information, diminished reality (or *illusionary transparency*, cf. [115–117]) aims at the *removal of physical objects* from users' field of view. This oftentimes requires extensive knowledge of the environment a user is in, for example through depth cameras and reconstruction as in SceneCtrl [197]. Alternatively, objects can be removed from users' sight computationally, for example through inpainting [65]. Inami et al. [73] use a retro-reflective projection technology to enable users to see through objects (e. g. invisibility cloak, transparent back of car) by projecting the environment (captured with a camera) onto a special fabric. Iwai and Sato [79] rendered objects on a desk transparent by projecting underlying contents onto them. In this dissertation, we create the illusion of transparent objects by projecting onto objects and the environment simultaneously (see Chapter 7), as well as by projecting content that is similar to an object's color in their surrounding space (see Chapter 8).

### 3.2.2 Augmented reality with physically dynamic objects

While many works focus on augmenting static surfaces, there exists research on augmenting physically dynamic objects. Bermano et al. [16] augmented an animatronic head with virtual textures. They exploit the dense correspondence between the physical avatar and the virtual shape to increase realism. In the field of shape-changing interfaces, prior work focused mostly on the combination of shape-changing interfaces with 2D graphics, such as caller information or maps on shape-changing mobile phone (e. g. MorePhone[50]).

Alexander et al. demonstrated interactions with a grid of small actuated (tilt) displays [3], whereas we focus on using perspective-corrected 3D graphics. Leithinger et al. enriched their 2.5D displays with virtual graphics through height field data [100] and see-through augmented reality [101]. We extend this line of research by displaying perspective-corrected 3D graphics directly on shape-changing interfaces. In our approach, we display *arbitrary* 3D objects and visuals onto shape-changing interfaces.

## 3.3 Augmented surroundings

With augmented reality, objects are typically enriched *directly*, i. e. virtual content is projected directly onto objects, or content is shown *standalone* as mid-air content, for example. Conversely, augmented surroundings refers to the concept of *indirectly manipulating* how a users perceives an object. This can be achieved by displaying contents around or in close proximity of a target object. Most prior work focused on creating virtual shadows to convey information. With Urp [179] and I/O bulb [178], Underkoffler et al. augmented real-world objects with shadows for conveying information and to enrich architectural models. This was developed in the context of the Luminous Room project [180], which aimed at unifying input and output of virtual contents in the real world, mostly through projection. On the same line, Naemura et al. [130] and Moon [127] projected virtual shadows of real-world objects for conveying a virtual light source.

Our work aims at extending objects, i. e. their appearance, through more complex augmentations. By incorporating illusions such as extended size and dynamic contours we argue that the space of possible interactions is greatly extended. This allows us to create effects that are typically challenging to achieve, such as enlarging objects as well as augmenting objects with surface properties unsuited for projection (e. g. reflective or transparent surfaces).

#### Visual illusions

Besides research from the fields of human–computer interaction and computer graphics, our work is inspired by work on visual illusions. They are one way to indirectly change the perception of a target object. The Ebbinghaus illusion, as an example, triggers a change in perceived size of a target by surrounding it with differently-sized inducers. We refer readers to different categorizations of visual illusions such as the work from Gregory et al. [51] and Changizi et al. [30], which we took as inspiration for exploring various effects such as size and color. Most classical illusions such as Ebbinghaus, Hering (perceived line bend) or Chubb (change in perceived contrast) work well with simple 2D stimuli such as circles and lines. However, from our experience, the rather small effects observed even when using 2D stimuli (the effect magnitude of the classical Ebbinghaus illusion is around 20%, cf. [128]) vanish when employed with more complex real-world objects, which we elaborate on later in Chapter 8. Nonetheless, from a broader point of view, we use these categorizations and create effects based on 3D renderings for dynamically altering e. g. the contour or perceived position of objects, as outlined in Chapter 8 with our work on virtual surroundings.

# 4

# Tracs: Transparency control for see-through displays

## 4.1 Introduction

Displays are dominating our office environments. They provide crystal-clear images of the contents presented on the screen, but block the view on the environment behind the display. Consequently, these displays are a constant visual barrier, thus a communication barrier. Additionally, they do not allow for face-to-face communication while seeing screen content, one of the main affordances of transparent displays [103, 104]. Even office workers sitting across a table have to either stand up or walk around the table to see each other. With increasing display sizes and decreasing costs [153], this problem might be exacerbated in future. Additionally, when all users need to see the same screen content, only shoulder-to-shoulder collaboration is possible, making gaze-awareness impossible (cf. [75]).


**Figure 4.1:** Tracs is a dual-sided transparency-controlled see-through display system to avoid visual interference on transparent displays and to support fast switching between personal work and collaboration. Users can control the transparency of specific parts of the display (*left*) or overall (*right*).

Transparent displays offer users the ability to see screen content as well as the environment behind the display. They have been explored for co-located collaboration [75, 103, 104, 137, 172, 173], exposing beneficial features like workspace awareness. However, conventional (non-transparent) displays still are ubiquitous. We believe this is because transparent displays lack basic features of conventional displays such as shielding from visual disturbances from the environment behind the display. In order to make see-through display devices more usable, we believe it is important to allow users to change the transparency of their display on-demand. This way, visual interference can be decreased and users can regain privacy by turning the display opaque.

In this chapter, we present Tracs (TRAnsparency Controlled Screens), a dual-sided transparency-controlled display system. With Tracs, users can control the transparency of individual areas of their display, as shown in Figure 4.1. Tracs is *one instantiation of dynamic objects* in the context of optically dynamic interfaces. Instead of augmenting a display device with video cameras to enable video see-through behavior, we manufacture the device with the later functionality (controllable transparency) in mind. This transforms a previously "static" device, with the exception of on-screen content, into an optically dynamic device. This means that users can change optical properties of the device to better fit their needs.

Tracs consists of two see-through displays enclosing a transparency-control layer, a backlight layer, and a polarization adjustment layer. The transparency-control layer is a grid of transparency-controllable patches and is created from a single piece of polymer dispersed liquid crystal (PDLC) switchable diffuser with additional pieces of transparent indium tin oxide (ITO) film for controlling the individual patches. The transparency of the PDLC diffuser is controlled by adjusting the voltage supplied to it. Switchable diffuser has been used in research previously, e. g. by Rekimoto [152] for architectural purposes and as an information display. In contrast, we use the material in combination with see-through displays and screen content, respectively, to open new possibilities for personal usage of and collaboration with digital content. Tracs' backlight layer, a LED matrix mounted on transparent ITO, improves the contrast of the see-through display. The polarization adjustment layer is added to enable users to see through the LCDs.

## 4.2 Implementation

Tracs gives users the ability to toggle the transparency of individual areas on the display through its transparency-control layer (see Figure 4.1). These areas support more fine-grained adjustments for users and to cover a broader range of use-cases and applications, e. g. to allow users to create collaborative (transparent) and personal (opaque) regions on their display simultaneously (see Figure 4.9). Additionally, this enables users to block visual interference locally without having to turn the whole display opaque.

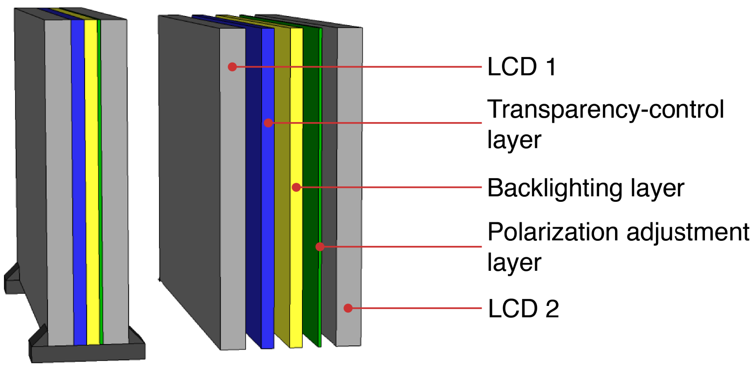


**Figure 4.2:** Schematic of the dual-sided screen system, consisting of two transparent LCDs, a transparency-control layer, a transparent backlight and a half-wave retardation film for adjusting polarization.

### 4.2.1 Hardware components

Tracs consists of a stack of five hardware components (see Figure 4.2). We use two transparent 22" LCDs (Samsung LTI220MT02) as a basis for displaying content. Each display offers a transparency of about 15% according to the specification. In-between the two displays, Tracs includes a transparency-control layer which allows users to manually control the transparency of the displays. Furthermore, we included a backlight layer to increase the contrast of the LCDs and a polarization adjustment layer to improve the transparency of the displays.

### 4.2.2 Transparent displays

Transparent LCDs rely on ambient light for displaying screen content. They are transparent when turned off or displaying non-black content. Since they do not emit light, contrast is limited compared to e. g. transparent OLED displays. We chose displays over projection to make Tracs self-contained and to cover a broad range of co-located collaborative use cases (e. g. office context with workers facing each other). Since we used transparent LCDs in our setup, we equipped Tracs with an additional backlight and polarization adjustment layer.

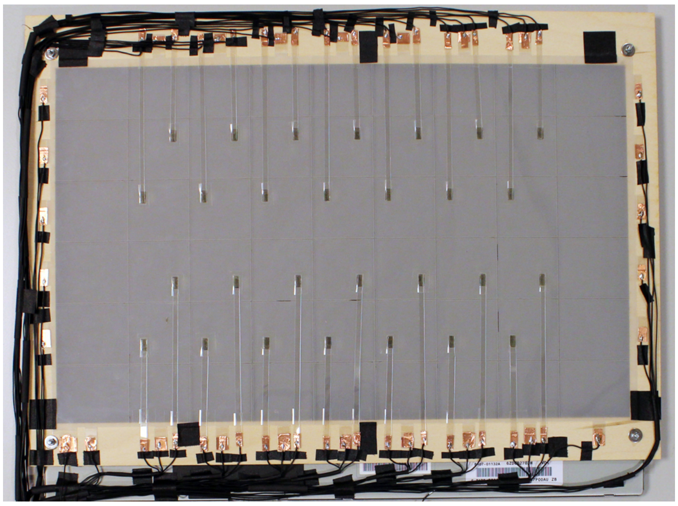

**Figure 4.3:** Our current 22" prototype of the transparency-control matrix. Each of the 54 patches can be controlled individually.

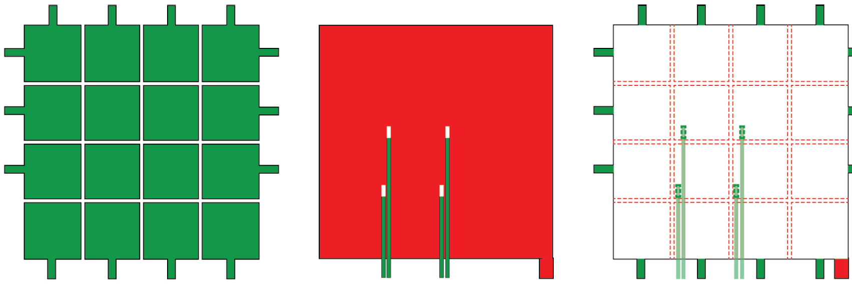


**Figure 4.4:** Schematic of a 4 × 4 transparency-control matrix: cathode layer (*layer C, left*) with individual patches, common anode layer (*layer A, center*) and *A* and *C* combined (*right*). Both layers have connectors for wiring (areas outside of grid). We added additional strips of ITO on top of the anode layer (*center*, light green) to connect inner patches.

### 4.2.3 Transparency-control layer

Our current prototype consists of a 9 × 6 matrix of individually controllable patches, each with a size of 5 cm × 5 cm. We constructed the matrix from a single piece of PDLC switchable diffuser (see Figure 4.3). This material is composed of two layers of ITO and one layer of polymer dispersed liquid crystals. The conductive sides of the ITO layers are facing each other, with the crystals in between. When no voltage is applied, the material is diffuse (60% visible light transport transmission, VLT, 90% haze), and becomes clear (80% VLT, 10% haze) when voltage is applied (60 volts). One ITO layer serves as anode (*layer A*), the other as cathode (*layer C*). Our transparency-control matrix is a passive matrix layout with a common anode.

We first laser engrave *layer C* of ITO in a grid-like manner to create the individual patches (see Figure 4.4, cuts are dashed red lines *right*, resulting patches *left*). Note that in order to disconnect the individual patches from each other, each line consists of two cuts (distance 1 mm). The ITO between the two cuts is removed manually to avoid current from flowing. The cuts only go through *layer C, layer A* is left intact (see Figure 4.4, *center*). Damaging the conductive coating of *layer A* side would not allow us to use it as a common anode. For controlling the patches in *layer C* individually, we connect them to the cables running at the outside. Therefore, we laser engrave 5 mm × 10 mm holes into *layer A* (see Figure 4.4, *right*).

Subsequently, we applied thin separate strips of ITO from the outside to the holes and connect the strips with the holes, *layer C* respectively. 3M™ Z-Axis electrically conductive tape 9703 is used for connecting the two layers of ITO. Patches at the edges are connected with small additional areas on the outside to avoid cutting holes. These transparent ITO connections in combination with the common anode from *layer A* allow us to control the transparency of each patch individually.

In our initial implementation, the patches were controlled with eight Texas Instruments TPIC6B595 high voltage 8-Bit shift registers, which can handle load up to 60 volts DC, connected to a micro controller. Controlling the switchable diffuser with DC instead of AC, however, lead to visible material fatigue in the switchable diffuser. The material lost in visual clarity over time (i. e. decreasing clarity with increasing number of activations) since the material is meant to be controlled with AC. Therefore, in a second version of control circuitry, we resorted to a combination of optocoupler (Fairchild MOC3021) and triacs (Philips Semiconductors BT-138S). Each patch is connected to one pair of optocoupler and triac, which are switched with multiple shift registers (Texas Instruments 74HC595) through an Arduino UNO micro controller (see Figure 4.5).

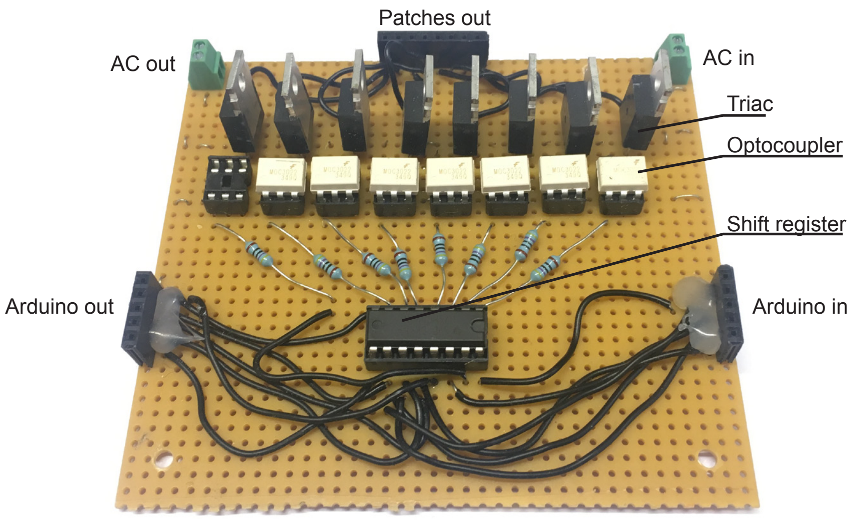


**Figure 4.5:** Circuit for switching a total of 8 patches.

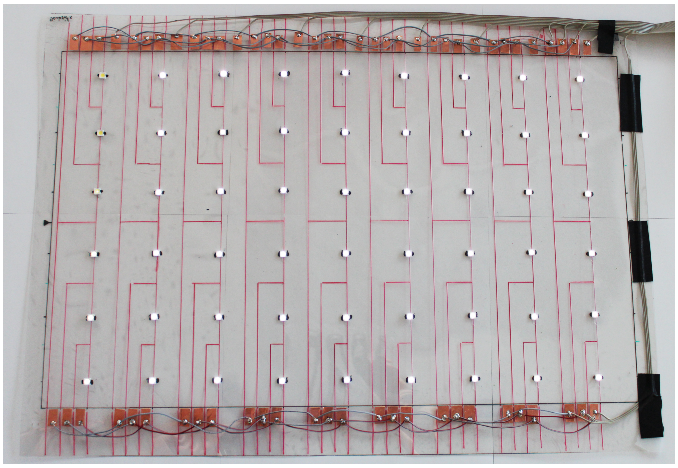

**Figure 4.6:** Tracs' backlight matrix is composed of LEDs mounted on ITO and controlled via a passive matrix addressing scheme. The red lines highlight the engraving in the ITO and are not visible for users.

### 4.2.4 Backlight layer

We constructed the Tracs backlight matrix by mounting LEDs on top of a piece of ITO (80% VLT, 2% haze) using wire glue. In order to being able to control the LEDs individually, we used a passive matrix addressing scheme, with individually addressable rows and columns. All lines for anodes and cathodes were engraved into ITO (Figure 4.6, red lines) to construct the matrix from a single layer of transparent material.

### 4.2.5 Polarization adjustment layer

Each LCD used for Tracs has a linear polarizer film (orientation 45°) applied to their front- and backside in order for them to work. Since the two LCDs are polarized orthogonally once positioned back-to-back, the displays would be completely opaque. In order to overcome this issue, we included an additional layer of transparent half-wave retardation film (American Polarizers APHW92-003-PC-280NM, 92% VLT) in-between the two displays (see Figure 4.7). The film acts as a polarization rotator by shifting the phase of the input light by $\pi$, rotating the incoming linearly polarized light by 90°, respectively, cf. Figure 4.2 [63].

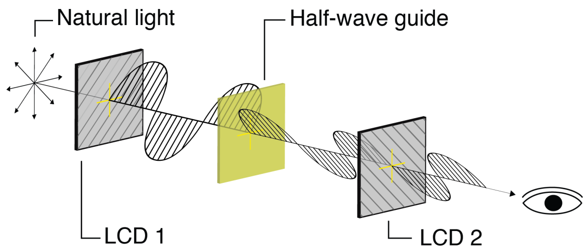


**Figure 4.7:** The polarization is adjusted using a half-wave retardation film positioned between the screens. Without the film, users could not see through the display, since the displays are polarized orthogonally.

## 4.3 Usage and applications

Tracs offers users with three basic usage states, which are (1) personal, (2) collaboration and (3) communication (see Figure 4.8). Each state is possible on either the full display or on a specific area (see Figure 4.9). In personal state, Tracs acts like a regular display, preventing visual interference and privacy. During collaboration, the display is transparent and the screen content visible. This way, users can relate to items visible on the screen just by looking at them and establish a shared focus. During the communication state, the display is transparent and no screen content visible, allowing users to communicate freely. Tracs' custom control software allows users to implicitly select the different states through controlling the transparency of patches and screen content.

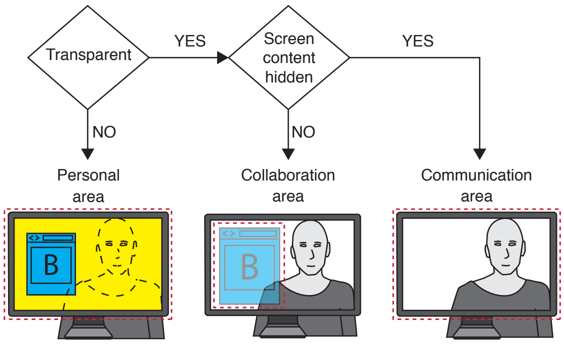


**Figure 4.8:** Tracs' three states: personal (opaque, *left*), collaboration (transparent, screen content visible, *center*), and communication (transparent, screen content hidden, *left*).

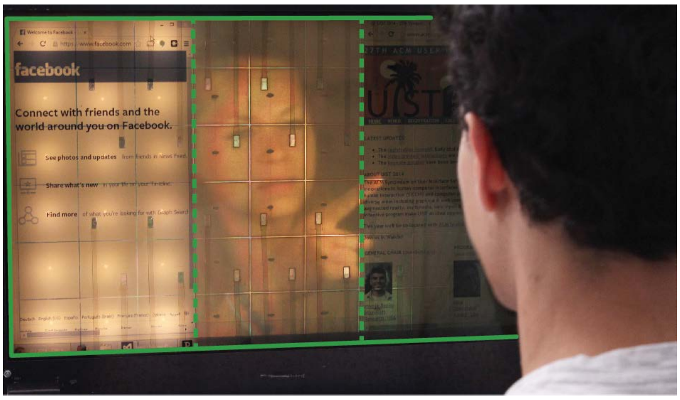


**Figure 4.9:** Tracs' states combined on a single display, personal (*left*), communication (*center*), collaboration (*right*).

## 4.4 Discussion

We believe that by being able to control transparency (thus privacy) and visual interference, the limitations of see-through displays can be overcome and new ways of collaboration be opened. Transparent displays could bring together benefits from shoulder-to-shoulder collaboration (shared perspective on screen content) and face-to-face collaboration (eye-contact) [74, 103]. Like reducing spatial distance can increase collaboration [94], we believe that removing the barrier introduced by traditional displays can make collaboration more instant and effective. However, we believe users need to be able to choose between transparent and opaque to increase to usability and acceptance of transparent displays.

### 4.4.1 Quality and transparency

Currently, Tracs uses transparent LCDs, which require additional environmental light to provide a good view through the display. Otherwise transparency would be limited. Transparent OLED displays do not suffer from such limitations, but their commercial availability is limited. Tracs is designed in a way to work with many types of see-through displays because of its flexible layering. The transparency-control layer can applied to an OLED to equip it with controllable transparency. The most influential factor for the Tracs' transparency are the LCDs, which give it a approximate overall transparency of 10%. This is a limiting factor for potential deployment and needs to be resolved.

Our current backlight emits light only at one side, and, while PDLC diffusely reflects the backlight also to the other side, there is a decrease in brightness and some minor reflections. To improve contrast from both sides, a second backlight layer could be added or a transparent electroluminescence display emitting light at both sides could serve as backlight.

Currently, the LCDs are 5 cm apart, so the LEDs illuminate the correct area and the switchable diffuser is not directly on the display, which results in a parallax effects. While in our experience this does not decrease Tracs' practical usability (e. g. when pointing), showing overlapping contents on both displays is currently not supported. In terms of ergonomics, our displays have to be positioned in a way that users are able to see through them without having any edges within their view. Feedback we received also addressed the possibility to tilt the display for a more comfortable viewing angle. Both points are important for transparent displays in general and need to be taken into account.

### 4.4.2 Switching and symmetry

While Tracs gives users the possibility to switch between transparent and opaque states, it is yet unclear, which one users prefer. Users might only switch to transparent mode in case of actual collaboration and not during daily work, therefore decreasing workspace awareness. However, we believe that by making switching effortlessly, users would be encouraged to collaborate more frequently. Since users can switch to transparent state all the time, these switches could occur in moments not wanted for one of the users. Switching is currently negotiated verbally by users. Being able to prevent unwanted switches, especially to transparent mode, is important for systems like Tracs.

# 5

# Transparency-controlled physical interfaces

## 5.1 Introduction

Chapter 4 introduced transparency-controlled see-through displays (Tracs), which are one instantiation of dynamic objects. In this chapter, we apply the same principles to more general objects beyond conventional WIMP devices. We describe *transparency-controlled physical interfaces* that, like other dynamic objects, are built with their future functionality in mind. These interfaces change their *appearance* (i. e. perceived shape) by changing their transparency. This enables toggling the visibility of parts of objects for communicating information or for hedonic purposes. Transparency-controlled physical interfaces enable the *illusion* of changes in shape, volume, or the appearance of holes, as well as nesting of

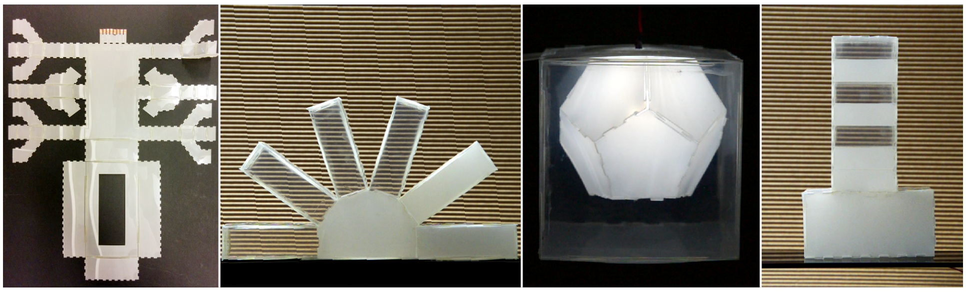

**Figure 5.1:** We create transparency-controlled physical interfaces by cutting and folding objects from a single sheet of material with controllable transparency (*left*). Parts of an interface are controlled individually, resulting in *the illusion* of changed shape. We demonstrate multiple examples, such as a notification indicator (*2nd left*), an appearance changing lamp (*3rd left*) and a physical progress bar (*right*).

objects. Appearance changes rapidly (with our current implementation ~8 ms to turn transparent; ~80 ms to turn opaque) and does not involve any mechanically moving parts. This approach requires no instrumentation since the ability for optical change is built into the device.

Transparency-controlled physical interfaces extend and complement other types of dynamic interfaces, such as physically dynamic interfaces. Shape-changing interfaces adjust their physical shape to match users' desires and needs. Current devices such as shape-changing mobile phones (e. g. [50, 155]) or shape displays (e. g. [45, 99, 142]) are limited in the rate of change as well as the type of possible changes: speed is limited by physical constraints of the actuators; topological changes (e. g. creating holes) are difficult due to the limited ability of motorized interfaces to shrink in volume. While transparency-controlled physical interfaces change their appearance (i. e. *simulated* shape change), they do not exhibit any dynamic tangible qualities. We see them as complementary to current shape-changing interfaces whose physical shape truly changes. We argue that by focusing on dynamic physical *and* optical properties, the space of possible interactions can be expanded. Transparency-controlled physical interfaces are well suited in situations where no tangible qualities are needed (e. g. user is distant to the device). Furthermore, they enable features such as creating apparent holes or nesting of objects, which are challenging to achieve with other technologies.

We create appearance changing devices by laser cutting the desired shape from a single sheet of PDLC switchable diffuser. This material transitions between opaque and transparent rapidly when voltage is applied and requires very little energy. To create 3D objects, we cut and fold the switchable diffuser in an origami-like manner. We include hinges that support arbitrary angle bends to avoid breaking the electrical connection between individual faces.

Our aim is to create objects that are seamless and require no physical support such as frames, or additional wiring. We eliminate the need for external frames by incorporating snap-fit connectors into the design of the optically dynamic 3D elements. To control individual parts of an interface, we route voltage through channels that are engraved in the switchable diffuser. Engraving and cutting are performed in a single fabrication step. Manufacturing an object only requires laser cutting its shape, folding and connecting to a standard flat flexible cable (FFC) connector. No wiring or soldering is needed.

We demonstrate a semi-automatic method for creating transparency-controlled physical interfaces. Our software analyses a given crease pattern and automatically adds hinges and snap-fits. When the user positions the connector, our software automatically adapts snap-fits and hinge patterns. Creating individually controllable parts is done by marking the respective areas and drawing paths for routing.

We present four physical interfaces demonstrating our concept. We describe two different appearance-changing activity indicators: a vertical volumetric progress bar and a flower-shaped notification indicator with actuated leaves. By toggling individual parts, progress can be indicated, or certain appearances can be generated for hedonic purposes. We also present a playful bug-shaped avatar, with legs reacting to displacement and exhibiting apparent motion. The switchable diffuser wraps around and extends a computer mouse for creating the desired shape with included displacement measurement. Lastly, we created a dynamic lamp shade with 3 different appearances, shaped like a cube, a dodecahedron or a cone, nested within each other. These shapes can be toggled based on user's hedonic desires or for notifications.

**Contributions**

- We provide a conceptual addition to conventional shape-changing interfaces through transparency-controlled physical interfaces. We explore the design space of these interfaces to outline their benefits and challenges.
- We demonstrate a simple production process for creating 3D transparency-controlled physical interfaces through origami-like folding, with electrical routing being included in the created objects. They require no additional mounts or wiring and consume little energy.
- We present algorithms for semi-automatic generation of 3D transparency-controlled physical interfaces based on a given crease pattern.
- We showcase four physical interfaces demonstrating the versatility of our concept.

## 5.2 Design space

In this section, we describe the design space of transparency-controlled physical interfaces. We take inspiration from previous work on shape-changing interfaces [108, 131, 148, 155] and non-traditional displays [134].

We develop the design space with respect to materials such as switchable diffuser, which means objects are composed of individually controllable planar pieces. While our implementation focuses on objects created from switchable diffuser, we believe the design space generalizes to other technologies that allow rendering parts of an object transparent, such as flexible transparent displays. Therefore, we consider our design space as largely technology agnostic. It aims at providing insights into capabilities and potential challenges of transparency-controlled physical interfaces.

We identified five dimensions that serve as a foundation for designing transparency-controlled interfaces, see Figure 5.2, i. e. face, object, composition, control area, and control type.

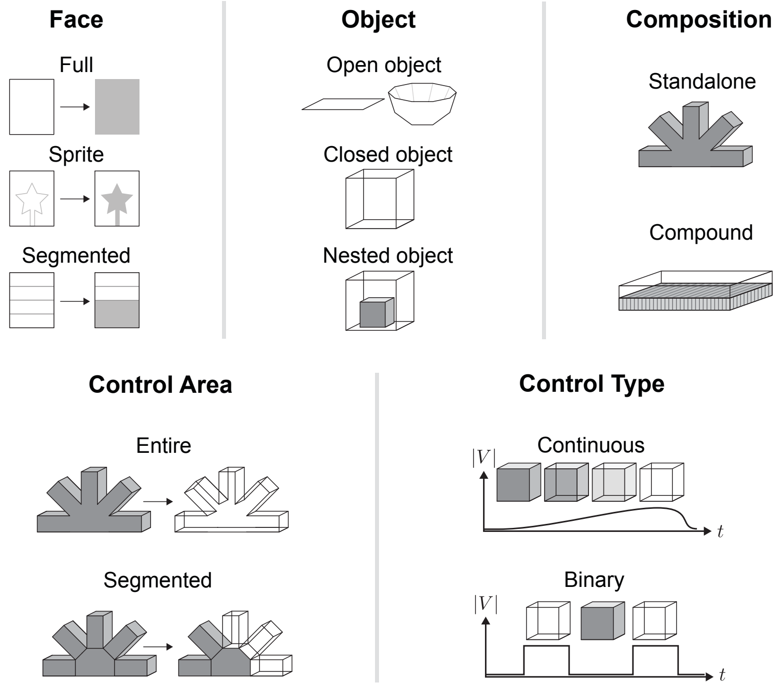


**Figure 5.2:** Design space for transparency-controlled physical interfaces. Gray areas indicate opaque state, dashed areas indicate external objects.

### 5.2.1 Face

A face is an individual side of a transparency-controlled object that can change its overall transparency. Objects are typically composed of multiple faces. Groups of faces can be controlled simultaneously. By including iconic shapes in a face, predefined *sprites* can be toggled, shown in Figure 5.2 and Figure 5.3. Thus, not only the appearance of an object but also icons on the surface can be used for communication and information. Sprites can also be used to render apparent holes (e. g. users can see through the engraved star in Figure 5.3). Furthermore, a face can be visually subdivided by *segmenting* it. Thus it can appear to be smaller or larger (as shown e. g. in Figure 5.1, 5.2 and 5.5), depending on the state of the segments.

**Figure 5.3:** A cube with an engraved sprite (star) in all 4 possible states.

### 5.2.2 Object

Objects are typically composed of multiple faces. We distinguish between 3 types of objects.

#### Open objects

Open objects do not enclose a volume, e. g. planes or the open bowl illustrated in Figure 5.2. Switchable diffuser, in its original shape, is a planar sheet with limited bending capabilities. It can be used as-is and included as individual surfaces in other physical interfaces (as *compounds*, discussed later). Open objects also allow creating transparency-controlled objects that serve as *containers*. For example, a transparency-controlled medication box could turn transparent at specific times to remind users about taking their medication.

#### Closed objects

Closed objects enclose a volume, e. g. the cube in Figure 5.3. We create 3D objects through cutting and folding, which allows us to create relatively complex transparency-controlled objects (e. g. our flower-shaped progress indicator consists of 41 individual faces). The complexity of the objects with our current origami-like manufacturing process is limited by a minimum size of planar pieces. Keeping manufacturing in mind, folding objects with a side length smaller than 1 cm is challenging and the hinge patterns cut in the material decrease visual quality. Furthermore, while it is theoretically possible to create highly complex shapes through origami (cf. [40, 171]), human folding capabilities are typically limited. Objects can also include hingeless bends if supported by the material.

### Nested objects

One key feature of transparency-controlled objects is the ability of *nesting*. Smaller objects can be included *within* larger ones and revealed through toggling the enclosing object's transparency. This is typically not available or challenging to achieve for other types of physical interfaces. Figure 5.4 shows an example of three nested cubes. Nested objects can be used for switching between appearances (e. g. between a cube and an enclosed pyramid), e. g. for representing iconic shapes. For us, while a nested object appears to be composed of multiple objects, they share the same surface since they are created from a single sheet of material. Thus, we see them as an addition to open and closed objects.

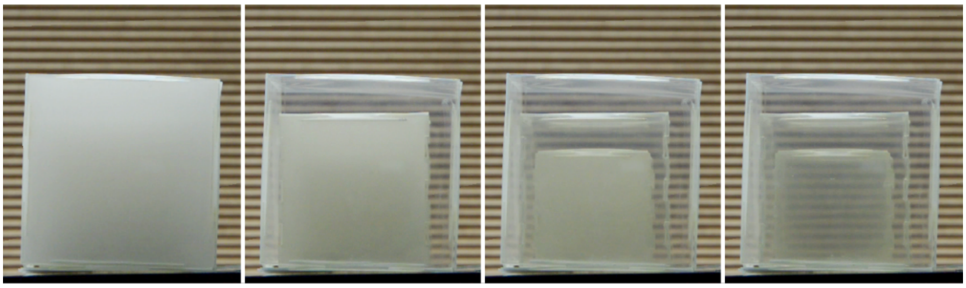

**Figure 5.4:** A nested object consisting of three cubes.

## 5.2.3 Composition

We refer to a composition as the final transparency-controlled interface, which is built from one or more faces and can be described as open, closed or nested. A composition can be created purely as transparency-controlled interface (*standalone*), or as a *compound*. A compound is a regular object (physically static or dynamic), enriched with one or more transparency-controlled objects.

One example of a compound composition is the actuated mobile phone shown in Figure 5.5. We adopted interactions typical for these devices (e. g. actuated flaps for notification, cf. [50, 155]) and extended it with a segmented transparency-controlled top surface. Including transparency-controlled segments into objects allows rendering features such as holes, for example for applications such as tracing objects underneath a device (e. g. Glassified [161]). By controlling the transparency of parts of a compound object, interior parts can be hidden or revealed (e. g. for privacy or hedonic purposes), or users can see through objects. This enables applications such as teaching mechanical functionality by revealing the inside of a priorly opaque device.

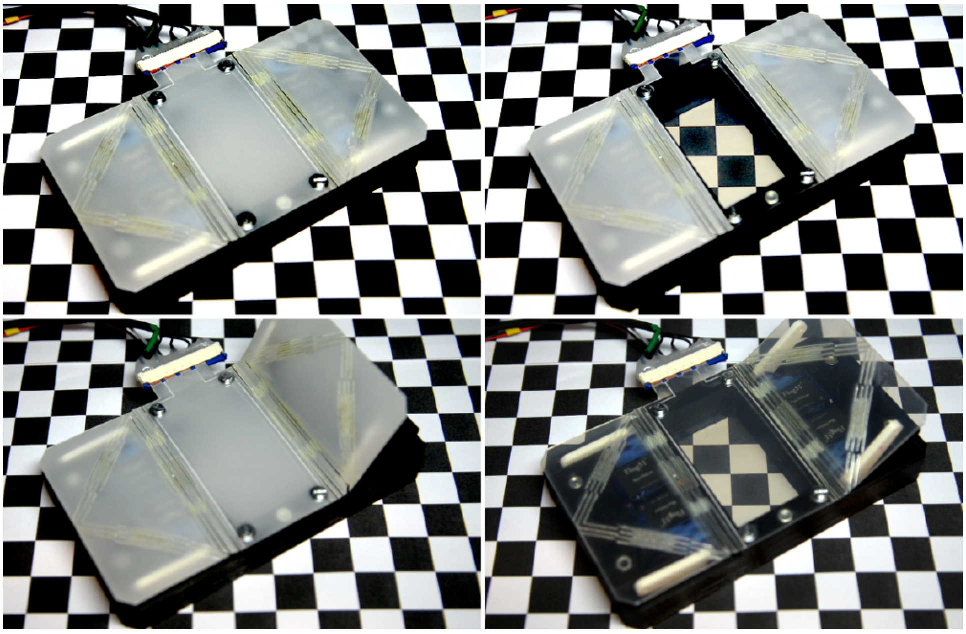

**Figure 5.5:** A shape-changing mobile phone with a transparency-controlled top surface (inspired by [108, 149]), modified to omit components in the center of the device; *top left*: base state, *bottom*: actuated state. Individual parts can be turned transparent (*right*).

### 5.2.4 Control area

We can control the entire surface of an interface, for example to reveal it only when needed. Furthermore, transparency-controlled interfaces with multiple controllable parts can be created for enlarging the space of possible applications and increasing expressivity (e. g. Figure 5.13). Controllable parts can be individual or segmented faces, or larger areas (multiple connected faces) of an object. For nested interfaces, the individual objects can be toggled to reveal interior objects (e. g. as in Figure 5.15).

### 5.2.5 Control type

#### Binary optical change

Switching transparency-controlled physical interfaces is performed rapidly, since the change is purely optical. This enables effects such as apparent motion. Negative effects of mechanical motion such as oscillations around a target position are avoided. Such switching speeds can be exploited for different applications, e. g. fast reaction to changes in underlying data for data visualization.

#### Continuous optical change

Besides binary switching, transparency-controlled physical interfaces can change their appearance continuously and arbitrarily slow. This allows changing the appearance potentially even without users noticing it in their peripheral vision and can be exploited, for example, for unobtrusive peripheral display applications.

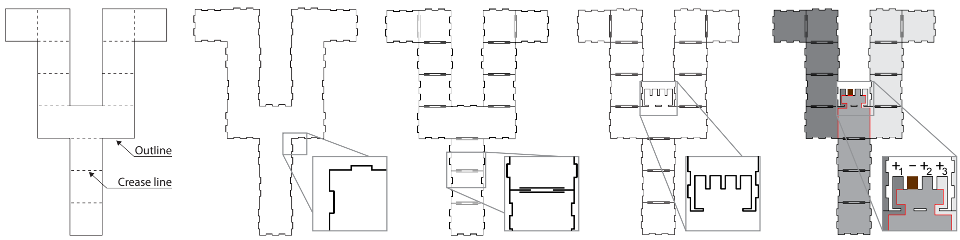


**Figure 5.6:** Our process for creating laser cut plans for 3D transparency-controlled physical objects from an input crease plan (*left*). We generate snap-fits for connecting edges, and hinge patterns to allow for bending without breaking the electrical connection. Thereafter, the connector and the electrical routing is added. Red lines in the right image indicate cutting only one layer of ITO for separating the individually controllable surfaces. Gray areas indicate individually controllable areas. The resulting 3D object, a box with 3 controllable multi-face parts, is displayed in Figure 5.11, *right*.

## 5.3 Implementation

In this section, we describe the creation of transparency-controlled physical interfaces. Software and hardware are available as open source[1].

### 5.3.1 Base material and individually controllable surfaces

We use PDLC switchable diffuser (Kewei Films Non-Adhesive Smart Glass) for creating 3D objects. The material consists of two layers of transparent conductive film (ITO) with polymer dispersed liquid crystals in-between and an insulating layer on the outside (see Figure 5.7). In default state, switchable diffuser scatters incoming light and thus looks diffuse. When voltage is applied (60 VAC), it turns transparent (90% parallel light transmittance according to specification). We control the switchable diffuser using custom circuitry (a combination of optocoupler and triacs, connected to shift registers). This allows for rapid toggling with low energy consumption (~10 mA/m$^2$ in transparent state, no power consumption in opaque state). Our current material exhibits switching speeds of ~8 ms to turn transparent and ~80 ms to turn opaque. PDLC, in transparent state, acts like a capacitor, resulting in lower switching speed from transparent to opaque. Different circuitry can decrease switching speeds (e. g. as in SecondLight [80] to 8.3 ms symmetrically). It can also be continuously controlled by varying the input voltage (e. g. applying 20 VAC results in less transparency than 40 VAC; approx. linear change). We control this through potentiometers in a voltage-divider circuit.

For applying voltage, the two layers of ITO have to be wired on the inside. This is because ITO only conducts on the inside due to the isolating polyethylene terephthalate (PET) layer, as illustrated in Figure 5.7. To create multiple controllable surfaces on a single object, we separate one ITO layer (top layer in Figure 5.7) and wire the resulting sides separately. The layer on the opposite side remains intact serving as a common electrode. This process is similar to Tracs (Chapter 4), however, we eliminate additional wires through engraving routes directly in ITO. This

[1] http://www.cg.tu-berlin.de/research/projects/transparency-controlled-physical-interfaces/

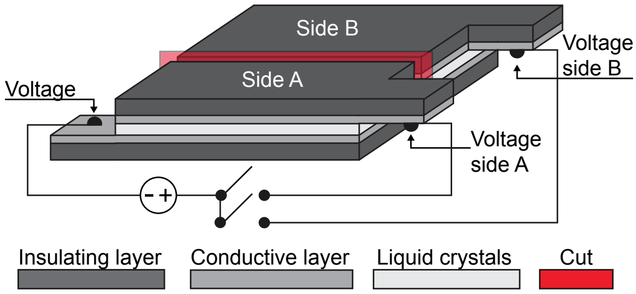


**Figure 5.7:** We separate individually controllable surfaces by cutting through one conductive layer. This allows us to control them separately, here side A and B. The cut (red) only goes through the top insulating and conductive layer, leaving the lower part (common electrode) intact.

simplifies production and improves the visual clarify of objects, since our method results in less constantly transparent areas than for Tracs. Furthermore, external wiring of nested objects is not always feasible.

### 5.3.2 3D objects from switchable diffuser

We decided to use an origami-like folding technique for creating 3D objects to avoid using external frames and to create objects that have few seams.

#### Folding

We describe our process as ”origami-like”, since classical origami starts from a rectangular piece of paper and does not allow additional cuts. Furthermore, we add snap-fit connectors to edges and partly disconnect individual pieces through cutting. This avoids overlapping faces (typical for traditional origami created from a single sheet), which would reduce the transparency of objects.

#### Processing

Creating a foldable 3D objects from switchable diffuser is a four step process, see Figure 5.6 for an overview. We describe the steps in the logical order of the processing pipeline. Processing starts with an input crease pattern, created in e. g. Adobe Illustrator or specialized software.

**Snap connectors** We first alter the outline of the input crease pattern to include snap connectors. This is done automatically in our custom software, described later. The snap connectors allow for connecting faces after folding to obtain objects which do not require additional frames. Figure 5.8 shows the cut switchable diffuser with snaps as outline and the resulting folded cube.

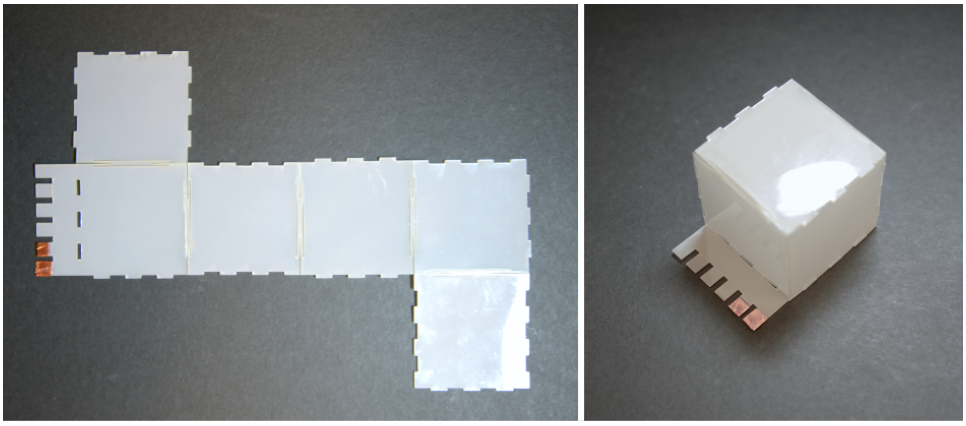

**Figure 5.8:** Left: laser cut folding pattern; right: the resulting object.

**Hinges** In the next step, hinges are added, which are essential for creating foldable objects. The hinges are laser cut patterns (*hinge patterns*), yielding different bending properties. This solution was inspired by classical engineering, which refers to this as living hinge. If switchable diffuser is bent without cutting the hinge patterns, it exhibits large bending stiffness and the tension in the bend regions pushes the liquid crystals away, creating always-transparent parts. Further, the ITO layer is only 15 μm thin and sensitive to bending for radii smaller than 4 cm with angles $> 90^\circ$.

To overcome this problem, we tested different hinge patterns, yielding different corner radii and bending stiffness, see Figure 5.9. A sharp edge is created with a single pattern consisting of 4 cuts (Figure 5.9, top left), which allows for arbitrary bend angles. Importantly, this pattern does not break the electrical connection in the ITO, since it relieves stress from the material. This allows creating 3D objects that retain their ability to change transparency. Other hinge patterns are used to create objects with different corner radii. We use patterns yielding low bending stiffness (Figure 5.9, *right*) for creating nested objects, which oftentimes require 180° bends with low bending stiffness.

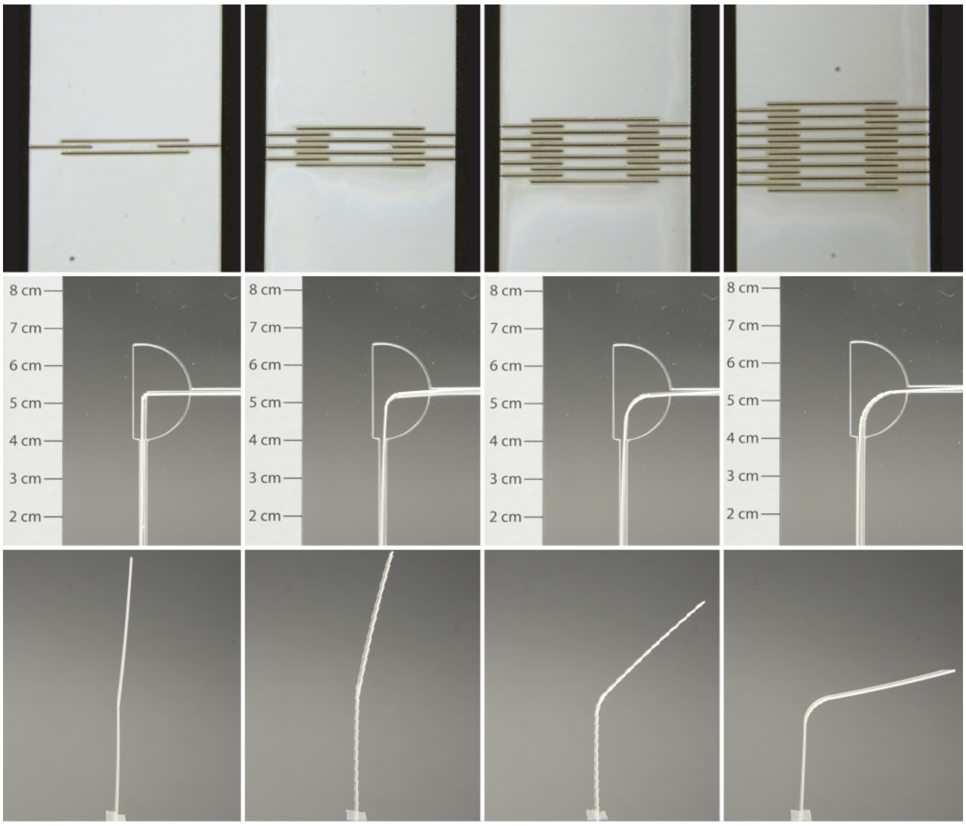


**Figure 5.9:** Corner radius (center) and bending stiffness (bottom) depend on the crease pattern (top).

Beside bend angle and radius, also the number of electrical routes has to be taken into account when designing hinges. Standard hinge patterns, as discussed above, only allow for a maximum of 2 routes. By further dividing hinges in multiple parts, more routes can be added, as illustrated in Figure 5.10.

**Connector** We add a ”connector” to the switchable diffuser (see Figure 5.8). The connecter is manually positioned by users in our custom software. The connector fits a standard FFC press-fit connector, thus no soldering is needed. Since

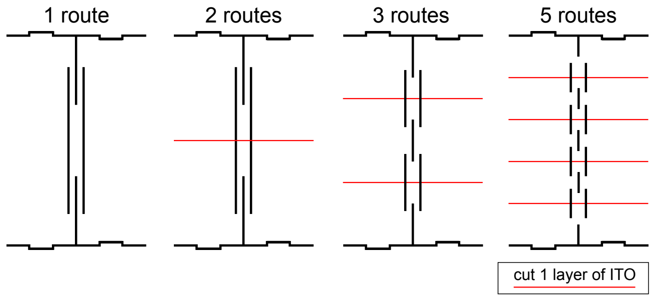


**Figure 5.10:** Crease patterns are adapted to number of routes. Each cut separates two individually controllable areas (or *channels*).

the FFC connector conducts only from one side, we wrap conductive copper tape around the laser cut pins to control the individual parts of the object. This also increases the durability of the object's laser cut connector.

**Routing** Routing includes both disconnecting specific faces from the overall object and adding channels, which are disconnected areas ranging from those faces to the connector. This is done by removing one layer of ITO, as described earlier. From a practical side, the channels which arise from cutting can be 1 to 2 mm thin. Routing is performed for creating segmented objects as well as for creating sprites.

### 5.3.3 Software

Our custom software assists users in converting input crease patterns to laser cut plans (see Figure 5.11). It automatically creates the snap outline and hinge patterns which are adapted to the routing. The software is developed in C++ with openFrameworks[2] for GUI and boost[3] for data structures and algorithms (e. g. boolean operations, polygon orientation correction).

#### Input

Our software takes a crease pattern as input (svg file). We opted for starting our processing pipeline after modeling the crease pattern, since there are specialized

[2] openFrameworks v0.9.0: http://openframeworks.cc/
[3] Boost C++ Libraries: http://www.boost.org/

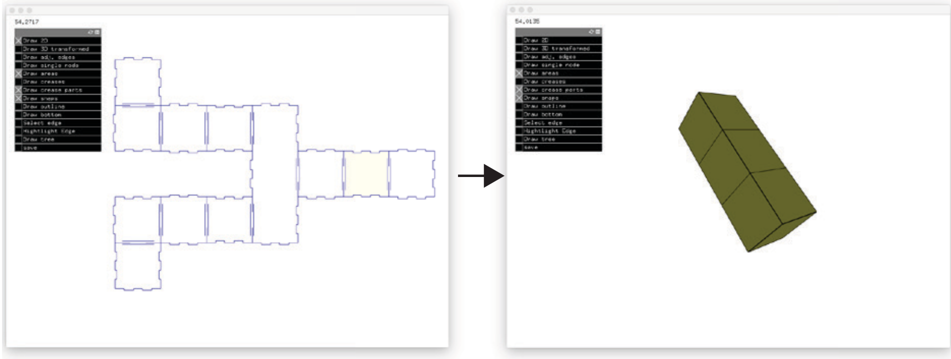

**Figure 5.11:** Our software converts a crease plan to a laser cut plan (*left*). It also features an automatic 3D preview of objects (*right*).

tools for creating origami patterns that feature semi-automated unfolding of 3D shapes (e. g. [135, 169]), as well as tools such as Adobe Illustrator with origami plugins (e. g. Boxshot Origami[4]) for preview that offer a rich set of controls. Our presented objects were modeled with Adobe Illustrator and Boxshot Origami.

The input contains an outline as well as crease lines, annotated as dashed stroke patterns (see Figure 5.6, left). By default, all crease lines represent bend angles of $90°$ in our 3D preview. Custom angles for individual crease lines can be specified by changing the name of the object's layer in the svg file. The input includes an outline, which is a 2D polygon $S = (s_0, s_1, \ldots, s_{n-1})$, $s_i \in \mathbb{R}^2$. Further, it contains a list of creases $C = (c_0, c_1, \ldots, c_{m-1})$, $c_j \in \mathbb{R}^2 \times \mathbb{R}^2$, with each crease consisting of two points $\{c_{j,0}, c_{j,1}\}$ located on the outline $c_{j,k} \in S$ (see Figure 5.6, *left*).

### Preprocessing

For creating the snap outline and the 3D preview, we first divide the outline into multiple areas (i. e. the faces of an object which are polygons enclosed by creases and the outline) based on the crease lines. These areas form a tree structure which we use for creating the 3D preview (hierarchical 3D transformation). Furthermore, this provides us with information which edges overlap when the object is folded. We use the available geometric information to determine areas.

[4] Boxshot Origami plugin: http://boxshot.com/origami/

For finding areas, we first visit all vertices $s_i$ and sort all outgoing edges (including creases $c_j$) which contain $s_i$ based on their outgoing angle. We then iterate through all crease endpoints $c_{j,k}$ and follow the outgoing edges (always taking the path of the smallest angle with respect to the incoming edge) until we reach the start point $c_{j,k}$, i. e. the path formed a cycle. This method yields all areas (i. e. the edges of connected components) for the outline $S$ and its creases $C$. We use this information for generating a snap outline that alternates correctly between male and female connectors.

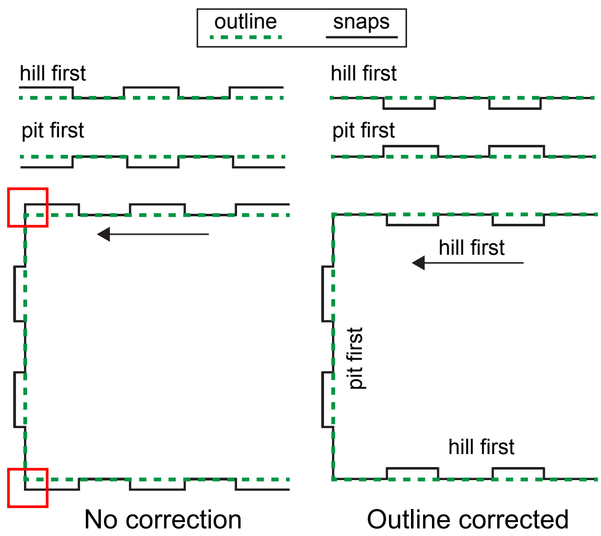


**Figure 5.12:** Without correction (*left*), the outline is shifted on corner points (marked with red squares), resulting in decreased fit when building objects. Outline correction (*right*) resolves this issue.

### Snap outline generation

For generating the snap outline, we successively visit all edges $e_i$ from $S$ where $e_i = \{s_i, s_{i+1}\}$. Each edge is divided into an uneven number of sub-edges (3 to 7, depending on the length of the crease), which gives us a list of sub-edges of $e_i = (e_{i,0}, e_{i,1}, \dots, e_{i,a-1})$. Each sub-edge of $e_i$ is assigned to be either a hill or a pit, in an alternating manner. For hills, the sub-edge is shifted along the normal for a distance $d$. Pits are shifted in the opposite direction with the same distance. We chose $d$ to be 1.2 mm, which approximately reflects the thickness of the switchable diffuser and allows for a good fit when snapping overlapping edges.

Next, we generate the overall snap outline with the aim of retaining corner positions to ensure good fit when folding an object. Each edge $e_i$ is assigned to be hill-first or pit-first in an alternating manner, see Figure 5.12. A naïve approach would be to perform simple shifting of pits and hills. This, however, would shift the overall outline $S$, specifically on corner points (Figure 5.12, left), which negatively influences the folded object's fit. Thus, we change the snap generation to be *outline corrected*. For hill-first edges, only the sub-edges with pits are offset, whereas hills remain on the outline. Conversely, for pit-first edges, only hills are offset from the outline.

### Hinge generation

The hinge patterns (illustrated in Figure 5.9, top) are created at the position of the crease lines. Initially, a standard hinge pattern (Figure 5.9, top left) with four cuts is created. Users then specify the position of the connector and indicate which areas they want to control by grouping faces. This allows us to automatically split hinge patterns according to the number of routes going through each hinge (see Figure 5.10).

### Routing

We opted for a manual routing process, which basically consists of drawing paths on the crease pattern. This allows users to influence the optical appearance of routes (e. g. to be inconspicuous) and to create sprites on individual faces, based on the desired design. In the future, we would like to include automatic routing, e. g. as in Foldio [135].

## 5.4 Applications

We built four prototypes to explore the potential of interfaces that can change their appearance through controlled transparency. All of them were built from a single sheet of switchable diffuser using our described technology. The prototypes serve as demonstrators for three different applications, i. e. as ambient activity indicators, as a playful moving avatar, and as a lamp shade with dynamic appearance.

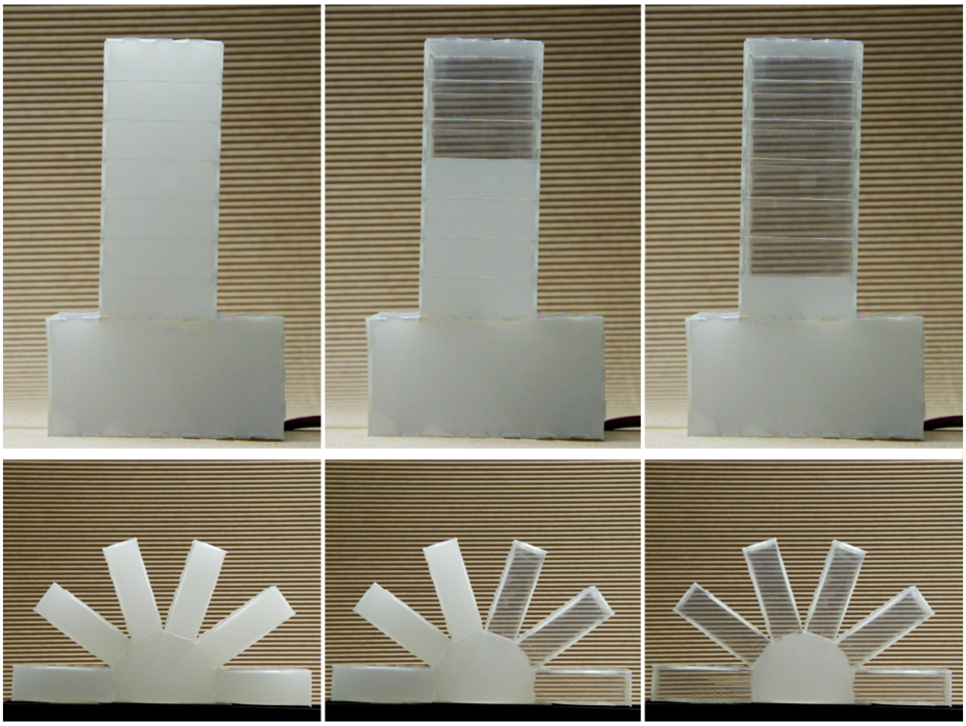

**Figure 5.13:** Our prototypes of progress and activity indicators.

**Activity indicators**

We created two different prototypes of ambient activity indicators, shown in Figure 5.13. Firstly, we built a vertical progress bar that is used for indicating activity (e. g. download progress). We also explored using it as an indicator of available hard disk space on peripherals such as USB drives. The amount of available disk space is indicated by the height of the indicator, i. e. little space used results in a lower height.

Secondly, we constructed a flower-shaped activity indicator. The individual leaves are toggled for indicating notifications. Furthermore, the appearance of the flower can be changed for hedonic purposes. When the indicators are not needed, they can be turned fully transparent, so that they effectively blend into the environment.

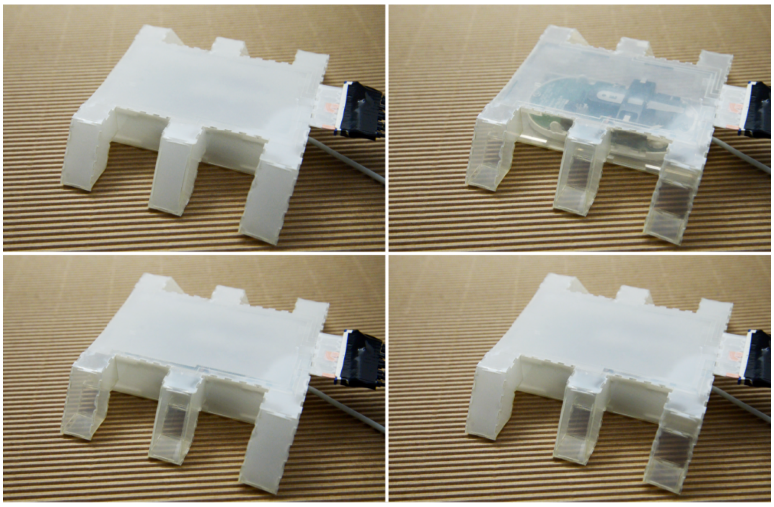

**Figure 5.14:** A bug-like avatar with transparency-controlled legs. Transparency of legs is toggled depending on velocity to create the illusion of motion.

### Moving avatar

We created a playful bug-like avatar (see Figure 5.14). Its body is a cuboid with six legs, each of which can be controlled individually. We included a regular computer mouse in the inside of the avatar for tracking displacement. Groups of legs are turned opaque in sequence. By toggling the transparency of the legs in sync with tracked motion, the illusion of the avatar moving on the ground arises. This demonstrates that, although the avatar does not have any actuated or moving parts, its physical appearance can be extended by perceived motion.

### Appearance-changing lamp

We created a lamp that can change its appearance. The lamp is a nested object composed of a cone inside a dodecahedron inside a cube, shown in Figure 5.15. All these layers can be individually switched or continuously controlled. This nesting of objects, as described earlier, is achieved by folding a single sheet of switchable diffuser. While it serves mostly hedonic purposes in its current state, it is easily imaginable to encode information in the different states. By rhythmically pulsing one of the layers, thereby effectively altering the lamp's perceived volume, users could be informed of notifications or activities happening in their environment.

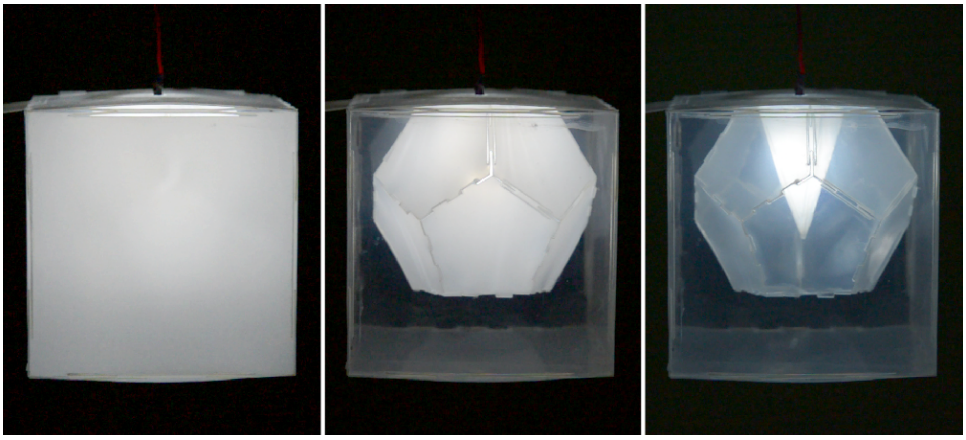

**Figure 5.15:** The appearance of the lamp can be switched between cube, dodecahedron and cone.

## 5.5 Discussion

Changing the appearance of physical interfaces through controlled transparency has a number of benefits when compared to mechanically-actuated interfaces. It is very fast and we do not believe that rapid changes would startle users. Since no mechanical parts are involved, transparency-controlled interfaces are less prone to wear and tear (life time specified as >10 years by manufacturer and we have not observed any wear out of hinges through repeated bending) or other mechanical challenges. They perform changes silently, and use very little energy. Our proposed interfaces allow performing changes that are typically difficult to achieve with shape-changing interfaces such as creating apparent holes (cf. [148]) or nesting. However, other challenges and limitations arise.

### Tangible qualities

Foremost, since change is solely optical, no dynamic tactile qualities are present. This is one of the main benefits of shape-changing interfaces, and we sacrifice it for the aforementioned benefits. However, as we have shown in our example of equipping a shape-changing device with transparency-controlled parts (see Figure 5.5), the technologies are complementary rather than mutually exclusive. We believe our work provides researchers in the field of shape-changing interfaces with a new tool that can be beneficial in many situations, as showcased in our applications. By including materials with shape-changing properties (e. g. shape-memory alloys or hydrogels), we plan to combine the benefits of both worlds in the future. As an example, hydrogels as used in GelTouch [124] can change their viscoelasticity and simultaneously change from transparent to opaque. Switching, however, takes far longer than changing transparency with switchable diffuser. Furthermore, hydrogel needs to be sealed in compartments, increasing its (already high) production complexity.

**Base material & transparency**

We use PDLC switchable diffuser because of its fast switching speeds and good visual clarity in transparent state. Electrochromic materials, as a potential alternative, require less voltage but typically exhibit longer switching times. Another alternative are LC shutters and transparent LCDs, which also require less voltage for state switching (3-5 VDC). However, since they do not allow for laser cutting or folding, they are less suited for creating 3D objects. Switchable diffuser is well suited for objects with dynamic transparency.

PDLC switchable diffuser, however, is prone to reduced transparency for oblique viewing angles. This could be resolved by using a different type of diffuser such as suspended particle devices, which use particles rather than liquid crystals, cf. [158, p. 14-26]. Nesting objects had a less deteriorating effect on perceived transparency, at least for the objects we tested (e. g. three layered cubes, resulting in 6 layers overall). Transparency decreased from 90% (1 layer) to approximately 50% (6 layers). We believe that future generations of switchable diffuser will have increased transparency (close to 100%), especially since the prime use case for this material are dynamic window blinds. Using switchable diffuser also allows us to not rely on external wiring for toggling of individual parts. By incorporating the separation of individual controllable parts directly in the manufacturing process of the switchable diffuser, we believe we will also be able to remove current artifacts originating from laser cutting.

### Software

Our software automates generating the snap outline and hinge pattern generation. These are the most tedious aspects when creating transparency-controlled 3D objects from a crease pattern. While routing could be performed automatically (e. g. as in Foldio [135]), our manual process allows users to take visual clarity into account (e. g. not routing through the middle of a face) and allows for creating sprites. Routing is always a tradeoff between the number of possible individually controllable parts and visual clarity. The number of routes increases linearly with the number of individually addressable faces. Therefore, having a large number of faces on small objects potentially deteriorates visual clarity. Adding a higher degree of automation to our software, including different modes for manual and automatic routing, will be subject of future work.

### Why transparency?

Applications like the progress indicator could be made with non-traditional display technologies such as PrintScreen [134], too. By changing the color for individual areas, progress indication could also be achieved. This would, however, always result in the perception of two or more distinct areas on the same object, depending on the current color configuration. We argue that unwanted parts of the interfaces can be hidden by using transparency-controlled objects. We envision that users always only see the parts of the object that are needed to resemble a specific appearance. Hidden parts blend into the environment. This strengthens the illusion of a specific appearance and avoids that users perceive multiple objects when there should really only be one.

Transparency-controlled physical objects share some capabilities with volumetric or stereographic displays. In contrast to these displays, however, transparency-controlled physical interfaces feature digital enrichment on an *object level*. This means that the optically dynamic elements are tightly integrated into objects, essentially forming their outer hull. Aforementioned displays typically are planar (stereoscopic displays), or spherical or cubic shaped (volumetric displays) and are used for general-purpose rendering of contents. While they have benefits in terms of display capabilities, they cannot be tightly integrated into other objects.

We think that transparency-controlled physical interfaces are complementary to current work on dynamic physical interfaces. It offers benefits in speed, power consumption and ability to render certain features that can be a valuable addition to physically dynamic interfaces. Furthermore, transparency-controlled physical interfaces allow rapid toggling as well as continuously changing transparency of individual parts. This additional temporal dimension potentially offers a range of novel behaviors, such as smooth transitioning between nested objects. For mechanically actuated physical interfaces, change is always continuous, whereas our proposed interfaces allow switching between shapes in a discrete manner. We believe that these temporal aspects and their in-depth investigation would result in interesting future applications of transparency-controlled physical interfaces. Lastly, our objects do not incorporate sensing capabilities, therefore rely on technologies such as camera-based interaction or manual activation for input. By including sensing capabilities such as capacitive touch (e. g. through additional layers of ITO on the outside of the diffuser), we think a transparency-controlled physical interface can serve as both, input and output technology.

# 6

# Evaluation of dynamic objects in the context of transparent displays

## 6.1 Introduction

Dynamic objects offer potential advantages such as the ability to change their appearance without instrumenting users. It is still unclear, however, what the benefits in contrast to traditional interfaces and devices are. We performed an experiment towards the usability of dynamic objects in the context of transparent displays. The findings inform the design of transparency-controlled displays and how users interact with them. Various researchers proposed transparent displays as a medium for collaboration and to increase consequential and situation awareness (e. g. [64, 104, 107]). These displays offer potential benefits in situations where users want to simultaneously observe screen content and the environment behind the display, or when attention is frequently switched between the two.


**Figure 6.1:** We conducted an experiment comparing a transparent display (*left*) with typical display configurations such as a conventional display at an angle of 30° (*center*), and a horizontal display (*right*) in a dual-task scenario (square-click as primary task, background observation as secondary). Results show constant primary task performance across all conditions, but increased background awareness for the transparent (83%) and horizontal (70%) display.

With conventional (opaque) displays (e. g. computer screens), usually positioned in front of a user, important events can remain unnoticed since the displays block the view on the environment. Examples of such events include a person approaching, a colleague beginning to be available for communication, or a change in situation in a command-and-control room. In order to overcome challenges of conventional displays, prior work has used horizontal displays (i. e. vertically tilted conventional displays) such as tabletop displays to allow users to perform individual or collaborative tasks while simultaneously observing the environment. Additionally, in situations where the locations of important events are known, users typically position their display in a way that the environment is not obstructed.

It is yet unclear how transparent displays compare to conventional and horizontal displays when users perform a primary task on their display and simultaneously try to be aware of events happening in the background (i. e. secondary task). Transparent displays have the benefit that users' primary task as well as the background are within the same visual area. In contrast, conventional opaque displays have to be moved aside to unblock the view and see the background. Horizontal displays intrinsically increase the visual angle between primary task and background since they are vertically tilted. However, in contrast to conventional displays, transparent displays no longer shield users from motion or other distractions occurring in the background.

Until now, most research focused on the benefits of transparent displays e. g. for communication and collaboration (e. g. [64, 103]). However, the effects of background awareness and impact of distraction as well as the gain in background awareness have not yet been investigated and quantified. It is hence unclear how display transparency influences users' task performance, both in terms of objective measure (e. g. time and errors) as well as subjective measures.

In our work, we conduct an experiment to investigate if transparent displays are useful for dual-task scenarios, and how they compare to conventional opaque displays that are offset from the background in which events occur, as well as horizontal displays. We are concerned with situations where users perform a regular task on their main display while *simultaneously* observing the environment for specific events, e. g. in command-and-control rooms. Other observation scenarios include air traffic controllers performing regular tasks while reacting to events triggered by their colleagues, or security members in a stadium observing a large crowd while performing regular control tasks on a display. We compare a transparent display with conventional displays at three different locations with respect to the background (directly in front of participants, at 30° and 60° beside participants), and a horizontal display. Participants performed an attention demanding primary task (square-click, cf. [144, 162]) and were instructed to react to specific stimuli in the background (letters). Background stimuli were displayed on a 100" projection 4 meters away from participants, which results in an observation area of 30° in participants' visual field (see Figure 6.1 and 6.6). Besides these target stimuli in the background, participants had to ignore other distractor-stimuli.

Our findings show that participants were able to focus on their primary task across all conditions, keeping their time and error rates constant. However, their ability to identify target stimuli was highly influenced by display configuration. For the conventional display positioned at 30° at the side, participants identified only around 55% of target stimuli, although the position was optimal in terms of visual angle (i. e. the visual field of interest was contiguous). Both the main display and the background were within their central to mid peripheral vision. Still, participants missed nearly half of the target stimuli. Background stimulus detection rates were best for the horizontal display (70%) and the transparent display (83%).

In the following, we present scenarios which we considered when designing our experiment, followed by background information regarding our choice of conditions in terms of display configuration. We argue that the positioning of displays and the resulting obstruction of visual field plays a key role for users' ability in the described dual-task scenario. Subsequently, we present our experiment, with quantitative and qualitative results. We conclude by discussing our results and giving implications and recommendations.

## 6.2 Scenarios

We designed the tasks used in our experiment with the following scenarios in mind.

1) A clerk at a train station performs organizational tasks on the display in front of her (see Figure 6.2). She sits behind an open counter, which allows her to talk to approaching passengers as well as observe the train station for specific events. Such events include people looking for guidance or other events which she can handle or report to colleagues. The display also allows her to see approaching people and talk to them while looking up information without blocking her view.

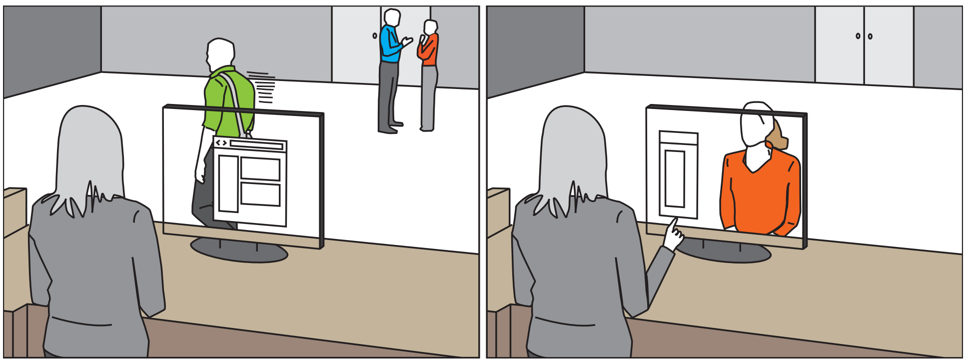

**Figure 6.2:** Scenario with a clerk performing tasks on her transparent display while observing the background (*left*) for important events such as a customer approaching (*right*).

2) A project manager supervises a team of designers and programmers. She performs organizational tasks individually but simultaneously observes the office for events such as approaching clients, or colleagues approaching a large shared white-

board situated in front of the room at a distance of a few meters. The whiteboard is used to keep track of tasks, with post-it notes relating to specific tasks, based on the current iteration of the software the team is developing. Team members can approach the whiteboard and take post-it notes with them to perform tasks. She needs to loosely keep track of who performs which task and help colleagues choosing tasks.

3) A shuttle flight controller works in a large mission control center. Her duties include supervision of other personnel. She monitors her personal display to keep track of incoming data while keeping track of her colleagues and respond to unforeseen events. The large shared display in front of the room also includes statistics and real-time data. The room is busy with other personnel coming and going. Furthermore, she has to answer incoming requests from colleagues approaching her, for which she needs her computer.

In all these scenarios the high-priority task is conducted on a personal display, therefore it is the users' *primary task*. Simultaneously, users observe the background for different types of events. Since there is constant motion in the background (e. g. through visitors, moving colleagues, or interactive content on a large shared display), users have to process information and actively ignore stimuli and unimportant events. Primary tasks like the ones described are typically performed with either vertical or horizontal displays. Clerks and office workers use conventional displays to perform their tasks and position them in a way that they do not block the view on the environment they want to observe. Horizontal displays (e. g. tabletops) are common equipment in command-and-control rooms since they can be approached by multiple people and allow for seeing the environment. Furthermore, laptop computers provide the same benefits since they do not block the view on the environment when positioned on a table. We see them as tilted horizontal displays, also reflected in our experimental conditions. Keeping these scenarios in mind, we designed the primary task to be attention demanding while the background stimuli are rather fast and constantly changing. Additionally, participants have to actively ignore stimuli and cannot simply respond to all events in the background, resembling the scenario of a busy environment.

## 6.3 Background

In this section we review relevant related work on task performance and display configuration. Furthermore, we provide background on the influence of visual angle and the human field of view on display settings and occlusion.

### 6.3.1 Dual-task performance and observability

Dual task scenarios have been investigated in the context of reading (e. g. [133]), pointing with task-relevant stimuli (e. g. [95]), and peripheral displays (e. g. [21, 114]). Probst et al. [144] and Hausen et al. [62], among others, investigated the idea of using separate input devices for peripheral interaction, which involves interactions that take place during or as interrupts of primary tasks. Bartram et al. [9] investigated the influence of motion on detection and distraction for on-screen notifications during a dual-task. Maglio and Campbell [114] investigated peripheral information displays, also in the context of performing dual-tasks. In contrast to our work, both Bartram et al., and Maglio and Campell focused on observability and notifications on a single display and how user reaction is influenced by different cues (e. g. motion, change in shape or color).

Reetz et al. [151] investigated the influence of gesture size on observability, more specifically on users' ability to observe others' actions (i. e. consequential awareness, cf. [56]). Our work is informed by their idea of different user configurations with respect to target stimuli (in their case gesture size, in our case display configuration) and their influence on task performance and consequential awareness. Furthermore, in the field of computer-supported cooperative work (CSCW), research focused on the impact and importance of consequential awareness (e. g. [56, 173]).

Our work contributes by providing insights into the influence of users' ability to observe the background with respect to typical and novel display configurations.

### 6.3.2 Display configuration

Inkpen et al. [74] explored the influence of different display factors on co-located collaboration. They report on a variety of factors influenced by the configuration, e. g. participants noted different ergonomic issues between vertical and horizontal displays. Ichino et al. [72] investigated the influence of display configuration in a museum context and found that tilted displays increased user experience (e. g. attracted attention and increased understanding of content). Forlines et al. [46] showed that both display configuration and group size influence the performance for visual search tasks. They tested single and multiple vertical displays as well as horizontal displays and found, among other things, that the choice of display influences reaction times. Rashid et al. [145] presented a survey of the influence of display factors on attention switching. They discussed user performance when using displays at different levels of depth as well as size.

Besides this work, a large body of work focuses on display size as influencing factors (e. g. [36, 112]). Swaminathan and Saton [168] provided a framing for various display configurations with displays spreading across a the visual field and across multiple levels of depths. They refer to displays which spread seamlessly across the visual field as *desktop-contiguous*, and to displays covering multiple visual areas (i. e. gaps between displays) as *non-contiguous*. In our experiment, we adopt both types, desktop-contiguous and non-contiguous, as baselines for comparison with a transparent display. Tan and Czerwinsky [170] investigate the influence of visual separation and physical discontinuities in a multi-task scenario (primary task and notifications). They focus on dual-display and display + projector setups, distributed on multiple depths. Their findings suggest a minor decrease in task performance when performing tasks on a display while reacting to notifications in the background. We extend their experiment to larger distances between display and projector (i. e. non-contiguous), as well as to horizontal and transparent displays.

In our work, we include the effects of display transparency as an important factor for dual-task scenarios. While having been proposed for a variety of use cases, it is yet unclear if users actually benefit from a transparent display or if potential benefits are outweighed by distraction from the background behind the display.

### 6.3.3 Displays and the human visual field

Traditional desktop computer setups feature one or multiple vertical displays, occluding a certain area of the background. As depicted in Figure 6.3, a single 22" display at a distance of 90 cm occludes approximately 30° of a user's visual field, roughly covering the central and near peripheral part. By increasing the number of displays or the display size, an area covering the mid peripheral visual field becomes occluded, limiting users' ability to observe events in the background.

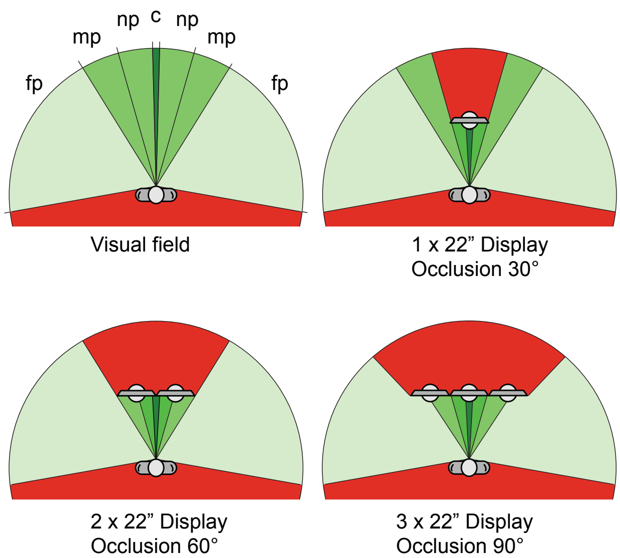


**Figure 6.3:** Occlusion in the visual field depends on display configuration. Red marks areas not visible to users. *Top left* illustrates properties of the human visual field with *c* (central) ranging from 0° to +/-1.5° , *np* (near peripheral) from +/-1.5° to +/-15° , *mp* (mid peripheral) from +/-15° to +/-30° and *fp* (far peripheral) from +/-30° to +/-100° .

When using conventional vertical displays, users have to configure (i. e. position and rotate) them in a way that they do not occlude important parts of the background. Prior work suggests that, when using multiple displays at different depth levels [145], having the displays edge-aligned results in no significant decrease in performance. We believe that users will exhibit the same behavior for observing events not only on a secondary display but also in the environment behind or beside displays. Dependent on the probability of important events in a certain area in the background, users will choose the position of their display accordingly, as depicted in Figure 6.4. Since the area behind the display is occluded (covering approx. 30° of users' visual field), the display has to be positioned beside the important area. In the best case, important events only occur in the area anticipated by users (Figure 6.4, green areas), i. e. right beside the display at a angular distance of within 30° to 60°. Events occurring outside this area potentially remain unnoticed since they lie outside users' mid and far peripheral visual field. Dependent on the screen real estate needed to perform tasks, the area which users can observe simultaneously while performing primary tasks on their displays can be highly limited. As depicted in Figure 6.3, a three-display configuration (covering approx. 90° visual field) leaves approximately 55° of visual field on each side for observation. This area only lies in users' far peripheral visual field, increasing the possibility of events going unnoticed. This problem is increased in situations where task complexity is high which can narrow users' visual field (cf. [97]).

In order to overcome the problem of occlusion with vertical displays, a large body of work in the field of CSCW and situation awareness suggests using horizontal displays (e. g. tabletop displays, [74]). Horizontal displays allow users to position themselves centered in areas where important events might occur. While this is beneficial for situation awareness and collaboration (e. g. face-to-face communication), other challenges arise. In contrast to vertical displays, content displayed on horizontal displays is less visible to others not standing close to the display. This is especially important in situations where others need to observe actions on the display (e. g. command-and-control centers, cf. [74]) . Additionally, horizontal tabletop displays can lead to ergonomic challenges since users constantly have to lower their head [74]. Transparent displays, as proposed in prior work (e. g. [104, 107]), offer a potential solution to these challenges. They allow users to position themselves so that potentially important areas in the background lie in their central visual field. Additionally, they allow users to simultaneously observe on-screen content and the environment behind the display. However, having screen content and the background in the central or near peripheral visual field might distract users.

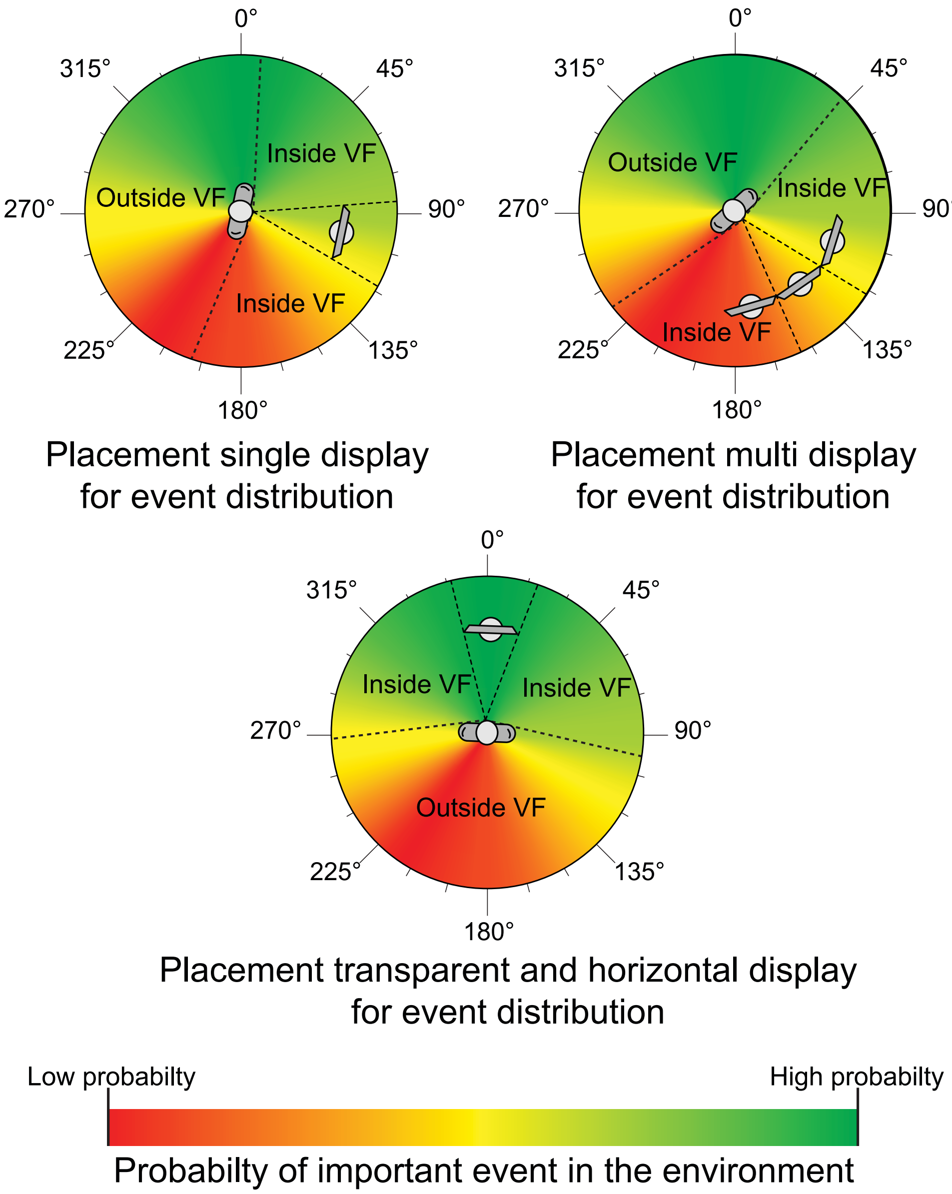


**Figure 6.4:** Display positions dependent on event probability. Visual angle and display size influence users' choice of display placement.

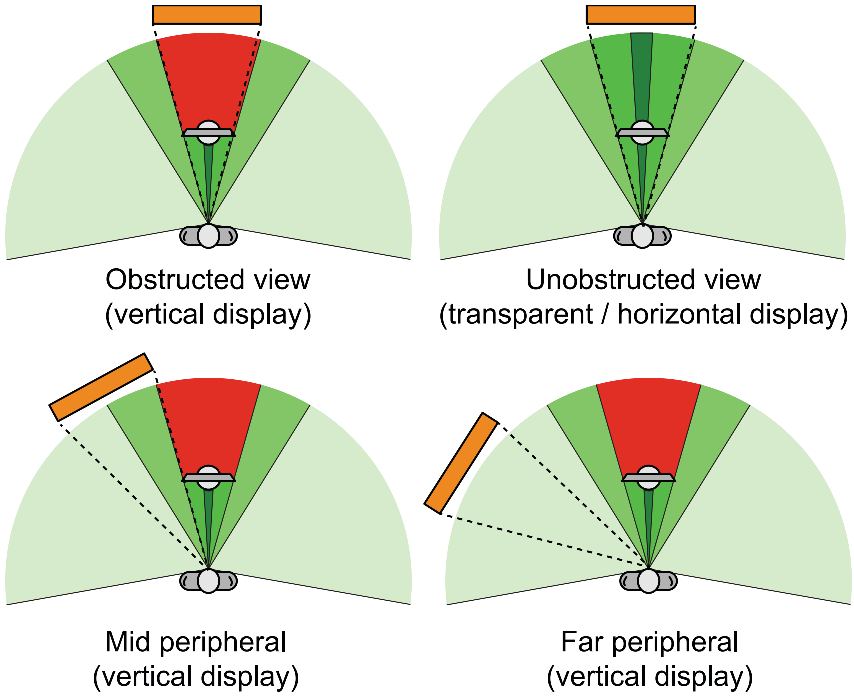


**Figure 6.5:** Display configurations and event locations we tested. Orange marks event locations, red the obstructed visual field.

Therefore, our experiment contained conditions comparing typical display configurations with participants observing important events in different positions with respect to the display. Figure 6.5 illustrates the scenarios we aimed to test, including when the view on the environment behind the display is or is not obstructed and the events occur in an optimal or suboptimal location relative to the display. We aim to inform the design of systems where vertical displays are beneficial (e. g. view on screen content is important), as well as compare transparent displays to horizontal displays. Additionally, we aim to answer the question if transparent displays are beneficial for awareness or if the distraction from the environment behind the display and any resulting decrease in performance outweighs potentially advantages.

## 6.4 Method

We conducted an empirical study in order to explore the influence of display transparency and configuration on task performance and users' ability to observe stimuli in the background. Therefore, we tested a dual-task scenario, with participants primarily focusing on the task performed on a display in front of them while simultaneously observing the background for target stimuli. Participants were performing tasks with 5 different display configurations. Configurations included a conventional display in three different configurations, i. e. positioned and rotated at 0°, 30° and 60°, as depicted in Figure 6.6. Additionally, we used a transparent display and a horizontal display. The transparent display and the horizontal display were positioned between participants and the background area they had to observe.

### 6.4.1 Participants

We recruited 20 paid participants (5 female) from a local university, aged between 20 and 33 years ($MDN = 26$ years). They were typical display workers ($M = 7.7\,h$ per day, $SD = 1.9\,h$ ). All had normal or corrected to normal vision (based on self-reports) and had no prior experience with transparent displays.

### 6.4.2 Apparatus

The study was conducted in a calm experimental room with controlled lighting. We used a rear-screen projection display positioned in front of participants as both, opaque and transparent, display. As a projection surface, we used a 22" sheet of PDLC switchable diffuser (Kewei Films Non-Adhesive Smart Glass), mounted on a laser cut acrylic frame. The switchable diffuser can be toggled between transparent and opaque (i. e. diffusing incoming visible light) by applying voltage to it (110 VAC). In transparent (activated) state, the switchable diffuser offers a visible light transmission of approximately 82% (5% when deactivated, according to specification). It serves as a projection surface in transparent and opaque state, and maintains resolution and comparable brightness and contrast in both states. For projecting onto the switchable diffuser we used a Benq W1060 DLP projector (resolution 1920 × 1080 pixels, 2000 ANSI lumens). The projector was positioned behind the display and carefully adjusted for each participant to avoid glaring or blending them. Positions of the display were marked to ensure consistency across participants.

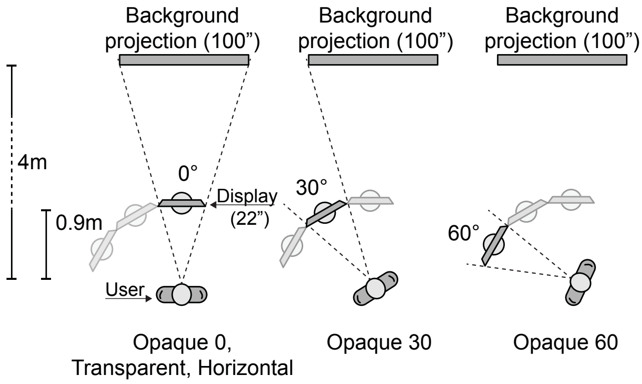


**Figure 6.6:** Experimental setup and display configurations used in the experiment. The opaque display was oriented at all three angles, whereas the transparent and the horizontal display were only positioned directly in front of participants.

We decided to apply this combination of switchable diffuser and projector, as it has a much higher level of transparency than commercially available transparent LCDs (e. g. about 15 - 20% for the Samsung LTI330MT02). We believe our experiment would be severely biased with a display technology (i. e. transparent LCDs) that is inferior to current non-transparent displays, especially since we consider the limited features of current commercial transparent displays as a limitation that will be overcome in the near future. Figure 6.7 shows a side-by-side comparison of the display in opaque and transparent state.

For displaying background stimuli behind the display, we used a 100" (2.54 m) wall projection (resolution 1280 × 800 pixels), at a constant distance of 4 meters to participants, illustrated in Figure 6.6.


**Figure 6.7:** The projected display used in the experiment for performing the primary task, in opaque (*left*) and transparent state (*right*, stimuli of background task, letter "M", clearly visible). Glaring effects in transparent state (light blue bottom) are photography artifacts and not visible to participants.

### 6.4.3 Design

We used a within-subjects repeated measures design with *display configuration* as independent variable. *Display configuration* consisted of 5 levels, *Opaque 0*, *Opaque 30*, *Opaque 60*, *Horizontal* and *Transparent*, illustrated in Figure 6.6, counterbalanced using a Latin square. For *Opaque 0*, *Transparent* and *Horizontal*, the display was positioned in front of the participants, i. e. between participants and the projection. Participants were allowed to lean over to see the background display for *Opaque 0*, however they were instructed to not change the position of the chair or to stand up. *Opaque 30* represents a contiguous setup distributed on multiple depths, whereas *Opaque 60* represents a non-contiguous setup (cf. [168, 170]). As primary task, participants performed a square-click task (cf. [144, 162]). As secondary task, we displayed random stimuli (i. e. letters) on the background projection and participants had to respond to specific ones.

#### Tasks

We adopted an attention-demanding primary task, the *square-click task*, usually applied in ambient information and peripheral interaction experiments (e. g. [144, 162]). In this game-style task, a square (150 × 150 pixels, approx. 4 × 4 cm) appears at random locations on the display in front of participants. Each square includes a random single-digit label and can be resolved by clicking the square and inputting the correct number before the next square appears. If no response is given, the trial is marked as error. New squares appeared at a random interval of 2 or 2.5 seconds. We controlled randomization for each condition to result in 120 trials, therefore each participant performed 600 trials overall (5 condition $\times$ 120 trials) during the experiment. We chose this task since we believe it provides a good balance between light to moderate cognitive load and visual load, while requiring participants' full attention.

As secondary task, participants had to observe the background for a specific target stimulus, the letter "K" in our case, appearing at random locations in the background. As soon as participants identified the target stimulus, they had to indicate this by pressing the space bar within a timeframe of 2 seconds. Distractor letters were displayed additionally to the target stimulus, with always one stimulus visible at a time. New stimuli were presented every 1 to 2 seconds (randomized). A total of 45 target stimuli were presented to users, with a target to distractor ratio of 1:7.

This task is an adoption of the n-back task, typically used in psychological and HCI experiments (cf. [28, 52]). In the n-back task, stimuli (typically letters) are displayed centered at a screen. For each presented stimulus, participants must indicate if the current stimulus matches a stimulus displayed $n$ trials (e. g. 1, 2, 3) ago, while ignoring other distractor stimuli. By varying the number $n$, the working memory load can be adjusted, with workload increasing as $n$ increases. Our secondary task can be seen as a 0-back task with random stimuli location, its mental load therefore is relatively low. We chose this task since it requires very little learning and resembles a scenario in which participants know immediately to which events they have to respond and which to ignore. Furthermore, we opted for a 0-back task with low mental demand since high mental demand (e. g. through higher $n$) can narrow users' visual field in which they notice events (cf. [97]).

### Hypotheses

We performed the experiment with respect to the following hypotheses. We expected that the amount of effort needed for simultaneously observing the background for target stimuli would impact primary task performance. Additionally, we hypothesized that the visual distraction from the background would influence especially task performance on the transparent display negatively. We expected *Opaque 30* and *Horizontal* to be beneficial for primary task performance and the observational secondary task, whereas distraction to influence primary task performance for *Transparent* negatively. Due to their rotational configuration, we expected primary task performance to be lower for *Opaque 0* and *Opaque 60*.

Therefore, we hypothesized from best to worst (i. e. *Opaque 30, Horizontal, Transparent, Opaque 0, Opaque 60*), which resulted in the following hypotheses:

*H1*. *Opaque 30* will result in higher primary task performance than *Opaque 0* and *Opaque 60*.
*H2*. *Transparent* and *Horizontal* will result in higher primary task performance than *Opaque 0* and *Opaque 60*.
*H3*. *Opaque 30* and *Horizontal* will result in higher primary task performance than *Transparent*.

Additionally, we formed two hypotheses regarding participants' ability to observe stimuli in the background (i. e. secondary task performance). We expected the rotational configuration of *Opaque 0* and *Opaque 60* to influence secondary task performance negatively. We did not expect any differences between *Opaque 30, Horizontal* and *Transparent*, since primary and secondary task stimuli appear within participants' near to mid peripheral visual field.

*H4*. *Opaque 30* will result in higher secondary task performance than *Opaque 0* and *Opaque 60*.
*H5*. *Transparent* and *Horizontal* will result in higher secondary task performance than *Opaque 0* and *Opaque 60*.

### 6.4.4 Procedure

Participants were briefly introduced to the setup and the experiment and completed a demographic questionnaire. Participants first performed a training session for the primary task, always with *Opaque 0*. They were asked to *complete the primary task as fast as possible without making any errors*. Afterwards, participants completed a short training session for the secondary task.

Subsequently, participants performed the task in all conditions, counterbalanced using a Latin square. Before each condition, participants were seated in front of the current display configuration and instructed to start the condition at will by clicking a button. Each condition took approximately 6 minutes. After each condition, participants completed a questionnaire based on Tyrrell's six scores of mental and visual fatigue ([177], cf. [41]), with answers on a seven point Likert scale of strongly disagree (1) to strongly agree (7). Additionally, they answered questions regarding overall distraction and potential usage of the system. Furthermore, they were asked for general comments on the display configuration they just used. After the experiment (duration approximately 60 minutes), participants were debriefed and compensated.

### 6.4.5 Data collection

We collected primary and secondary task data for times and error using custom logging software. All input events were logged, including time to resolve squares in the primary task, and errors. For the secondary task, correct responses including time information (starting from appearance of the target stimuli) as well as errors were recorded. Errors were counted when participants missed a target stimulus or responded to a distractor.

## 6.5 Results

In this section, we report on our quantitative results from primary and secondary task as well as qualitative results from collected post-condition questionnaires.

### 6.5.1 Primary task results

Trial completion time was calculated from appearance of a square to participants correctly resolving it by clicking on it and inputting the correct number. Error rates were measured as the number of unresolved squares. Trial completion and errors were analyzed using two individual $5\times$ (*display configuration*) one-way ANOVAs ($\alpha = .05$) on the dependent variables trial completion time (averaged trial completion times, only correct trials) and error. Results did not show significant differences between conditions, neither for trial completion time nor error. Participants resolved each square at an average of 1.61 s ($SE = 0.01\,s$), without differences between conditions ($F(4, 95) = 0.168$, $p = .954$), as depicted in Figure 6.8. On average, participants made 8.19 errors ($SE = 0.55$) per condition (of 120 trials), across all conditions ($F(4, 95) = 0.063$, $p = .993$).

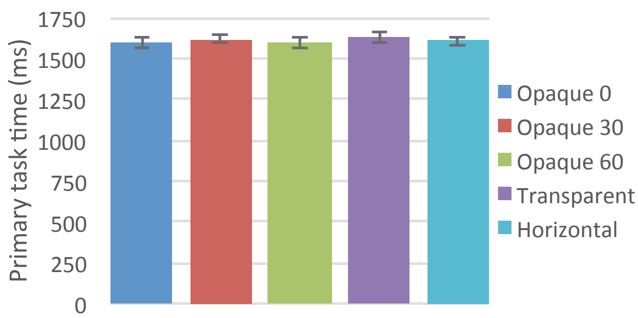


**Figure 6.8:** Primary task trial completion times in milliseconds across all conditions. Error bars indicate standard error.

**Discussion**

Interestingly, participants' primary task performance was not influenced by the display configuration. First, this suggests that participants followed our instructions to focus on the primary task and handle background stimuli as secondary task. Secondly, and more surprisingly, participants were *not* influenced by the background task for conditions with the transparent display. As reflected in participants qualitative ratings, they did not feel distracted. Our results show an even distribution of trial completion time across all conditions. Statistical analysis indicates that the minor differences in trial completion time between conditions are of random nature, which contradicts our hypotheses formed with respect to primary task performance (*H1* - *H3*).

### 6.5.2 Secondary task results

We analyzed secondary task performance with the dependent variables being *number of detected background* stimuli and *detection delay*. False-positive errors occurred very rarely (48 in total for all conditions and participants) and were therefore not analyzed. Both dependent variables were analyzed using individual one-way ANOVAs ($a = .05$). The stimulus detection delays were not statistically different across conditions ($F(4, 91) = 0.19$, $p = .943$) with a mean of 1.08s ($SE = 0.02s$). Means across conditions are depicted in Figure 6.9.

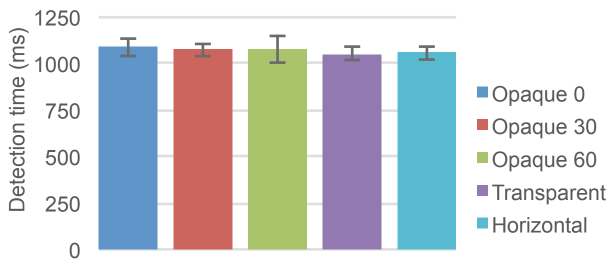


**Figure 6.9:** Secondary task stimuli detection delays in milliseconds across all conditions. Error bars indicate standard error.

An analysis of the number of detected background stimuli showed a significant main effect ($F(4, 95) = 38.375$, $p < .01$). Mean values and standard errors are illustrated in Figure 6.10. A Tukey post-hoc test revealed that the detection rate across several conditions was significantly different. Participants detected significantly less stimuli for *Opaque 0* ($M = 24.6\%$, $SE = 3.7\%$) and *Opaque 60* ($M = 30.1\%$, $SE = 5.4\%$) than for any other condition (all $p < .001$). For *Opaque 30* ($M = 55.33\%$, $SE = 4.4\%$), the detection rate was lower than for *Transparent* ($M = 81.8\%$, $SE = 2.4\%$, $p < .001$) and *Horizontal* ($M = 70.4\%$, $SE = 3.6\%$, $p = .067$, $p < .001$). The difference of 11.4% between *Transparent* and *Horizontal* was not statistically significant ($p = .274$).

### Discussion

In contrast to primary task performance, results were highly different across conditions for secondary task performance, more specifically detection rate. Participants had problems detecting background stimuli especially during *Opaque 0* and *Opaque 60 conditions*. Therefore, hypothesis *H4* (*Opaque 30* better than *Opaque 0* and *Opaque 60*) is supported.

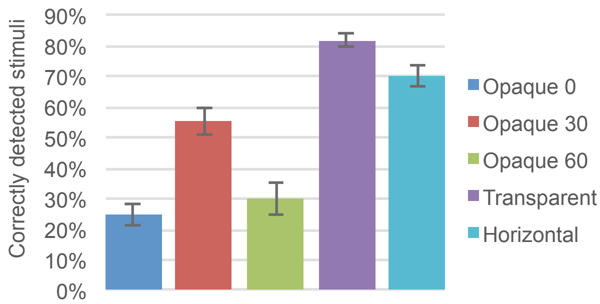


**Figure 6.10:** Percentage of detected background stimuli during secondary task. Error bars indicate standard error.

During the experiment, we saw participants having different strategies for *Opaque 0*, with the most common strategy being to lean over to see both screens. However, even for those participants, background stimulus detection rates were mostly around 35%. Only one participant (P2) reached 50% with *Opaque 0*, but also performing exceptionally well in the other conditions (e. g. *Transparent* and *Horizontal* with 93%). With *Opaque 60*, the background was outside participants' mid peripheral visual field, making it hard to focus on both tasks simultaneously.

Hypothesis *H5* (*Transparent* and *Horizontal* better than *Opaque 0* and *Opaque 60*) is also supported. Additionally, we saw that participants detected nearly 30 percentage points more background stimuli for *Transparent* than for *Opaque 30*. This was surprising, since we expected that the ability to have both displays in the mid to near peripheral visual field for *Opaque 30* would give participants the ability to achieve high background stimulus detection rates.

Lastly, we saw that the difference in detection rate between *Transparent* and *Horizontal* was not significant, although the average detection rate differed by 11.4% in favor of *Transparent*. For *Horizontal* and *Opaque 30*, the background is visually offset to the screen. It seems like the horizontal rotation of *Horizontal* had less impact on secondary task performance than the vertical rotation of *Opaque 30*.

### 6.5.3 Qualitative results

We analyzed data gathered from post-condition questionnaires using a series of Friedman tests. Participants answered questions on a 7-point Likert scale, ranging from strongly disagree (1) to strongly agree (7). Results showed significant differences for answers to two questions, which were "My back and/or neck hurts from sitting in this position while performing this task." ($\chi^2(4) = 17.091, p < .01$) and "I would use the system I just tested." ($\chi^2(4) = 38.906, p < .001$). Answers for the other questions did not differ significantly across conditions.

For gaining further insights, we conducted a series of Wilcoxon Signed-Rank tests (Bonferroni adjusted $\alpha = .005$). For the question if participants felt that their back hurt, results showed significant differences between *Transparent* ($M = 1.70$, $SD = 1.30$) and *Opaque 0* ($M = 3.80$, $SD = 2.21$, $p < .005$), and *Transparent* and *Opaque 30* ($M = 2.80$, $SD = 1.88$, $p < .005$). No other significant differences to *Horizontal* ($M = 2.35$, $SD = 1.57$) or *Opaque 60* ($M = 2.70$, $SD = 1.53$) were present.

Participants' ratings on the usage of the system revealed significant differences between conditions (best to worst: *Transparent*, *Horizontal*, *Opaque 30* and *Opaque 60*, *Opaque 0*). In more detail, there were differences between *Transparent* ($M = 4.05$, $SD = 1.36$) and *Opaque 0* ($M = 1.60$, $SD = 1.57$), *Transparent* and *Opaque 30* ($M = 2.35$, $SD = 1.31$), and *Horizontal* ($M = 3.45$, $SD = 1.90$) and *Opaque 30* (all $p < .005$). Difference between *Transparent* and *Opaque 60* ($M = 2.35$, $SD = 1.63$) did not reach statistical significance ($p = 0.0053$).

#### Discussion

In general, participants' scores on Tyrrell's questions on physical and mental fatigue were low (overall $M = 3.02$, $SD = 1.89$), showing that they could perform all tasks without too much effort. Participants' preference for some of the conditions is reflected in the comments we received as part of the questionnaire, such as P6's written comments to *Opaque 30* "This system is not as good as the transparent display, but better than all other configurations I tested.", or "Because I experienced the see-through display already I don't want to use the system I just tested." or P8 when starting the *Transparent* condition "This is awesome, I want one of these.". While we believe that the novelty of the transparent displays positively influenced the ratings, we saw participants acknowledging the usefulness of the transparent displays.

The horizontal condition was also perceived positively, though participants complained about the constant head movement (up and down). This is reflected in the comments we received such as "It seemed to generate better results (higher hit rate), but at the same time it felt more stressful (for the eyes and brain)." (P2), "Most comfortable position, can give equal attention to both the tasks." (P5, with a background stimulus detection rate of 93% for *Transparent* and 68% for *Horizontal*), or "Always looking down and up again is pretty annoying." (P11).

## 6.6 Discussion

In our experiment we found that the transparent display and the horizontal display were superior in the observational secondary task compared to a conventional display, independent of its angular position. This is especially surprising for conditions with the conventional display tilted only up to the degree where display and background are equally visible (i. e. edge aligned, cf. [145]). We expected that when both the primary task display and the background are within the near and mid peripheral visual field, participants would perform equally well as with the transparent or horizontal displays. This makes us believe that users ability to perform a typical display worker task while simultaneously observing a busy environment *beside the display* results in poor observational performance.

Although our participants were able to anticipate the occurrence of target events, they did not identify more than 55% of these events with *Opaque 30*. We believe this is due to the angular difference in visual field between primary and secondary task. Although the visual separation is as small as possible with a conventional (opaque) display by placing it right beside the observation area with *Opaque 30*, results show that even this small separation decreases secondary task performance significantly, especially when compared to *Transparent*.

Comparing *Opaque 30* and *Horizontal*, we believe the decreased performance for *Opaque 30* could be a result of differences in visual area. Both displays were 16:9 (i. e. landscape orientation), thus the visual area that needs to be observed is bigger for *Opaque 30* + Projection compared to *Horizontal* + Projection. Therefore, it might have been easier for participants to see both displays simultaneously for Horizontal.

While we acknowledge that the events in our secondary task appeared in very frequent order, we believe that these results give insights for systems where users have to perform a regular display task and observe the environment simultaneously.

We did not see a decrease in primary task performance as found by Tan and Czerwinsky [170]. This is most likely due to the difference in tasks, since also the detrimental effects found in their work were rather small. For secondary task performance, however, effects were rather large for our experiment. This suggests that the decrease in secondary task performance is guided largely by visual separation, but likely also task specific. In contrast to our work, Tan and Czerwinsky used a notification-based secondary task, which allows for easy detection of stimuli because of their sudden appearance and the lack of distractors.

### Implications and recommendations

From our quantitative and qualitative results, we draw the following implications and recommendations.

- **Proximity to visual field is important:** Users' ability to observe the environment for events is highly influenced by offset in visual angle. If the location of an event is known, the proximity in terms of visual angle must be matched as closely as possible. However, even then, our results show that when working on a regular-sized display, chances are high that users might miss up to half of the important events. Therefore, conventional displays should only be used for dual-tasks when the events in the background are clearly distinguishable from other ones and do not appear at a high frequency. If users work on even larger screens (e. g. >35"), the possibility of missing events increases.

- **Horizontal displays are better for background awareness than conventional displays:** When designing for dual-task usage, horizontal displays should be given preference over a conventional display with offset. The horizontal display allows users to keep their primary task performance consistent while highly increasing users' ability to observe the background and identify events. Our horizontal display was slightly tilted to be edge aligned with the background, as was the conventional display in the *Opaque 30* condition. The horizontal display covered less visual area than the conventional display, mostly because of its orientation, which was beneficial for participants. Decreasing the visual area of conventional displays through tilting, however, is not feasible.
- **Transparent displays free users' visual field while keeping primary task performance constant:** Users can perform attention demanding tasks equally well on transparent displays as on conventional opaque displays. Furthermore, transparent displays highly increase users' ability to observe the background compared to a conventional display, *independently* of its rotation or position.
- **Users can choose between transparent and horizontal displays:** We did not find any statistical differences between these two display configuration. Therefore, users as well as designers of system can use both transparent and horizontal displays without sacrificing on primary task performance or ability to observe the background. Using a transparent display can be beneficial in situations where users prefer vertical displays (e. g. to increase the area from which the display is visible and increase comfort) over horizontal displays. However, our qualitative findings and suggest potential negative impact on comfort and posture of horizontal displays for longer-term usage. This needs to be subject of future investigation.

In general, we believe that the impact of angular offset from the central peripheral field highly influenced users ability for observation. Our results show that, while participants performed the primary task equally well across all conditions, their secondary task performance decreased with increasing visual field. The only exception to this is *Opaque 0*, where participants had to take a rather large effort to also see the secondary task. We included this condition knowing that it was impossible to see the background without additional effort to see users reaction and observe strategies. Participants tended to lean over to find to *optimal* viewing angle to see both displays. However, this additional effort drastically decreases secondary task performance. Therefore, in situations where the location of events in the background can be anticipated, even with optimal viewing position of a traditional display (i. e. *Opaque 30* in our experiment), users should be aware of the high probability of missing events in the background.

**The very large display experience**

Researchers in HCI often argue that increasing screen real estate increases efficiency and opens a wide range of applications. However, typical screens are more in the range of 20" - 30", and not coming close to the size of many TV sets at home. Only for very few occasions there seems to be justification for users to sit in front of a large wall of displays for their daily work (e. g. workers in stock exchanges observing a multitude of graphs simultaneously). We argue that one of the reasons conventional displays are not as large as they could be is because users would feel uncomfortable sitting behind a 150" wall of displays. Therefore, having a (at least partly) transparent display (e. g. [107]) potentially increases user acceptance and efficiency.

**Visual interference**

In our experiment, users saw the background as well as the primary task display during conditions with the transparent displays. While we found that distraction had no impact on task performance, we did not test for situations where on-screen contrast is low because of transparency. We believe that in order for transparent displays to be used in uncontrolled environments with arbitrary background, issues of visual interference need to be resolved. While outside the scope of this work, other research suggests several ways to eliminate visual interference on transparent displays such as heads-up-displays (e. g. [102, 163]). Incorporating these techniques and evaluating various display configurations (including transparent displays) is a necessary step to evaluate the usability of transparent displays.

# 7

# Creating augmented objects by combining shape-changing interfaces and spatial augmented reality

## 7.1 Introduction

The previous chapters introduced examples of dynamic objects. They are created from optically dynamic material to change properties such as their transparency, or perceived shape and size. In this chapter, we intrude an example of augmented objects. We enrich physically dynamic objects with spatial augmented reality to extend their space of possible appearances. By combining physically and optically dynamic interfaces, we leverage the benefits of both worlds and increase the expressivity of those interfaces.

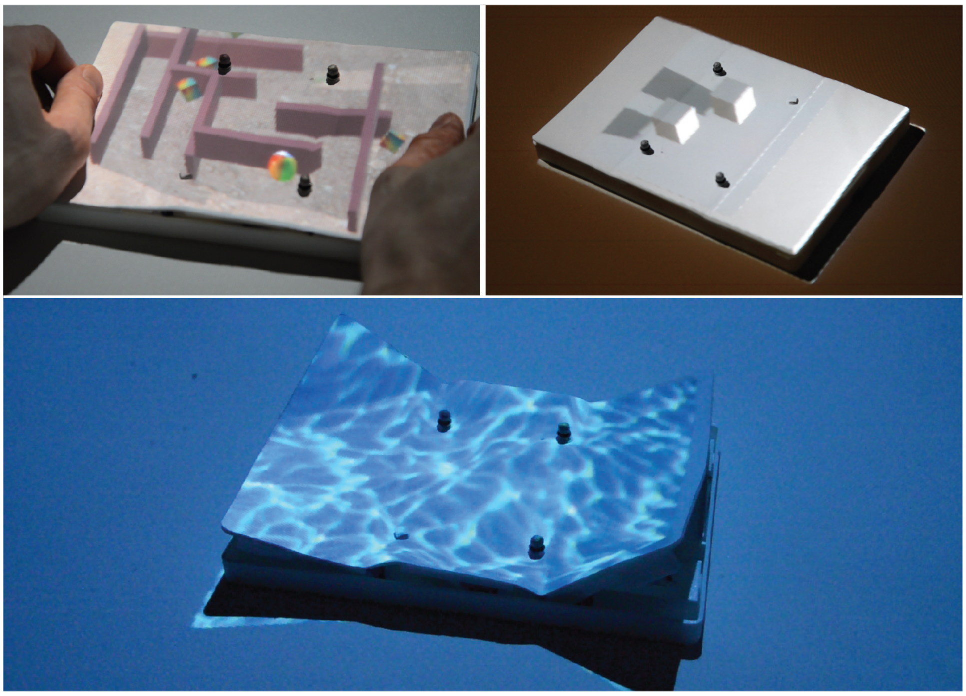

**Figure 7.1:** Shape-changing interfaces physically render coarse shapes and slow movements. Spatial augmented reality is used to display arbitrary textures, objects, and fast motion, directly onto the shape-changing device. This allows us to take advantage of natural depth cues and occlusion. We demonstrate the combination of the two techniques in three applications, i. e. a weather app with waves rendered physically and virtually (*bottom*) and a game with position of important game elements emphasized through shape change (*top left*). We show capabilities of enriching shape-changing interfaces with spatial augmented reality, e. g. the ability to render arbitrary objects and shadows (*top right*)
.

Shape-changing interfaces provide benefits with their ability to dynamically alter their physical shape. They allow for dynamic tactile sensation (e. g. [195]), haptic data exploration (e. g. [101]) or in general diverse dynamic affordances (e. g. [77]). We refer readers to the surveys by Rasmussen et al. [148], Coelho et al. [33], and Sturdee and Alexander [166] for comprehensive overviews over technologies and applications. However, changing the physical shape of devices to resemble desired target shapes can be challenging. Especially highly detailed shapes and fast motion are hard to achieve for shape-changing interfaces, which oftentimes only feature few actuators (e. g. MorePhone with 4 actuators [50]).

Typically, shape-changing devices are limited to their predefined range of possible shapes (i. e. deformations, or *states*), e. g. the faders of 2.5D displays (e. g. [45, 100]), or the predefined air-chambers of pneumatic interfaces (e. g. [195]). Additionally, even if fast physical motion is possible, it can be problematic since it could startle users (cf. [101]).

While research on shape-changing interfaces focused mostly on altering objects *physically*, a wide range of prior work aimed at changing the *optical* appearance of objects and environments, for example through spatial augmented reality (e. g. [16, 58, 83, 147]) or fish tank VR (e. g. [164]). Spatial augmented reality is defined as augmentation of real surfaces in a physically co-located way, for example through projection or displays. Users need no additional instrumentation such as see-through glasses (cf. [19]). By enriching static physical objects with visual content, users are presented with *the illusion* of different textures, shapes, or motion. These systems benefit from the high resolution of displays and projectors, typically featuring millions of pixels, whereas creating shape-changing devices with even only several hundred individually controllable elements (e. g. the 900 actuators of Relief [45]) is challenging. However, with spatial augmented reality, only the optical appearance of objects is dynamic, with the physical shape remaining static. Hence, benefits of shape-changing interfaces such as dynamic tactile sensation are not available. Furthermore, the ability of spatial augmented reality to render 3D graphics is limited by the particular display surface. While it is possible to render virtual objects smaller than the surface, it is not possible to render larger objects. Display performance also depends on the viewing position. If users look at a display surface from a steep angle, virtual objects cannot be rendered correctly since they would exceed the surface bounds and thus are cut off.

In this chapter, we propose combining shape-changing interfaces, typically mechanically constrained and low-resolution, with spatial augmented reality, which allows for displaying high resolution graphics and arbitrarily fast motion. Our aim is to bridge the gap between desired (real) objects and their physical and virtual representation. We argue that by incorporating perspective-corrected virtual renderings, shape-changing interfaces can come closer to an accurate visual representation of desired arbitrary objects. We use physical shape change for *dynamic tactile* *sensation and haptic feedback*, and combine it with spatial augmented reality's virtual graphics for *high-frequency textures and the illusion of fast motion*.

Until now, most research has focused on either physical *or* visual change. In the field of shape-changing interfaces, devices showed co-located 2D information (e. g. MorePhone [49]), were augmented with 2.5D height field data (e. g. Relief [100]), or extended with see-through augmented reality (e. g. Sublimate [101]). None of these approaches extends shape-changing interfaces with perspective-corrected 3D visuals in a physically co-located way. We achieved perspective-correction by tracking a user's head, while physical co-location is achieved using projection mapping. In contrast to prior work, we display 3D virtual content directly on the shape-changing device, which allows us to take advantage of natural depth cues and occlusion. Additionally, this allows for leaving users uninstrumented.

Conceptually, we see the connection between shape-changing interfaces and pixel displays analogous to the connection between (coarse) 3D models and high-detail 2D textures in computer graphics. The geometry of a 3D model is created e. g. through polygonal modeling. Subsequently, low complexity is compensated for through sophisticated textures. Therefore, various techniques such as bump mapping, shadow mapping or animated texture mappings are used to increase the realism of virtual objects. We frame our exploration of shape-changing interfaces and spatial augmented reality in this context, which allows us to identify challenges of both techniques and opportunities of their combination.

We demonstrate the concept of combining shape-changing interfaces and spatial augmented reality with a handheld prototype, more specifically a shape-changing tablet enriched with spatial augmented reality through projection mapping. *However, we envision future shape-changing devices to be fully covered with high-resolution displays.* More specifically, we believe that each surface of a device will incorporate controllable pixels, especially with the rise of flexible and printed displays (e. g. [50, 134]). We use projection for *simulating* such future interfaces. We present three applications demonstrating different interaction modalities: a spatial navigation application, an ambient display, and a labyrinth-style game.

In order to map the deformation of a physically constrained shape-changing device to arbitrary 3D graphics of spatial augmented reality, we present a real-time algorithm for matching the two based on what we call *mechanical distance fields* (*MDF*). Our MDF encodes mechanical deformation in a single regular-spaced voxel grid. By matching desired virtual target shapes with MDF, we can determine the necessary deformation to represent any virtual object. This allows us to match the deformation of shape-changing interfaces in real-time to virtual 3D content, even if the device features deformation along 3 dimensions. Furthermore, we present a technique for enhancing the quality of spatial augmented reality by incorporating a user's perspective, which we call *view-dependent shape change*. By dynamically altering the display surface, challenges of spatial augmented reality introduced by limited display space can be overcome. This way, we can render virtual 3D objects which would usually be cropped due to the lack of display surface.

**Contributions**

- An exploration of combining physically changing devices and spatial augmented reality for rendering arbitrary 3D objects and visual effects, framed in the context of computer graphics.
- A technique for extending the design space of shape-changing devices with rendering of complex textures, fast motion, and non-topologically equivalent shape changes.
- A real-time algorithm for matching constrained shape-changing devices to arbitrary 3D models.
- A technique for extending the effective display area for spatial augmented reality based on view-dependent shape change.

## 7.2 Design space

In this section, we first identify challenges when designing and building shape-changing interfaces taking related work into account. We then address these challenges and frame solutions with analogies to techniques from computer graphics.

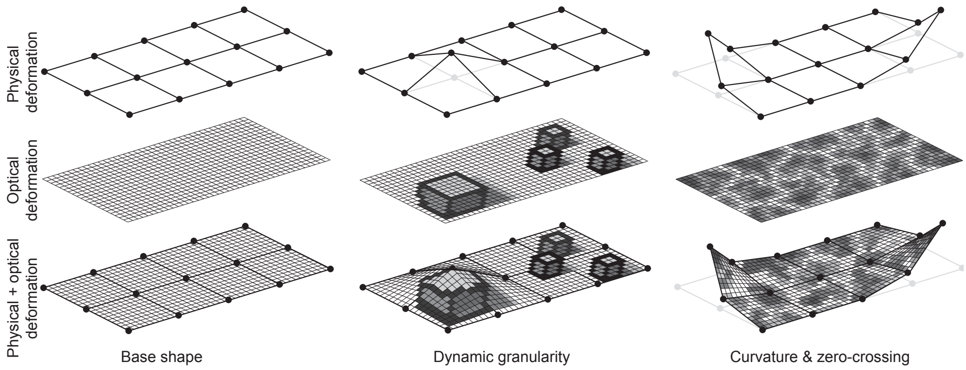


**Figure 7.2:** The physical deformation is combined with the optical "deformation", i. e. a shape-changing device with a pixel display. This allows for higher and changing granularity (*center column*), as well as to render varying curvature and a high number of zero-crossings (*right*). For illustration purposes the display is rather low-resolution ($15 \times 30$ pixels in this case). We envision the optical component to be a high resolution deformable display (e. g. >300 ppi). Figure adapted from [155].

### Challenges of shape-changing interfaces

Looking at work by Roudaut et al. [155] on shape resolution and the taxonomy of shape-changing interfaces by Rasmussen et al. [148], we can identify the main challenges we address in our work:

- *Limited granularity*: The limited number of actuators on many shape-changing interfaces makes rendering more detailed shapes or high frequency textures, such as bumpy surfaces or waves, challenging.
- *Limited speed*: Due to mechanical limitations, actuation speed is limited and high frequency motion is difficult to achieve. Additionally, even if faster motion was possible, it would often not be desirable due to user interaction (e. g. startle users, risk of jamming fingers).
- *Optical appearance relies on physical shape*: Shape-changing interfaces are bound to their physical appearance and environmental influence. Shadows thrown on the device depend largely on (uncontrolled) ambient light. Physical features such as edges of a device depend solely on the current deformation. Current devices cannot change the ways users perceive them beyond physically possible changes (e. g. perceived levitation or holes).

These limitations reduce the expressiveness of shape-changing interfaces and thus their ability to achieve various goals (cf. [148]). Devices with hedonic aims such as conveying emotion (e. g. Thrifty faucet [175]) could benefit by not only triggering emotion through bending but also through a highly dynamic surface texture that moves and shows small spikes, for example. Devices with functional aims such as communicating information (e. g. shape-changing mobiles phones [50, 155]) face challenges when trying to express information beyond the capability of their few actuators, e. g. indicating information about upcoming weather (e. g. strength of wind, cloudiness). Although spatial augmented reality does not allow for tangible change, it can give users the *illusion* of features such as high detail texture or high frequency motion. We argue that the combination with the tangibility of shape-changing interfaces opens more possibilities for researchers and designers. Additionally, we believe spatial augmented reality benefits from the combination with shape-changing interfaces by dynamically altering the display surface.

#### Connection to computer graphics: 3D models and 2D textures

We frame our proposed techniques in the context of *computer graphics*, more specifically in the context of polygonal 3D models and 2D texturing. We identified and applied three different techniques which we found to be applicable when using spatial augmented reality to address challenges of shape-changing interfaces: *bump maps*, *animated texture maps*, and *shadow maps*. Furthermore, we show how other optical properties of shape-changing devices such as transparency or area (cf. [148]) can be altered, based on the principle of *environment maps*.

### 7.2.1 Bump maps: Increasing perceived resolution

*Bump mapping* is a technique for the depiction of high detail surfaces with only low resolution 3D models, introduced by Blinn in 1978 [22]. The surface normals are changed, e. g. through perturbation, to render effects such as wrinkles or uneven surfaces, highly contributing to the perceived realism of surfaces. The geometry of the object is not modified. This allows for high realism, since the perceived intensity of light is mostly determined by the direction of the surface normal, and less by the position on the surface (cf. [22]).

Like for coarse 3D models, the granularity (i. e. the number of actuators with respect to the device area) of shape-changing devices is rather low (cf. [155]). Increasing the granularity can be challenging, since it means adding actuators and their control mechanisms on a limited area (e. g. within a tablet-sized device), and increased power consumption. In contrast, traditional displays or video projection feature several million controllable pixels. By adding virtual graphics to shape-changing interfaces, their perceived resolution can be increased. Analogous to using bump maps on 3D geometry, shape-changing interfaces provide the coarse, low resolution geometry. 3D graphics are responsible for *simulating* high resolution spatial and temporal variations (cf. high-detail 2D textures). Like 2.5D shape displays, regular displays can change their perceived granularity through simultaneously changing (i. e. grouping) actuators, ranging from changing only 1 pixel to all pixels simultaneously.

We use shape-changing devices to render the coarse (tactile) shape of a desired target, e. g. a wave-like form mimicking a water surface, or an architectural model. Spatial augmented reality is used for more detailed surface features such as bumps on a map for small hills and valleys, ground textures, or objects for games. Those features are displayed perspective-corrected to appear realistic to the observer. Figure 7.2 (*center*) illustrates changing a device with multiple control points physically, and multiple boxes added by 3D graphics, as well as the combination of the two. Granularity also influences other features of shape-resolution, such as curvature, zero-crossings (i. e. ability to produce wave-like shapes), and closure (cf. [155]). As noted by Roudaut et al., curve deformation is challenging to achieve due to different material properties. Shape-changing devices need to be manufactured from, or covered with, flexible material (e. g. foam or Lycra [34, 100]). Dependent on the amplitude and curvature, only a relatively low number of zero-crossings can be achieved. This problem is even more apparent for devices with only very few actuators, e. g. shape-changing mobile phones. In combination with spatial augmented reality, the number of available perceived zero-crossings, i. e. the frequency of wave-like shapes, can be increased drastically. As illustrated in Figure 7.2 (*right*), low frequency waves are rendered physically, whereas smaller wave-like structures are simulated.

### 7.2.2 Animated texture maps: Increasing perceived speed

Animated texture maps are used for presenting predefined animations on other virtual surfaces, without changing the actual surface geometry. The animations can be of arbitrary speed and size.

Roudaut et al. discuss speed and animation, however mostly in terms of a device's ability to perform actions with varying speed. While the speed of most shape-changing interfaces is fixed (i. e. determined by mechanics, cf. [155]), we argue that in combination with spatial augmented reality, perceived speed of shape-changing devices can be varied as well as increased beyond the physically possible motion of the device. Like with *animated texture maps*, which are used to render motion on static or low resolution surfaces, we render fast virtual motion on the rather slow moving shape-changing interface. Additionally, we render motion of more fine-grained textures (e. g. small waves) and objects (e. g. a rolling marble).

The maximum velocity of a shape-changing interface depends on its implementation. Devices such as 2.5D shape displays with motorized pins rely on the actuation speed of the pins, whereas pneumatic interfaces rely on the pumping and vacuuming speed. Dielectric electro active polymers and shape-memory alloys usually rely on rather high voltage or current, respectively, and oftentimes suffer from low (de-)activation speeds. Higher velocity motion would be possible but poses challenges in terms of manufacturing as well as user interaction. When the velocity is increased above a certain threshold, it is challenging to accurately match a desired target deformation due to the mass of the object and resulting oscillation around the target position. Additionally, high speed deformation poses potential risks to users (e. g. jamming fingers or startling users).

By combining shape-changing interfaces and spatial augmented reality, high velocity movements can be performed virtually. Physical deformation can be performed simultaneously, but with lower velocity, overcoming these potential limitations. In our prototypes, physical actuation is included in our shape-matching algorithm with predefined maximum velocity and a damping factor, allowing for smooth motion which does not distract or startle users.

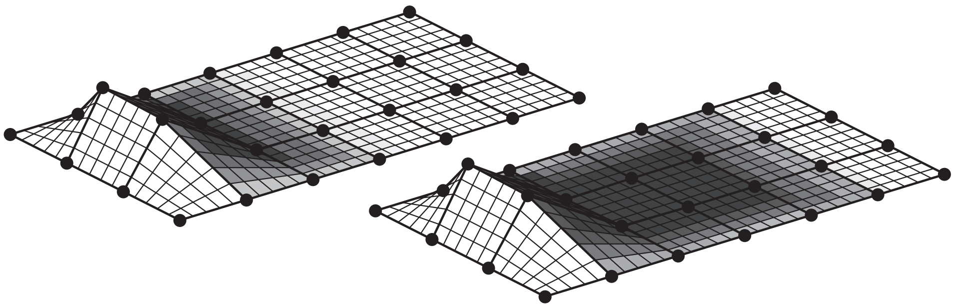


**Figure 7.3:** Two different virtual shadows for the same physical shape.

### 7.2.3 Shadow maps: Adding virtual depth

Shadow mapping, introduced by Williams in 1978 [189], is a technique for efficiently rendering shadows. A shadow map is generated by rendering the scene from the light source's perspective and storing depth information (z-buffer values). Through comparison of the scene part displayed in each pixel, the position of shadows is determined. This allows for greatly increasing the realism of virtual scenes. Additionally, by moving the virtual light source, different effects such as change in time (e. g. virtual sun clock) and perceived object position (e. g. levitation) can be achieved.

Current shape-changing interfaces are bound to their physical appearance when rendering objects. Occlusion or shadows depend on the current shape of the device and the illumination coming from the environment. Using spatial augmented reality allows for simulating these properties. We change the position of a virtual light source in the "scene" (i. e. the environment where the tracked shape-changing device is in), effectively adding virtual shadows to a physically deformed device (see Figure 7.3). This allows us to illuminate the device in a way that gives users the impression of different shadows on the device, independently of ambient illumination from the environment. Previously, this was not possible since shape-changing interfaces relied on actual shadows thrown on or through the device. Shadow maps allow designers of shape-changing interfaces to convey different information, e. g. progress of long running processes, or simply time passing, indicated by shadows moving according to the sun (i. e. our virtual light source).

Additionally, *shadow maps* can be used to change optical features of shape-changing interfaces. As illustrated in Figure 7.4, by emphasizing or blurring physical edges, their appearance can be altered. By emphasizing soft edges with sharp lines, the edges appear more pronounced, and vice versa.

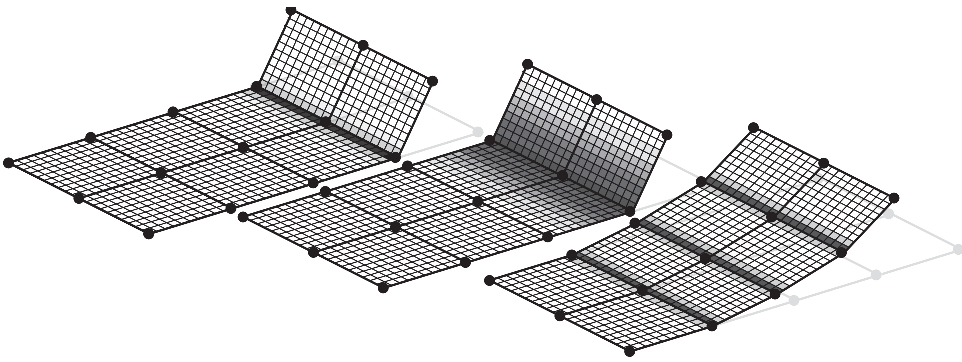

**Figure 7.4:** By adding emphasized or blurred virtual edges, perceived shapes such as the sharpness of edges can be altered.

Finally, spatial augmented reality allows for rendering arbitrary objects on shape-changing devices. This makes it possible to render a background texture on the device as well as other objects including their virtual shadow. By exploiting this property, it is possible to create virtual objects on the device that e. g. float above the ground (see Figure 7.5 and 7.1), or disappear and reappear on demand. This shadow effect has been used in prior work to make fairies fly in front of a statue as part of culture heritage communication at a historic castle [37].

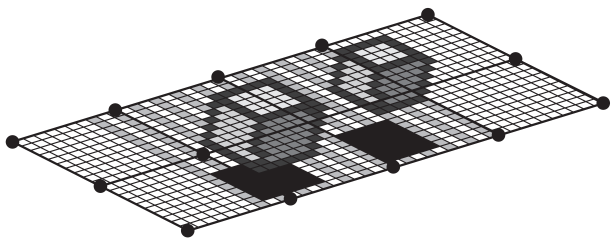

**Figure 7.5:** Two virtual cubes are rendered on the shape-changing device. By adding virtual shadows, the illusion of objects hovering over the device can be created.

### 7.2.4 Extension: Environment maps for optical effects

Besides increasing the perceived resolution and speed of shape-changing devices, other optical effects such as transparency can be included in future shape-changing interfaces. By showing the background on top of the device, features such as holes (i. e. partial transparency) can be rendered. This also allows for decreasing the perceived size of a shape-changing device, which is challenging with current technology.

This approach, however, requires knowledge of the environment, such as the surface underneath, beside, or behind the shape-changing device, thus deviates from the other proposed concepts. Therefore, we consider changing these optical properties of shape-changing interfaces as *extensions* of our proposed framework.

In computer graphics, this technique is referred to as environment maps (or reflection maps, [23]). The texture of the environment is captured and stored for later mapping it on a virtual device. This allows for efficient rendering of reflection without having to render the geometry of the environment. Our current implementation uses projection mapping, which allows us to control the appearance of the environment by using it as projection surface (see Figure 7.6).

**Figure 7.6:** By rendering contents on the table, the device blends into the environment and can render apparent holes.

## 7.3 Matching physical and virtual shapes

One of our main objectives is to find appropriate physical and optical properties of shape-changing devices to match desired virtual objects. For achieving this goal it is therefore central to match the physical shape of the device to the shape(s) in the virtual environment.

We note that this step was comparatively simple in prior work, as it focused either on 2D information displayed on shape-changing devices or on height field data displayed on 2.5D shape displays. In our work, however, shapes may cover an arbitrary volume in 3D and they need to be represented by shape change that goes beyond a simple change of height.

Consequently, our goal is to design an algorithm that computes the actuation parameters of the shape changing device so that the virtual model is fit as closely as possible, and in real-time. In the following we develop a model for the problem that makes the idea of a *close-to-optimal fit* precise and facilitates fast computation.

### 7.3.1 Overview

Our algorithm takes a model of an actuated device (e. g. as a set of 3D models, one per deformation state) as input. The input, representing individual actuators and states, is then encoded into multiple 3D voxel grids. Each voxel holds information on how the device needs to be deformed to cover it. The individual actuators of a device are encoded separately and later combined into a single voxel grid. This representation allows us to store actuation efficiently, and enables real-time matching.

For matching a shape-changing device to an arbitrary target shape (e. g. given as 3D model), we first decode the target into a voxel grid as well. We then overlay the initially computed voxel grid with the target voxel grid and take the overlapping voxels into account for fitting. Subsequently, we calculate the optimal fit between the initial and the target voxel grid (see below). Figure 7.7 illustrates the complete algorithm.

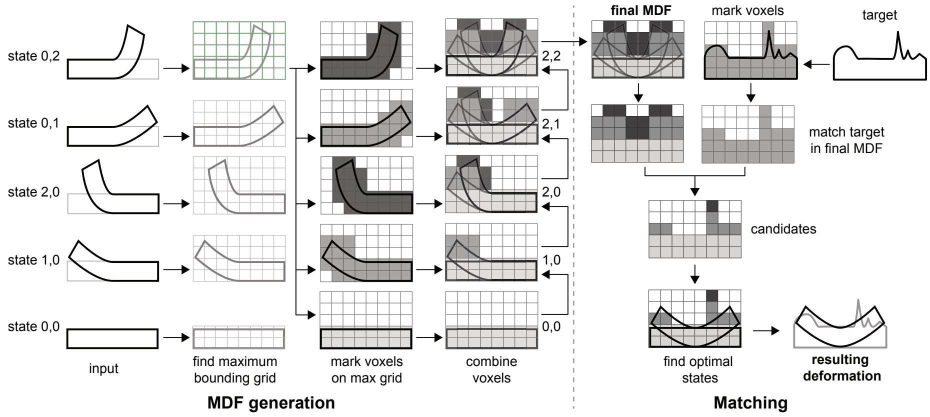


**Figure 7.7:** Computation of the initial mechanical distance field (*left*) and matching to a target shape (*right*) in a 2D example. Each state of the device (*left*) is encoded individually, and combined to a single MDF (*center*). The target object is also encoded into a voxel grid and then matched with the initial voxel grid. This gives use candidate voxels (encoding device actuation). We then find the close-to-optimal fit in the MDF, which gives us the resulting deformation for any given target (*bottom right*).

### 7.3.2 Representation

The most important part of the algorithm lies in the representation of the shape changing device. Typically, a shape-changing device is composed of multiple actuators (i. e. degrees of freedom), which, when activated, deform a specific part of the device. The resulting motion need not be linear, which is typical for devices such as shape-changing mobile phones (e. g. [50, 155]). We assume the actuation level of each actuator to be discretized. Then we represent the *state* of the shape-changing device as $\mathbf{s} \in \mathbb{Z}_m^n$, where $n$ is the number of actuators and $m$ is the number of actuation levels per actuator. Note that there is a total of $n^m$ distinct states for the device.

We represent the shape of the device as a discrete grid of voxels above the zero-state $\mathbf{s} = (0, \ldots, 0)$ of the device (i. e. all actuators in state 0). Each voxel is either *covered* or *uncovered* by the device, where covered means the surface has been pushed far enough such that the voxel is below the surface. The status of each grid point is represented by 0 for uncovered and 1 for covered. The shape is a binary vector depending on the state, i. e. $V(\mathbf{s}) \in 0, 1^{x \times y \times z}$, where $x, y, z$ are the dimensions of the voxel grid in space. We note that the discretization need not be rectilinear, however, this is clearly the most convenient choice. We require that the voxel representation can be computed for any state $\mathbf{s}$, i. e. that the shape-changing device is known a priori.

The virtual target shape to be represented by the device is likewise represented in the volumetric grid and denoted $V_T \in 0, 1^{x \times y \times z}$. In a typical scenario the target shape is given as manifold, intersection-free mesh without boundary (sometimes referred to as 'watertight'), which can be scan-converted into the voxel representation. In principle, however, our approach works with every shape representation that can be converted into a voxel grid.

### 7.3.3 Reduction and simplification

In general it is not feasible to store the grid for each of the $n^m$ possible states, as this would require $xyz \cdot n^m$ bits storage, which is cumbersome for realistic dimensions. In particular, real-time computation would be difficult on this data volume.

Our solution is to *decouple* the individual actuators. This means we will consider the effect of each actuator on the shape changing device *independent* of the state of the remaining actuators. This is justified as only few voxels are actually influenced by more than one actuator. Moreover, we do model the interaction of actuators, albeit probabilistically: we consider the expected value for a voxel depending on the state of an actuator, considering the likelihood of the actuation levels of the other actuators being equal. For example, if a voxel is covered at state $j$ of actuator $i$ for half of the possible states of the other actuators , we consider the value of the voxel to be 0.5. This means we can now store the voxel grid for each actuator $i$ and each actuation level $j$ independently in a real valued vector $\tilde{V}_{i,j} \in [0, 1]^{x \times y \times z}$. Note that now the complexity has reduced from using $n^m$ to using $nm$ grids.

We further simplify this representation, making a *monotonicity* assumption: as the actuation levels of the actuators are increased, the enclosed volume never shrinks. This means, if a voxel is covered for actuator $i$ at level $j$, it will also be covered for any level $j' > j$. We notice that most voxels are affected only by a single actuator. Such voxels will at some actuation level change their value from 0 to 1. Rather than storing all 0's and 1's it suffices to store the actuation level for the change. Furthermore, voxels that depend on more than one actuator usually behave linearly, and we decided to store the values of any voxel for actuator $i$ as a function that is equal 0, then linearly increases to 1, and then stays constant. This means we can store values of each voxel for a given actuator as two actuation levels: the first actuation level at which the value is non-zero, and the first level for which the value is 1. For voxels that are dependent on a single actuator, the two values will be identical. We store this representation as two vectors $V_i^0, V_i^1 \in \mathbb{Z}_m^{x \times y \times z}$. When needed, this information can be used to quickly generate the values $\tilde{V}_{i,j}$.

The representation of the shape for a single actuator has a nice intuitive visualization: the actuation levels $V_i^0, V_i^1$ symbolize the 'mechanical work' that has to be exerted by the actuator to 'reach' the voxel. Based on this idea, and in analogy to distance fields (cf. [24, 85, 159]) we nickname the representation *mechanical distance field*. The generation of the MDF is visualized in Figure 7.7 and the result for the set of all actuators combined in our prototype is shown in Figure 7.8.

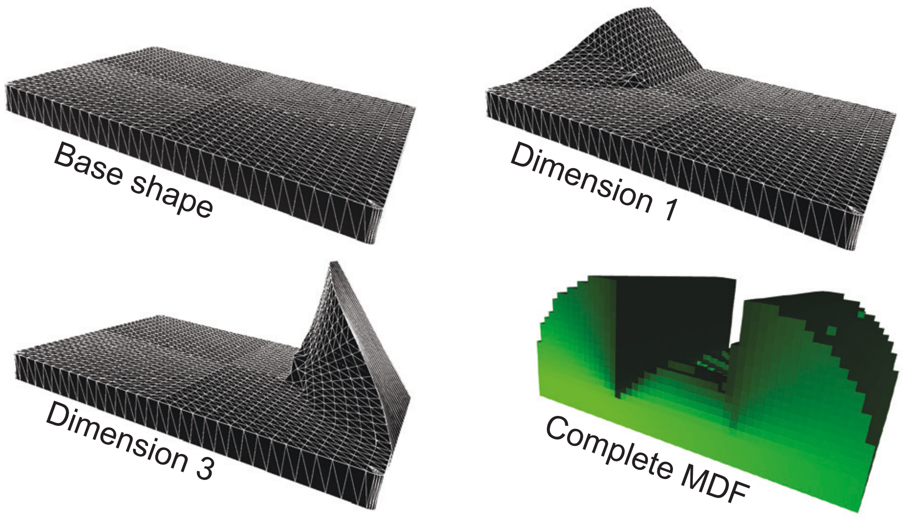


**Figure 7.8:** Example of a fully computed MDF (*bottom right*) for a device (base state 0 *top left*). Two of six dimensions of deformation (2 actuated states for dimensions 1 and 3) are illustrated. Darker green indicates larger state.

### 7.3.4 Real-time matching

Based on the precomputed grids $\tilde{V}_{i,j}$ we use a voting approach to decide on the best state $\mathbf{s}_T$: for each actuator $i$ we check all actuation levels $j$ and try to maximize the fit between the volume represented by the shape changing device and the target shape.

We assess the error of fit for level $j$ for actuator $i$ as the symmetric difference between $|\tilde{V}_{i,j} - V_T|$ (where the absolute value $|\cdot|$ is performed element-wise). Note that this evaluates to zero whenever the values are identical; and to one whenever either the voxel should be covered but is not, or should be uncovered but is. Naturally, we want to choose $j$ such that this value is small.

On the other hand, we cannot expect to cover the shape with a single actuator so the error will usually be large. Among states with large error we would prefer those that represent most of the volume of the target shape, i. e. we would like to maximize the intersection of the target shape and the volume covered by the device. This volume is given by $\tilde{V}_{i,j} \cdot V_T$.

Based on these measures we defined the optimal state for actuator $i$ as

$$s_i = argmin_j \frac{|\tilde{V}_{i,j} - V_T|}{\tilde{V}_{i,j} \cdot V_T}. \tag{7.1}$$

For numerical reasons we only consider values $j$ for which the common interior volume $\tilde{V}_{i,j} \cdot V_T$ is bounded away from zero. Together these values form the optimal state vector $\mathbf{s}_T$.

#### Computational considerations

We first determine the size of the voxel grid by computing the bounding box of the shape for all states. The smallest side length of the bounding box divided by the resolution $r$ defines the side length of the voxels. In the case of our tablet prototype, as an example, we sample the deformation of each actuator $r = 33$ times, resulting in an initial voxel grid of $x = 83 \times y = 33 \times z = 59$ (~160K total) voxels, which gives us a good balance between resolution of the mechanical deformation and computational effort.

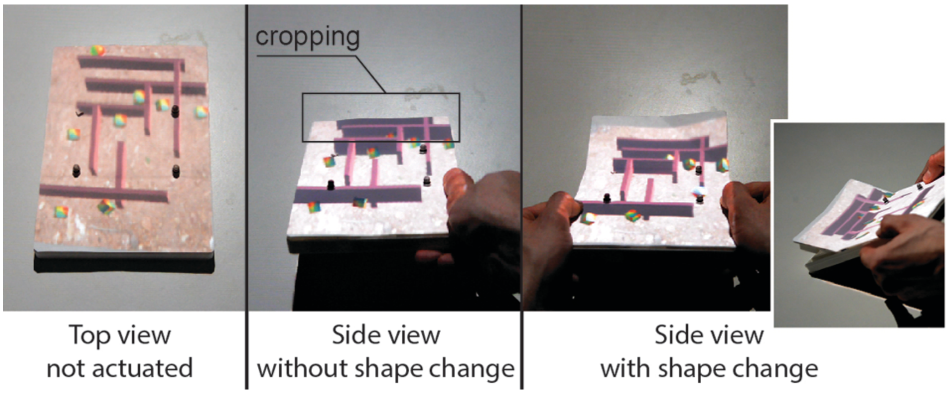


**Figure 7.9:** The virtual world displayed on the shape-changing device (*left*). Without shape-change, the display area of the device is not sufficient for users to see the virtual world fully (i. e. walls are cropped, *center*). With view-dependent shape change, the virtual world can be seen fully (*right*, deformation state *inlay*).

The computation of all best fits can be performed in a single iteration over all voxels, collecting the products and differences for each actuator, and calculating the best fit. For the above mentioned voxel resolution our algorithm runs in real-time, with the most expensive part being the intersection and difference computations. The complexity of the algorithm grows linearly with the number of voxels, actuators, and actuation levels.

## 7.4 View-dependent shape change

Typically, the ability of spatial augmented reality to present 3D contents to users is limited by the display surface. If a virtual object extends beyond the bounds of the projection surface, it is cropped. This is true for projection mapping as well as for other techniques such as stereoscopic displays. We make use of the shape-changing devices' ability to extend its surface to avoid cropping without the need to wear glasses. The surface of the device is deformed depending on the angle between the user and the device. In our prototype, this means that an edge is bent upwards to increase the effective projection surface area, as shown in Figure 7.9. Figure 7.10 illustrates the computation of the deformation. The angle $a$ between the user and the device is calculated as the dot product between the viewing direction $\vec{u}$ and the normal of the device $\vec{n}$. We then calculate the deformation simply as $\beta = \theta - \alpha \times \frac{\alpha}{\theta}$, where $\theta$ denotes the maximum deformation of the device (in our case, 90°). We deform the actuators which are furthest away from the user.

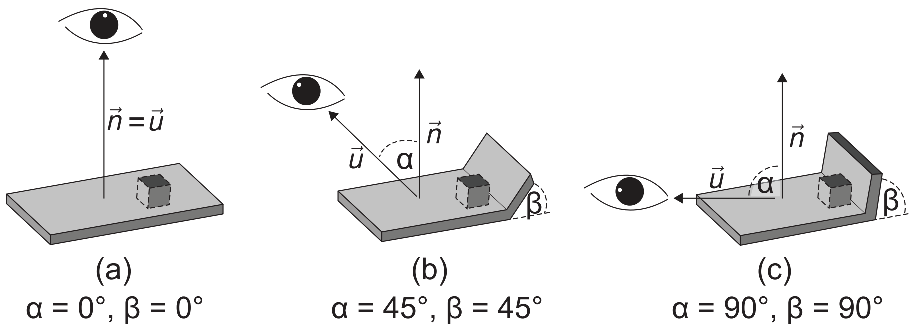


**Figure 7.10:** The shape-changing device is deformed to enable the best viewing angle on virtual objects displayed on the device.

## 7.5 Implementation

We built a prototypical shape-changing device enriched with spatial augmented reality through projection mapping to illustrate the concept of combining the two. Software and hardware is released as open-source[1].

### Shape-changing tablet

We built a shape-changing tablet (21 cm × 15 cm × 2 cm) from a custom 3D printed housing with 6 servo motors (Flug51 SG90) as actuators. The design of the device was inspired by Rasmussen et al. [149], who were using a smaller, phone-sized and wired version with 4 actuators for their user study. The device is powered by two 2000mAh LiPo batteries. The servo motors are controlled through an Arduino Pro Mini, connected wirelessly via Bluetooth (HC-06 module) to a computer running our custom software. The top of the device is 3D printed from flexible filament (NinjaFlex), which allows for bending according to the servo position. For increased projection quality, the top is additionally covered with a sheet of paper (see Figure 7.11).

[1] http://www.cg.tu-berlin.de/research/projects/sci-and-ar/

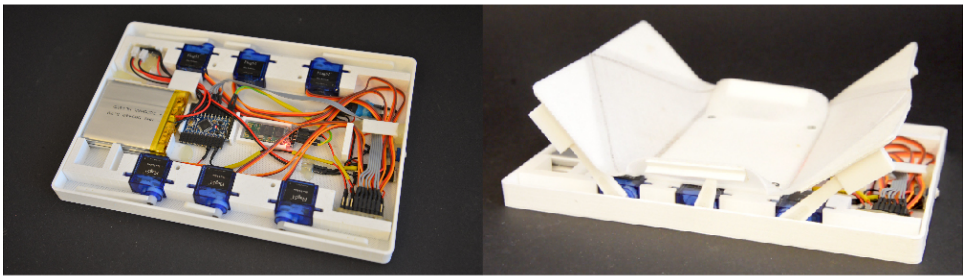

**Figure 7.11:** Shape-changing tablet containing 6 servo motors. Each corner can be bent, as well as the center of the tablet.

### Projection mapping

3D projection mapping is based on having a 3D model of the physical object(s). This allows us to incorporate the device's geometry into our projection mapping system. We use a rigged 3D model of the tablet within our projection mapping application. By positioning and calibrating the projection system so that the relationship of the projection to the physical object corresponds to the virtual camera's relation to the 3D model, we can project the digital model onto the physical elements of the installation, thereby augmenting the physical object. We track the device as well as the user's head position using an OptiTrack IR tracking system. Projection is done using a single projector positioned ~2 m above the surface.

### Software

The control software for our sample applications is written in Unity, with rigged models of our prototypes created in Blender. We use a proprietary software for calibrating our projection mapping system, written in Java. The MDF algorithm is developed in C++ using openFrameworks and CGAL [174]. For voxelization of shapes, we use a simple box-triangle intersection algorithm as described in [1]. The C++ application handles generation and persistence of the MDF. The Unity software handles real-time shape matching.

## 7.6 Applications

We demonstrate three different applications showcasing the potential benefits of combining shape-changing interfaces with co-located 3D graphics.

### Spatial navigation

Users navigate in a virtual 3D environment, i. e. a 3D map. The shape-changing device deforms according to the geometry, as depicted in Figure 7.12. Details of buildings and places are rendered onto the device using 3D co-located graphics. Users can rotate the device to explore the map freely. This application uses our view-dependent algorithm. When the user look from an angle, the deformation adapts, making it possible to render features extending the bounds of the projection surface (e. g. by tilting the device 90°).

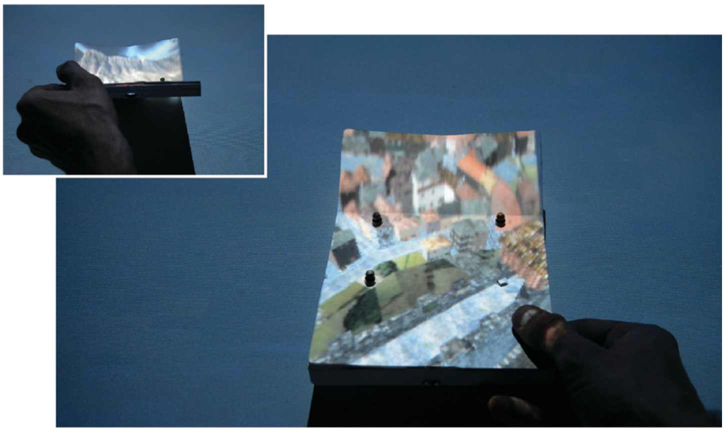

**Figure 7.12:** Users navigate on a 3D map. Tilting of the tablet is compensated for using view-dependent shape change (*inlay*).

### Ambient display

Information of current and upcoming weather and the time are displayed on and with the shape-changing tablet (Figure 7.13). Virtual shadows of 3D objects convey information about time of the day, following a virtually moving sun. Additionally, a wave simulation is rendered physically through deformation and virtually through spatial augmented reality. The spatial and temporal frequency depends on the strength of wind. The shape-changing tablet deforms according to the wave simulation (i. e. stronger waves result in faster and larger deformation). Spatial augmented reality is used for rendering more fine grained details of the simulation. Furthermore, we added virtual fog to convey information of wind strength. The fog seems to float on top of the tablet.

A problem with shape-changing interfaces is that sudden deformations are not always suitable. To solve this, we implemented the concept of *anticipating shape change*. First, only the virtual waves move on top of the device indicating an upcoming shape change. With a short delay, the physical movement starts.

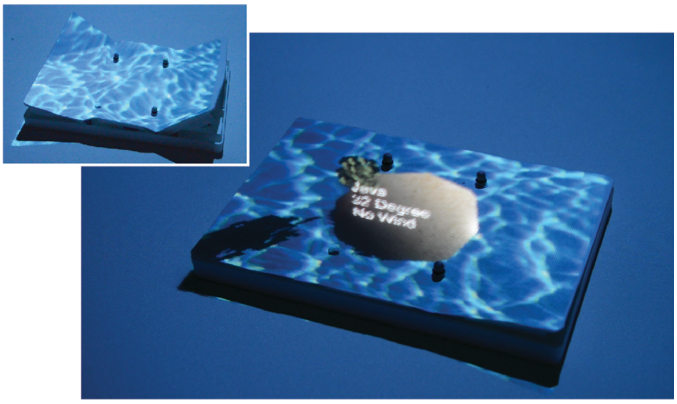


**Figure 7.13:** An ambient display conveys information about time and weather. Waves are rendered physically and virtually (*inlay*).

### Labyrinth game

We implemented a simple labyrinth game (Figure 7.14). Users control a (virtual) rolling ball and maneuver it around to pick up colorful objects to complete a level. 3D information of the environment is rendered virtually on the device, including perspective-corrected walls, elevated game objects and high frequency ground textures. The ball also serves as input for our shape-matching algorithm and triggers deformation. Especially for regions covered by player's hands, this seeks to increase the feeling of a physical game object. Additionally, game effects such as stairs are rendered as a combination of shape-changing and virtual graphics.

### Additional application using dynamic porosity

By using knowledge of the background, we can render holes into the shape. The background is created by our application, therefore no camera is needed. We implemented this in a simple physical simulation (see Figure 7.6). The virtual ice first has holes, then breaks apart, thus revealing the underlying surface.

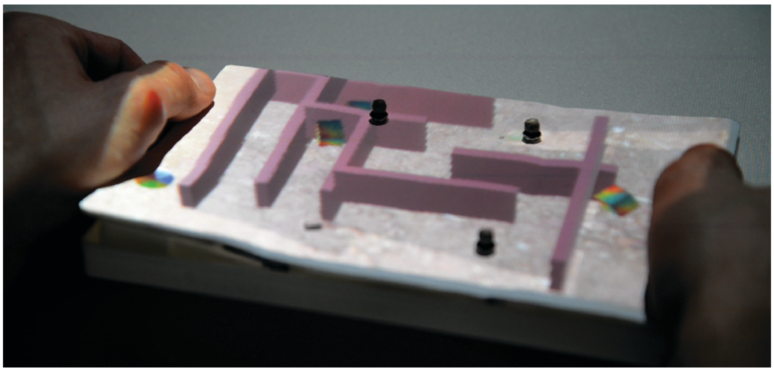

**Figure 7.14:** In the labyrinth game, the position of the ball (*bottom left*) is also rendered physically for greater immersion.

## 7.7 First user experiences

We conducted a small informal user study with 5 participants (24 - 36 years), all students and staff from a local university, for getting first insights into our proposed technique. The study was performed as a think-aloud session. Participants, all unexperienced with both techniques, were first introduced to the applications described earlier. Thereafter, they experimented with the applications themselves. For the ambient display, participants reacted positively to the combination of the two techniques. Participants immediately understood that physical shape-change corresponds to the virtually rendered waves. One participant noticed: "Oh, now it is clearly getting windy!" after the shape change started on the weather simulation and virtual fog was moving faster. Another participant suggested that the device should spray water, indicating that the device becomes more organic when it is enriched with 3D content. For the labyrinth game, participants felt like the virtual game world was moving whenever the ball was at a position featuring actuation. This makes us believe that participants considered the device to resemble the objects they saw (i. e. the device *becomes* the water or the game world) and not just serve as a container for the displayed elements. This is different to other work, where virtual content was rendered and controlled through a "container" device, such as pCubee [164]. The combination of the two techniques increased the "realism" of the device and, consequently, immersion. The correspondence between the physical and virtual deformation guided users' perception. Researcher can take this into consideration when designing shape-changing interfaces.

## 7.8 Discussion

The combination of shape-changing interfaces and spatial augmented reality allows for a wide variety of applications and extends the range of perceived appearances of shape-changing interfaces. It enables shape-changing interfaces (with inherently limited spatial resolution) to feature high-resolution 3D content that can overcome physical constraints. Furthermore, shape-changing interfaces only have to render slow and coarse motion, since arbitrarily fast motion can be rendered with spatial augmented reality. Lastly, the combination can be used for rich haptic exploration of 3D visual information, allowing to better convey information. We also believe that by increasing the resolution of shape-changing interfaces, devices can not only serve as a ”container” for digital content, they can also come closer to actually representing virtual objects. This would allow them to serve as *avatars* of digital objects, currently only present as on-screen content. By enabling shape-changing interfaces to effectively transition between representations, the space of possible usages can be drastically increased. Incorporating high detail 3D graphics is a step in this direction.

**Tangible qualities**

Spatial augmented reality allows for changing the optical appearance of a shape-changing device, however these changes are intangible. As soon as users touch the surface which is displaying a high frequency texture, the illusion breaks. Furthermore, only the perceived physical shape of an object can be altered. Other properties such as stretchability, viscosity or strength remain unchanged. They rely completely on the resolution of the shape-changing device. Therefore, it is desirable to increase the range of shape-changing interfaces as well as the resolution of incorporated spatial augmented reality. Including tactile technologies such as TeslaTouch [10] in future devices would help overcome restrictions of combining shape-changing interfaces with pixel displays. Furthermore, the visual appearance of an object is tightly coupled to its physical shape. While our concept allows for altering an object’s perceived appearance, large-scale changes (e. g. doubling its size) are challenging to achieve. Thus, there are limitations in terms of possible optical illusions. We address this challenge in Chapter 8.

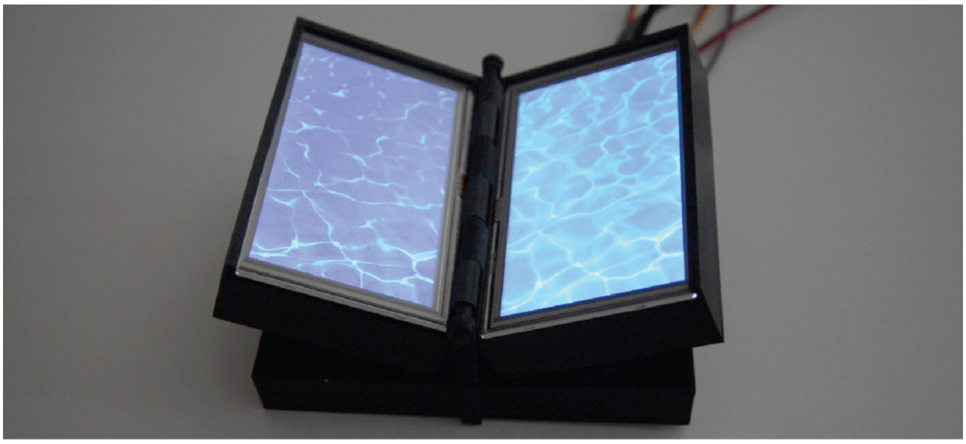

**Figure 7.15:** Our new prototype of a shape-changing device with two displays and two actuators features significantly higher pixel resolution.

**Display quality**

Typically, the field-of-view of spatial augmented reality approaches is limited. If users look at a surface from a very steep angle, graphics disappear or are cropped (like with bump maps). We partly overcome this problem with our view-dependent shape change approach, however, this solves the problem only to a certain extent. Furthermore, spatial augmented reality relies on perspective correction with respect to user perspective. However, we believe that our approach is superior to currently used see-through augmented reality technologies because it features natural depth cues and occlusion.

Our vision is that spatial augmented reality is achieved by OLED-style displays wrapping around shape-changing interfaces, or any other display technology featuring high-resolution flexible displays. Currently, our system only features displaying digital content through projection mapping with a single projector, which gave us the ability to develop and test our concepts. However, we argue that our approach is agnostic to the way the image is generated. We plan to extend this approach to cover shape-changing devices with digital content on all sides. This would allow for better showcasing how this system solves the surface deformation for not just 2D with height information, but real 3D. We implemented a first prototype of a shape-changing device with two physical displays and two actuators (see Figure 7.15). The display resolution is significantly higher than when using projectors and does not suffer from projector occlusion. In the future, we plan to extend our prototype with flexible and autostereoscopic displays and an increased number of actuators. Furthermore, using autostereoscopic displays instead of regular displays would allow us to display 3D content to multiple users. Finally, incorporating our technique in shape-changing devices featuring physical change along all sides will bring us closer to our goal of accurate representation of virtual 3D objects in real life.

# 8

# Changing the appearance of real-world objects by modifying their surroundings

We present an approach to alter the perceived appearance of physical objects by controlling their surrounding space. We call this concept *augmented surroundings*. Many real-world objects cannot easily be equipped with displays or actuators in order to change their shape. Common approaches such as projection mapping enable changing the appearance of objects without modifying them. Certain surface properties (e. g. highly reflective or transparent surfaces) can make employing these techniques difficult. In this chapter, we present a conceptual design exploration on how the appearance of an object can be changed by solely altering the space around it, rather than the object itself. In a proof-of-concept implementation, we place objects onto a tabletop display and track them together with users to display perspective-corrected 3D graphics for augmentation.

**Figure 8.1:** We modify the size of object such as the book (enlarged in width and height). A set of keys is virtually raised and tilted by placing them on a rendered ramp. The transparent bottle is augmented with virtual colors to remind users to stay hydrated.

This enables controlling properties such as the perceived size, color, or shape of objects. We characterize the design space of our approach and demonstrate potential applications. For example, we change the contour of a wallet to notify users when their bank account is debited. We envision our approach to gain in importance with increasing ubiquity of display surfaces.

## 8.1 Introduction

In the virtual world, from traditional desktop computing to virtual reality, changing the appearance of objects or interfaces is one of the main means to communicate information. Objects are altered in color for emphasizing them, they are hidden or revealed when needed, or resized to afford manipulation or communicate importance. Objects in *virtual* environments are easy to modify: research in computer graphics explored creating complex visual changes in real-time, for example through dynamic bump mapping [22] or environment mapping [23]. With the help of these techniques, arbitrary objects and alterations are achieved.

Research in HCI and computer graphics aimed at enabling similar techniques for objects in the real world. Projection mapping (or spatial augmented reality) enables changing the appearance of physical objects in real-time and on-demand (e. g. [8, 37, 108, 147]), without the need to actually modify the object. Projecting onto objects, however, is not always possible. Objects with transparent, dark or reflective surfaces are unsuited for projection, and would lead to drastically compromised image quality. Furthermore, projection mapping only allows for alterations *within the bounds of a target object*. The color or texture of an object may be changed, e. g. to optically shrink an object, however, enlargement is not possible due to the lack of projection surface (cf. [108]). A different approach is to create objects directly from optically dynamic material (e. g. [110]), or equip them with in-situ displays (e. g. [134]). These approaches and work on physically dynamic interfaces (cf. [77]) and ubiquitous computing (e. g. [187]) usually modify objects by including functionality through sensors, displays or optically dynamic materials. This, however, increases manufacturing complexity and later modifications would require tampering with the underlying hardware.

We propose a different approach. Instead of augmenting objects directly, we *change their appearance by altering their surrounding space.* This enables changing the appearance of objects without physical modification, and of objects with surface properties unsuitable for projection. This work provides a conceptual design exploration of our proposed approach and an initial proof-of-concept implementation. We take inspiration from work on *visual illusions*, which typically alter the perception of an object through other objects that are present in its surroundings, similar to our approach. Furthermore, we build on computer graphics research on how objects are visually altered (e. g. through normal or environment maps). We create optically dynamic objects and contribute one additional technique (besides existing techniques such as projection mapping, shape-changing interfaces, and optically dynamic materials) for bridging the virtual and the real world. We envision display surfaces to be ubiquitous in the future, for example through electronic wallpapers or projection (e. g. [146, 178]), so that objects are commonly surrounded by display surfaces. Our approach then allows augmenting objects e. g. for communicating information and status.

Effects such as changes in size, position, or color are rendered perspective-corrected, which is enabled by tracking objects and users. As an example, we extend physical objects with virtual shapes such as the book in Figure 8.1 to increase its size. By showing virtual content such as platforms or ramps underneath an object, we can alter its perceived position, e. g. visually elevated by placing it on a platform. We change the color of objects by displaying differently colored areas in their close proximity. To decrease visual saliency and effectively "hide" objects from an observer, we blend them with the environment.

We demonstrate a proof-of-concept realization of our concept with a conventional tabletop display and an optical tracking system. We take inspiration from prior work on anamorphic graphics (e. g. [15]) and ubiquitous display surfaces (e. g. [146, 178]). We show the possibility of changing the perceived size, position, contour, color, or visibility of real-world objects. In our current implementation, objects are manually specified by users or are automatically recognized by our system based on predefined marker sets. We employ these effects in three scenarios: as ambient displays, for notifications, and for increased privacy. In the ambient display scenario, the color and position of existing objects (water bottle, medicine box, keys) are changed to communicate status. For displaying notifications, we change the perceived contour of a wallet to get dynamic spikes for indicating bank transactions. Lastly, by displaying colored areas in an object's surroundings we change its visibility to "hide" it from distant observers for increased privacy.

**Contributions**

- We provide a design exploration of the concept of altering the appearance of existing, unmodified objects by displaying content in their surrounding space and characterize the design space of the approach.
- We provide a proof-of-concept implementation with a conventional display and an optical tracking system. It allows changing the appearance of objects without equipping them with additional output capabilities.
- We showcase three application scenarios demonstrating the versatility of our concept.

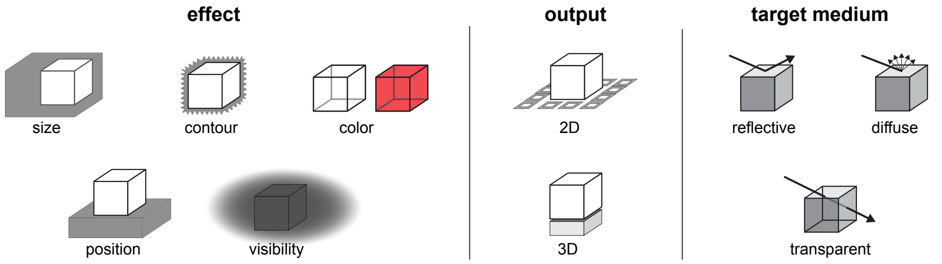


**Figure 8.2:** Design space for objects with dynamic appearance through virtual surroundings. Cubes illustrate the target objects.

## 8.2 Design space

We developed a design space for dynamically altering the appearance of objects by changing their surroundings, depicted in Figure 8.2. It is inspired by work on non-traditional displays [134], shape-changing interfaces [131, 148], and taxonomies on visual illusions (e. g. [30, 51]).

### Requirements

We created the design space with technologies such as (ubiquitous) display surfaces in mind. For any augmentation, the position and orientation of the object to be modified relative to the display needs to be known. Depending on the desired effect, we also need coarse or precise information on the object's properties, such as its color (transmission and/or reflectance) or shape (contours of full geometry). Most, but not all, effects require perspective rendering from the viewpoint of the user, which requires tracking the head or eye positions. We place objects depicted throughout this paper onto a display. For simulating display surfaces that are included in furniture, we display a wooden surface texture in addition to the actual effects. We envision that in the future, displays will be ubiquitous, however currently our concept can only be achieved by turning a display into an actual desk. By including display surfaces into everyday furniture, objects that are surrounded by a display can easily be augmented using our approach. We argue that the design space generalizes beyond technologies used in this paper, e. g. see-through augmented reality. For all these technologies, also the environment *surrounding* an object can be visually altered, which is the main requirement of our concept.

**Dimensions**

The design space consists of three dimensions, which are *effect*, *output*, and *target medium* (Figure 8.2). Each effect has a *target*, which is the object it alters. The perception of the target is driven by the effect, which can be achieved with one or more types of output. The target medium influences which output and effects can be produced.

### 8.2.1 Effect

**Size**

The perceived size of an object is altered by virtually extending the target, i. e. displaying a larger virtual object with edges that are aligned to the target. This allows enlarging objects, as displayed in Figure 8.3. However, it does not allow decreasing the perceived volume of a target. In contrast, for methods like projection mapping a reduction in size is easily achievable (given a suitable projection target). For our approach, this effect works best for displaying 3D contents around an object, and also works for targets unsuitable for projection.

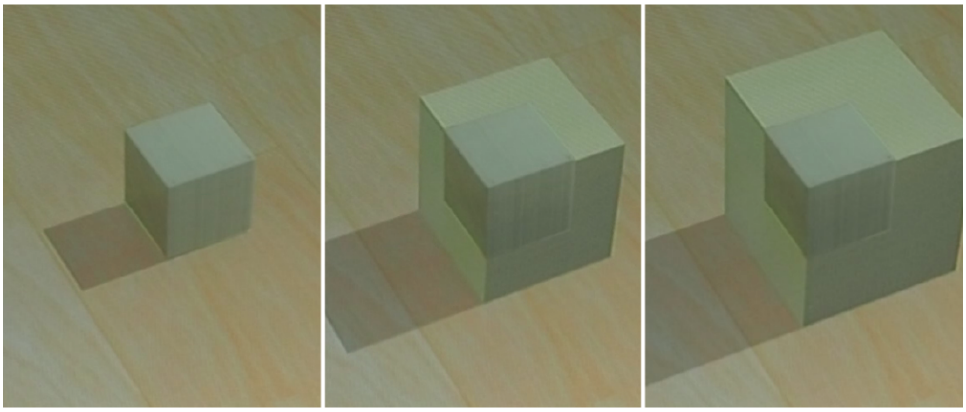

**Figure 8.3:** Effect of *size*. The cube is virtually extended to increase its size. Only the small cube (*left*) is an actual physical object, ground texture, shadows and extensions (*center* and *right*) are on-screen renderings.

### Position

Our method alters the perceived position of a target, specifically it allows ”raising” the target by adding virtual objects such as platforms underneath (see Figure 8.4). By rendering virtual ramps, the illusion of a tilted object can be induced (shown in Figure 8.1, *center*). The magnitude of this tilting effect, however, is relatively small from our experience due to other cues such as stereoscopic vision. Displaying shadows for physical and virtual objects adds to the realism of this effect. Another method to induce changes in perceived position besides using solid virtual objects (e. g. the platform) is to add virtual shadows which make an object appear levitating on the surface (related to [130, 179]). This effect of perceived levitation, however, not only requires the addition of virtual shadows, but also a motion of the displayed ground texture dependent on the observer’s current position. The ground texture is moved in concert with the motion of the observer to induce the illusion that the target is floating on the ground.

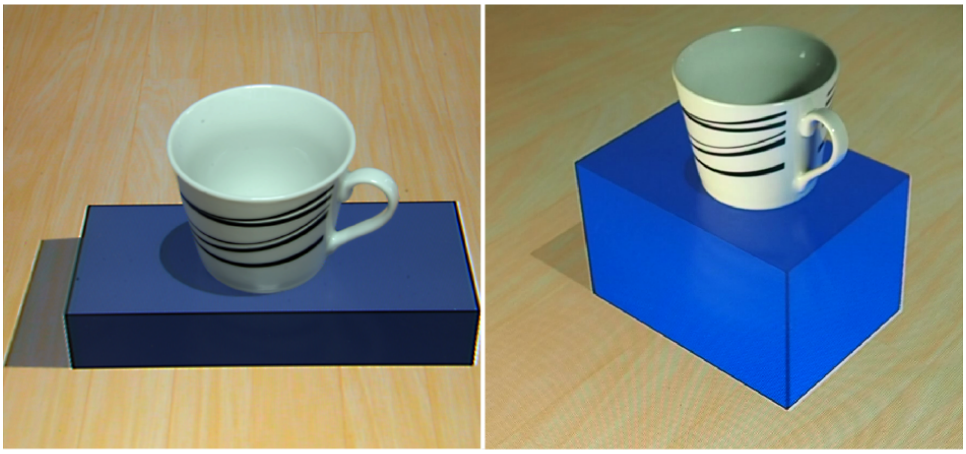

**Figure 8.4:** Effect of *position*. The cup is raised by placing it on a platform.

### Contour

Similar to effects of size, the contour of an object can be altered, as shown in Figure 8.5. In the case that the target contains holes like the cup in Figure 8.5, those can be shrunk or fully closed. Similarly, the outline of the target can be modified, for example for notifications (e. g. the contour of a phone becomes ”spiky”).

This again highlights the complementary nature of our approach compared to projection mapping. With projection mapping, holes can be rendered if the background is known (i. e. video see-through, cf. [108]), however holes cannot be closed due to the lack of display surface. Our method allows to close holes in a surface, however it cannot render additional holes into the target.

**Figure 8.5:** Effect of *contour*. The cup is augmented with a virtual contour. The hole of the handle is shrunk (*left*) and the contour is altered (*right*).

### Color

Modifying the color of an object can be an effective way for communicating information or status. The means of modification to change the color depend on the target medium. Transparent objects, as shown in Figure 8.1 (*right*), can be altered by displaying differently colored areas *behind* objects. The notion of "behind" depends on the viewing position and needs to be adapted according to the user's head position. The color of non-transparent surfaces is altered by displaying colored areas around the target, depicted in Figure 8.8. This allows for exploiting light coming from the display surface, effectively shining colored light onto objects. This effect is increased with the surface reflectance of the target object. The process of finding the corresponding area on which color has to be displayed for covering the whole target can be compared to ray tracing. Since the geometry of the target is know, rays can be traced from the observer's position to the target and follow their reflection. The area covered by the rays bouncing from the surface of the target corresponds to the minimum area that needs to be colored to change the color of the target.

### Visibility

By extending the uniformly colored areas of a target, its visual saliency can be altered. As shown in Figure 8.6, the outline of the phone is less visible by displaying a black area around it. The surface of the target is extended, drawing attention away from the actual target. This is useful in situations where users don't want objects to be immediately visible, e. g. drawing attention away from the phone when briefly leaving their desk. For targets with higher frequency textures, the textures can also be extended. We also call this effect of altering the surroundings to hide objects the *inverse camouflage effect*. In contrast to classical camouflage techniques where the color of a target is altered, we alter the surrounding space to visually hide the target.

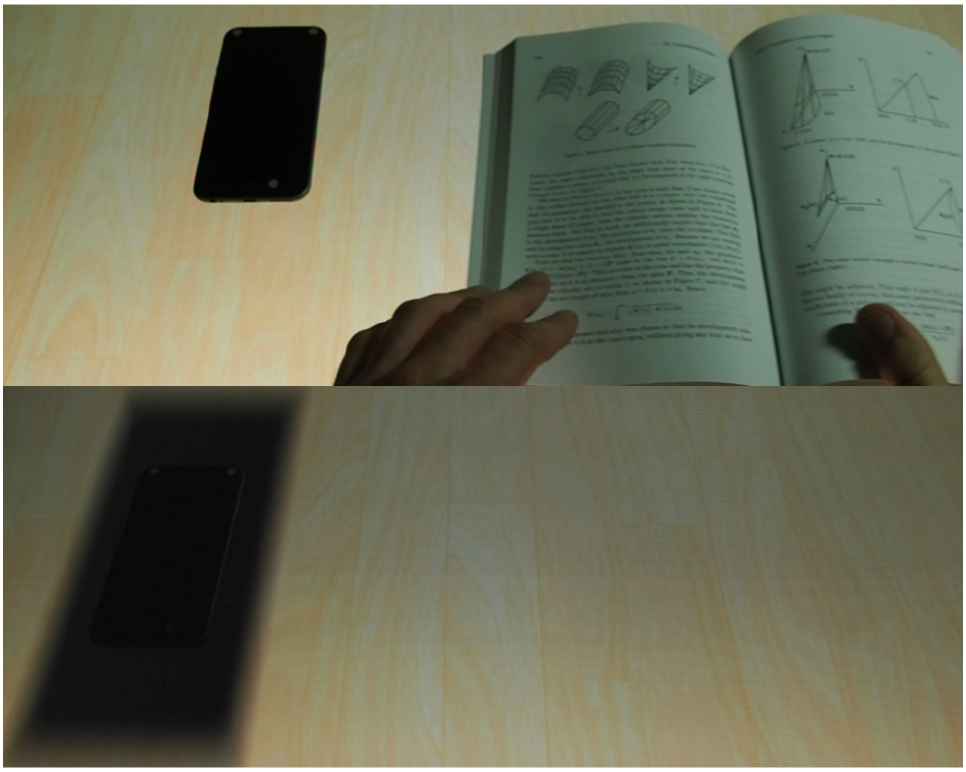

**Figure 8.6:** Effects of *visibility*. The target (phone, *top left*) is hidden by displaying matching color in its surroundings (*bottom*).

### 8.2.2 Output

The dimension *output* refers to the type of rendering that is employed for displaying effects, which can be either in 2D or 3D. While all renderings are obviously 2D (i. e. on a display), this dimension refers to the perception of virtual contents by the observer. 2D output, as used in classical illusions, can yield changes in color and visibility (Figure 8.8 and 8.6, respectively). For observers, these effects are flat 2D stimuli, without any notion of 3D. This means that the vanishing point is solely dependent on the observer's location. Since no perspective-corrected rendering is needed, 2D effects have the benefit that only the target's approximate position and size have to be known, not its exact geometry or the user's position.

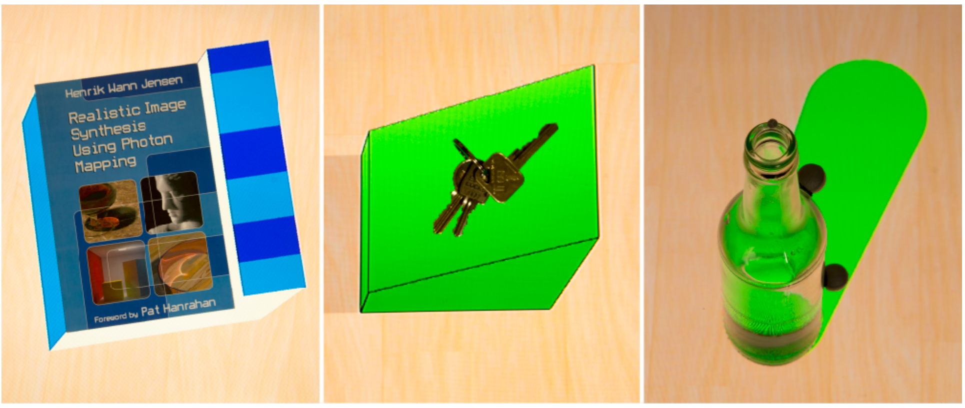


**Figure 8.7:** Top view (off-axis) of effects of size, position and color. All graphics, including the wooden surface texture, around the targets (book, keys, bottle) are displayed on a 42 inch horizontal display. Graphics are rendered perspective-corrected to appear 3D.

3D output (off-axis view shown in Figure 8.7) is required to correctly align physical and virtual objects when creating effects such as changes in contour, size or position. Thus, those effects are created with respect to the observer's location as well as with knowledge of the object's position and geometry.

3D output aims at being indistinguishable from targets, which is challenging with current displays due constraints in color and resolution (our current display only has a resolution of about 38 pixels per inch). We discuss technical requirements for high quality illusions in the implementation section.

### 8.2.3 Target medium

We categorize three different target media for our approach. The medium governs which effects can be achieved as well as their effectiveness. Targets with *diffuse* surface are suitable for effects that do not rely on reflection and transparency, such as changes in size, contour or position. Effects of color (shown in Figure 8.8 with a reflective target), are possible, however their effect is weaker than for other media when employed on a target with diffuse surface. This is because effects of color rely on directed reflection, which is weak for diffuse objects.

Targets with *reflective* surface properties such as the green cup in Figure 8.8 (*bottom*) are suitable for changes in size, position or contour, and also suitable for effects of color that are based on reflection.

**Figure 8.8:** Effect of *color* with an reflective opaque surface. The reflection on the surface in combination with the displayed color alter the target's color.

Lastly, targets with *transparent* surface are especially suitable for effects of color. Altering the color of transparent surfaces is highly challenging for techniques that rely on direct projection. Our approach results in the perception of a dynamically colored target, as shown in Figure 8.1 (*bottom*), although it really is the surface *behind* the objects that is modified.

## 8.3 Implementation

Our system uses a passive-marker optical tracking system (OptiTrack), and a control application written in C++ with openFrameworks[1] for rendering. Graphics are displayed on a horizontally-placed 42 inch LCD screen (Philips 42PF7621D, resolution $1366 \times 768$ pixels, 38 pixels per inch). We chose this display for its low parallax, i. e. there is little space between objects placed on the display and the virtual content.

Our system tracks the display surface, the user's head, and objects placed on the display. For perspective-corrected rendering we match the virtual camera's position to the tracked head position. The view frustum is modified so that the corner points of the near and far plane align with the corners of the tracked display, which is commonly referred to as fish-tank augmented reality (cf. [119, 164]), used for example for 3D tabletop displays [60, 129] or 3D see-through displays [67].

The virtual objects surrounding the real-world objects were designed to match the shape of the target objects. Coarse 3D shapes were created programmatically or 3D modeled as proxies for the physical objects (e. g. a simple box for the wallet in Figure 8.10). This is necessary since effects of size, contour and color (for transparent objects) require knowledge of a target's geometry. Colors were manually adjusted to decrease the perceived difference between the target and the augmentation. Objects placed on the display are recognized based on their markers, configured as predefined marker sets (rigid bodies) in the OptiTrack control software.

### Limitations of virtual extensions

For effects of size, target objects are visually extended with virtual content that is aligned to the target. This, however, only works if the area of the target that should be extended is facing away from the observer. Consider an observer's viewing direction $\vec{v} \in \mathbb{R}^3$ that is pointing towards a face on the surface, whose normal is denoted by $\vec{n} \in \mathbb{R}^3$. We then calculate the angle $\alpha$ between the two vectors $\vec{v}$ and $\vec{n}$. The extension can only be display correctly if $|\alpha| < 90°$, i. e. the virtual content is displayed without being occluded by the target, as depicted in Figure 8.9.

[1] http://openframeworks.cc/

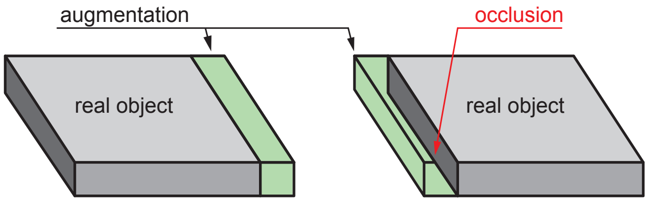


**Figure 8.9:** Extending a target only works if the normal of the extended side faces away from the observer (*left*). If the angle formed by the normal of the extended face and the viewing direction is larger than 90°, the target occludes the virtual content (*right*).

If the angle is larger than 90°, the augmentation is occluded by the target, effectively hindering correctly displaying the virtual extension, illustrated in Figure 8.9, *right*. In our implementation, if $a$ is approaching 90°, the extension is set to zero length, effectively disabling the illusion of extended size for the particular face. We chose this method to not break the illusion of extension.

### Technical considerations

Our proof-of-concept implementation does not provide the visual fidelity to make virtual content and real-world objects indistinguishable. In the following, we discuss the most important aspects which systems would need to provide to allow for such high quality visual effects.

### Tracking the user's eye position for displaying 3D contents

Accurately displaying 3D contents requires providing two different images for the user, one for each eye. Therefore, not only head tracking but eye-position tracking is needed. This aspect is especially important when aligning virtual 3D content with real-world objects when viewed from close proximity (distance $< 1.5$ m). When only a single image is provided, vergence discrepancies may arise, i. e. the virtual and real-world object are only correctly aligned for one eye. Autostereoscopic displays (e. g. [42, 139]) overcome these challenges, for example by tracking user's eyes with a retroreflective camera and providing two different images with an active shutter [139]. This approach can be extended to multiple users with a multiview parallax display (e. g. [4]).

#### Recognizing objects to allow for augmentation

To enable augmentation, objects that are situated in a user's environment have to be detected and, if needed, recognized. Detecting objects through segmentation has been demonstrated using commodity depth cameras, e. g. by Karpathy et al. [89] or Valentin et al. [182]. The resulting data can serve as input for recognition systems, e. g. model-based approaches such as by Mian et al. [120]. Besides camera-based sensing techniques, it is possible to include sensors directly into objects (e. g. using RFID [121], or optical markers like in our prototype) and combine this with position tracking. All these approaches require a priori knowledge of the objects that should be augmented (i. e. their shape, color, texture, illumination).

#### Matching color & material of virtual and real-world objects

Accurately determining the texture and other material properties such as transparency of physical objects can be challenging. While a camera is sufficient for simple color tracking, more complex surface properties such as reflectance of transparency are not captured. Tracking the reflectance and transparency of objects can be performed e. g. using active LED-based approaches ([12, 55, 156]). For these approaches, objects are illuminated with LEDs with different wavelengths for gathering their optical properties. While those systems achieve high accuracy, specialized hardware is required. Alternatively, camera-based machine learning approaches can be used (e. g. using SVM classifiers [160]). Those provide a good balance between hardware requirements and fidelity of the results. Due to changes in ambient illumination, a closed-loop approach for capturing and displaying colors (e. g. [54]) is needed for all approaches.

## 8.4 Applications

We created prototypical implementations for three scenarios in which multiple objects are augmented to demonstrate our approach.

### Ambient displays

We created three different ambient display applications. First, our system changes the perceived color of a transparent water bottle from green to yellow to red dependent on the time elapsed since the user drank last (see Figure 8.1, *bottom*). Secondly, we augment a pack of medicine with a virtual platform. The platform raises slowly to convey an elevated position to remind users of taking their medication. Finally, we display a virtual platform underneath a set of keys (as in Figure 8.1, *top right*). When the user leaves the desk, one side of the platform is lowered, thus the platform becomes a ramp. This virtual ramp should remind users to not forget their keys. All these perceptual changes are achieved *without actually modifying the object* and aim at unobtrusively capturing the user's attention.

### Notifications

We use virtual contents for delivering notifications to users. We augment a wallet (shown in Figure 8.10) for helping users keep track of their bank transactions, inspired by the Proverbial Wallet by Kestner et al. [90]. The wallet changes its contour every time money is withdrawn from the account, e. g. from monthly subscriptions or transactions on a shared credit card. The shape, size and speed of the dynamic contour correlate with the amount spent, i. e. the higher the amount, the larger and faster the notification gets.

### Privacy

Users can hide objects in their workplace by virtually blending them into the surrounding space. Our system detects objects based on their markers placed in the environment and modifies the color of the surroundings. This way, objects draw less visual attention since their color or contour becomes less salient. As an example, a phone placed on a desk (see Figure 8.6) is less easily visible from a distance. We exploit the fact that larger, uni-color areas (i. e. the target and the area around it) potentially attract less attention than individual objects with sharp contours and distinct colors (cf. Treisman et al. [176]).

**Figure 8.10:** The wallet changes its contour when money is debited from the user's bank account.

## 8.5 Preliminary experimentation

In order to gather preliminary insights if users are able to perceive our proposed effects, we performed a small scale experiment. We placed a 3D printed cube (side length 40 mm, known to participants) on a virtual platform (see Figure 8.11) and randomly varied the height of the platform between 10 mm and 110 mm. Eight participants (3 female, all staff from local university) were asked to estimate the height of the virtual platform in 10 trials each.

The data of these 80 trials is well approximated by a linear model, showing a mean estimation error of 14.5% (SD=11.5%), i. e. 11.3 mm ($SD = 10.4$ mm). A simple linear regression was calculated to predict participants' estimate based on the actual height of the virtual platform. A significant regression equation was found ($F_{1,14} = 486.585, p < .001$), with an $R^2$ of .862. Participants' estimated height is equal to $3.388 + 1.113$ (actual height), measured in millimeters. This shows that participants perceived the desired effect of an elevated physical object and that the desired effect size strongly correlates with its perception. While none of our participants were 'fooled' by the effects due to the low fidelity of current implementation, they were able to perceive the desired effect.

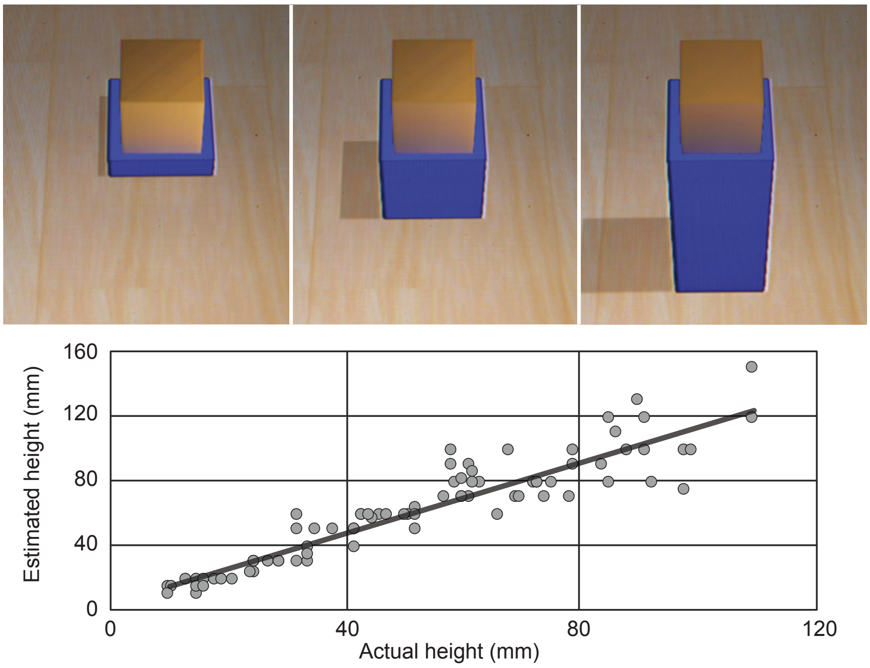


**Figure 8.11:** *Top:* Participants were asked to estimate the randomly varied height of the virtual platform underneath the white 3D printed cube (here from left to right: 10 mm, 40 mm, 100 mm). *Bottom:* Data collected in our preliminary experiment with fitted linear model.

## 8.6 Discussion

Altering existing objects and devices through virtual illusions, or more generally, through displaying contents around them has a number of benefits compared to existing approaches, summarized in Table 8.6.1. Direct augmentation techniques such as projection mapping allow changing the appearance of objects, however only if the objects exhibit surface properties that are suitable for projection. Many works on augmented reality focused on the *addition* of virtual contents (e. g. rendering virtual characters) to real scenes (e. g. [14, 67]). Our work focused on changing existing objects, e. g. changing their color or perceived size. Shape-changing interfaces require including the desired functionality during manufacturing. Modification of existing devices is not always feasible. Additionally, changes in perceived size such as a mechanical increase in volume can be challenging to achieve. Transparency-controlled interfaces (Chapter 5)

| | Our approach | Projection mapping | SCI | TCI |
|---|---|---|---|---|
| **Size** | larger | (smaller) | ✓ | ✓ |
| **Position** | ✓ | × | ✓ | × |
| **Contour** | ✓ | (✓) | ✓ | × |
| **Color** | ✓ | (✓) | × | × |
| **Visibility** | ✓ | (✓) | × | ✓ |

**Table 8.6.1:** Comparison of different techniques: our approach, projection mapping, shape-changing interfaces (SCI), and transparency-controlled interfaces (TCI, [110]). Effects created with projection mapping are marked in brackets since they require a suitable projection surface.

allow changing volume and visibility, however they also have to be manufactured with these functionalities in mind. Furthermore, they do not allow changing the color of targets and their ability to induce the illusion of changed position is limited. However, our work yields limitations in terms of visual changes that are possible to achieve as well as providing feedback other than of visual nature.

#### Tactile limitations

Our proposed concept does not provide any dynamic tactile sensation. While it is still suitable for interactions where no tactile feedback is needed (e. g. ambient displays, peripheral interaction), it lacks the tactile qualities of shape-changing interfaces. However, it has benefits in terms of rendering possibilities.

#### Visual fidelity and stereo vision

Making virtual contents indistinguishable from physical objects is challenging, as discussed above on a technical level. Systems aiming at providing high quality object augmentation have to meet requirements in terms of *display quality* (resolution and capability to accurately reproduce materials), *stereo rendering* as well as *system responsiveness*. Not meeting one of those requirements will most likely break perceptual effects. We aimed at providing an initial design exploration of our proposed approach. Our current prototype does not aim at 'fooling' users into believing that the illusions are real, but at conveying information through visual effects.

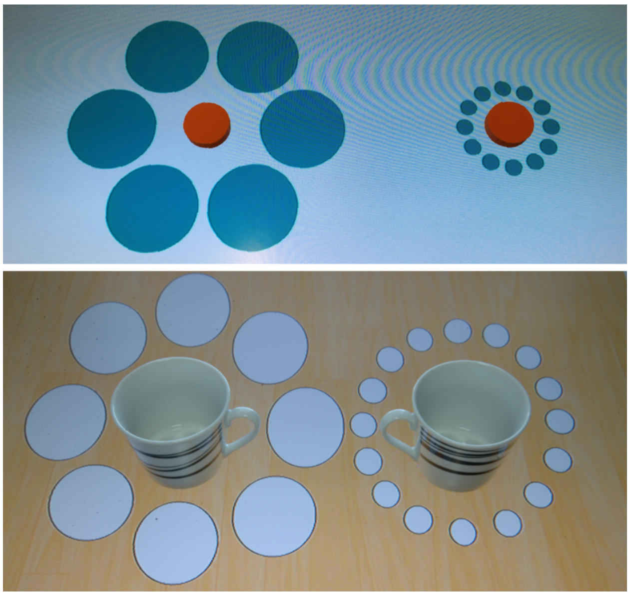

**Figure 8.12:** The Ebbinghaus illusion for a simple 3D target (puck, *top*) and a geometrically more complex target (cup, *bottom*). Although both pucks have exactly the same size (36 mm), the right one appears to be larger. The illusion was not present for more complex objects like the cup.

## 8.7 Classical visual illusions in 3D

We see our approach related to work on visual illusions, which typically have a target object that is perceptually altered by surrounding objects. However, even typically strong visual illusions such as the Ebbinghaus illusion (see Figure 8.12) only have an effect magnitude of about 20% (cf. [128]). Other illusions such as the Watercolor or the Delbouef illusion yield even more subtle effects (cf. [30]). Illusions such as Ponzo or Hering work well in 2D since they rely on a mismatch of 2D and 3D visual sensation ("errors in perception", [51]). When applying these illusions to complex 3D stimuli, however, effects such as increase in size did not emerge in our initial tests. As shown in Figure 8.12 (*top*), the Ebbinghaus illusion when used on a simple 3D printed cylinder triggers the illusion that the right target appears larger (both have the same size). When used on geometrically more complex objects like the cup in Figure 8.12 (*bottom*), the illusion is not present. The same is true e. g. for contrast illusions like the Chubb illusion [31] (or *simultaneous contrast effect*) as shown in Figure 8.13.

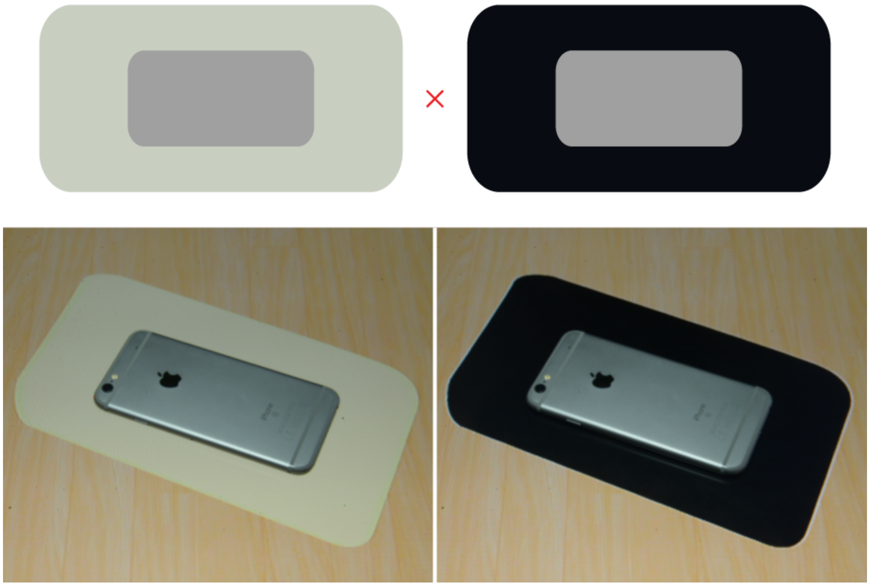

**Figure 8.13:** Comparison of the classical Chubb illusion with 2D primitives (*top*) and real-world objects (*bottom*). Objects in the center have the same color. For the classical illusion, the left object in the center appears darker, especially when focusing on the red cross in the center. This effect is decreased for real-world objects.

These observations lead us to believe that knowledge from classical illusions in 2D is not easily transferable to complex 3D stimuli. Therefore, we resorted to a more complex environment using techniques from computer graphics (such as perspective-corrected 3D rendering) for creating the *illusion* of dynamic appearance. We emphasize that this does *not* necessarily mean that classical illusions do not work in 3D. We note, however, that it might require very careful design of (new) illusions and parameter tuning before effects emerge. We believe more research is needed to understand the effect of classical illusions (that are effective in 'flat' environments) in the context of genuinely 3D stimuli.

# 9

# Connecting optically dynamic interfaces to the virtual world

## 9.1 Dynamic interfaces and the virtual world

Optically dynamic interfaces conceptually focus on *enriching, extending and augmenting* real-world physical interfaces, and connecting them to the virtual world. This is done via building them from optically dynamic material or augmenting them directly or indirectly. While we argue that this increases the space of possible appearances and interactions, possibly beyond physically plausible phenomena (e. g. rapid changes in transparency), some manipulations are highly challenging in the (constrained) physical world. In virtual reality, objects can undergo essentially arbitrary changes, up to the point where the whole world that surrounds a user changes.

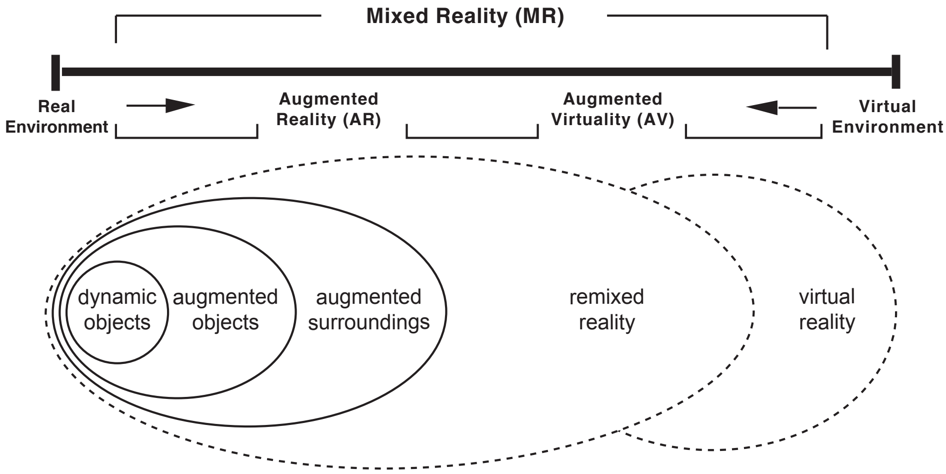


**Figure 9.1:** The model of optically dynamic interfaces with its extension to the virtual world, called *Remixed Reality*, set in perspective with the Mixed Reality continuum by Milgram and Kishino [123]. The dashed line indicates that while Remixed reality shares similar capabilities to other types of optically dynamic interfaces, it can perform those *changes purely virtually.*

However, as emphasized before, this poses one key challenge: *nothing the users sees is real*. Adding realistic haptic feedback (e. g. through real-world props [66]) can lead to more compelling experiences, but is still technologically challenging.

In this chapter we explore extending dynamic interfaces into the virtual world. We do so to explore possibilities of novel dynamic interfaces and experiences without having to build them, essentially *prototyping a completely dynamic world*. We call this conceptual extension *Remixed Reality* and illustrate its connection to optically dynamic interfaces and the classical taxonomy of Milgram and Kishino [123] in Figure 9.1. While optically dynamic interfaces and their three levels cover a large part of the mixed reality spectrum, they do not allow for fully immersive experiences and applications. Furthermore, augmented objects and augmented surroundings rely on precise alignment of virtual and physical objects, the ability to extract background information from behind the physical object, and a display technology that is capable of overlaying the physical object with the extracted background information in a way that the original object is no longer visible. Virtual reality, on the other side of the spectrum, requires technological interventions to also feature haptic experiences or the allow connections to the physical world.


**Figure 9.2:** We present a novel take on mixed reality, called Remixed Reality. The user (*left*) wears a VR headset and sees a live reconstruction of the environment (*second left*) captured through multiple external depth cameras. This allows leveraging the benefits of virtual environments (e. g. easy modification of the environment by removing geometry, *second right*) and dynamic viewpoints while allowing users to see the actual physical world (e. g. teleportation, *right*).

In contrast to augmented reality, the environment the user sees does not necessarily mirror the real world. Even a VR user viewing a virtual representation of the room they physically occupy would typically find that any changes in the real room would not be reflected in the virtual room since the geometry they see is not live. *Remixed Reality* aims at combining the benefits of augmented reality and virtual reality in a single approach. We created a mixed reality environment in which users see a live view of their environment. In contrast to other mixed reality approaches such as the work by Miyaki and Rekimoto [125], users do not see the image from a head-mounted front-facing camera. Instead, we equip the environment with multiple RGB-D cameras (Microsoft Kinect v2). These depth cameras are positioned to obtain maximum visual coverage of the room. Users wear an immersive head-mounted display (HTC Vive) and see a reconstructed live view of the environment (see Figure 9.2). We transform the user's viewpoint in this reconstructed live representation of the room to match their physical position in the room. This means that physical objects and their virtual representations are aligned. Since the data is live, any changes in the physical environment (e. g. furniture moving, or people entering the room) are visible in real time to users.

Remixed Reality can be seen as a different type of see-through augmented reality that allows users to experience their physical environment, while it also enables manipulations that are usually only available in purely virtual environments. For example, users can easily "erase" geometry in the room (Figure 9.2, *second right*), enabling applications such as simulated room remodeling. Since the cameras cover the environment from multiple angles, users can erase parts of the room and see the environment behind these parts. This approach also allows recoloring, moving or copying objects. Since all changes are applied continuously on the live geometry, any moved or copied object continues to update dynamically in real time after the operation is invoked. Furthermore, since the environment is a 3D model, changing the user's view is as simple as modifying the graphics camera eye point, enabling virtual motion (in VR referred to as teleportation). For example, a user can move to another part of the room virtually to inspect the environment from a different perspective without changing their physical location (Figure 9.2, right). Remixed Reality also allows making temporal changes. Users can halt time (i. e. suspend receiving live data) and move freely in their static reconstructed environment. They also can record events (e. g. meetings) and play them back at any desired speed. Since the data from the environment changes at every frame, we separate the operations we wish to apply to dynamically changing geometry from the geometry itself. Instead of performing modifications on a polygonal reconstruction or point cloud, we use an underlying voxel grid that records interactions on the geometry, e. g. applying transformations like translation and scaling, and changing visibility. These interactions are applied continuously on live geometry, a crucial feature in our system. Remixed Reality extends the space of possibilities in mixed reality and provides a way to seamlessly bridge the virtual and physical world. It gives users full control over how they perceive their physical environment. This allows them to make nearly arbitrary modifications, including those that would not be plausible in the real world. Many of those changes, e. g. temporal and spatial changes, are not yet possible with optically dynamic interfaces. Remixed Reality allows us to prototype manipulations that we wish to implement while still giving users the ability to experience the haptic sensation of the real world. Therefore, while all changes are of visual nature, they go beyond what is possible with conventional augmented reality and optically dynamic interfaces.

## 9.2 Design space of Remixed Reality

We explore the concept of Remixed Reality by identifying a design space of possible modifications, illustrated in Figure 9.3. The dimensions are defined by four types of modifications: spatial, appearance, temporal and viewpoint modifications. The design space is inspired by work on shape-changing interfaces [148], mixed reality [111, 115, 123], and other dynamic interfaces [110, 134].

**Requirements**

The main requirements for Remixed Reality are a means to capture a live 3D model of the environment, and a display that allows users to see the (modified) physical world and virtual objects at the same time. For some modifications, it is important that the capture extends beyond what the user may physically see from their physical viewpoint and moment in time. This helps to overcome occlusion and see the physical world behind users, for example. We currently use an immersive head-mounted display (HMD) and multiple depth cameras in a room. The number of cameras required depends on the environment to be covered and the location of potential points of interest in the room. We currently use eight Kinects for a room of approximately $4 \times 5$ meters. While the proposed taxonomy was created with our current implementation in mind, we believe it also applies to other technological contexts. Future implementations might use see-through displays that are capable of completely overlaying the physical world with content. Data could be acquired with future hardware such as a few 360° RGB-(D) cameras.

### 9.2.1 Spatial modification

Remixed Reality is appropriate for real-time modification of objects and the environment since everything the users see is reconstructed geometry. The ability to modify every aspect of the geometry, such as shape, size, or position, with available techniques from computer graphics, is key to making typically challenging modifications (e. g. completely removing an object from user's view) simple. All changes are illustrated in Figure 9.3 (*top left*) and can be performed manually by users or automatically by the system.

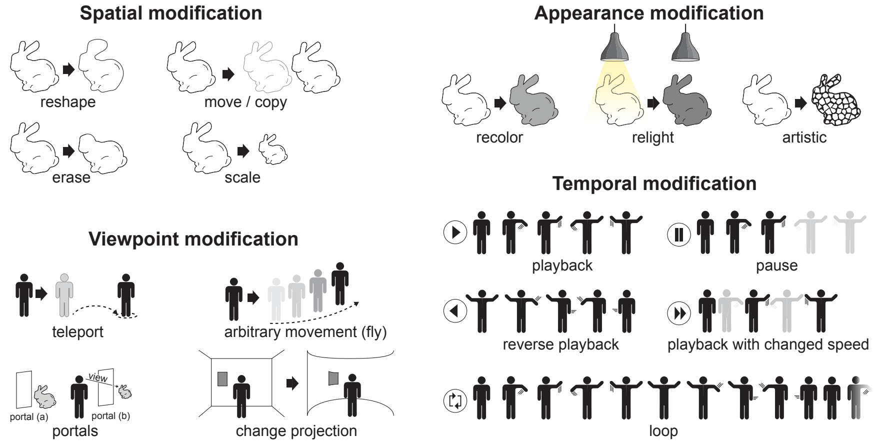


**Figure 9.3:** Design space of possible manipulations for Remixed Reality with four dimensions for modifications: spatial, appearance, viewpoint and temporal.

**Reshape**

Geometry including parts of physical objects and the environment can be reshaped by moving parts of the geometry or performing geometric operations such as erosion, expansion or morphing, for example. This enables applications such as live-sculpting of physical objects, e. g. for industrial design.

**Erase**

Objects in the environment can be partly or completely removed from the user's view. This can be valuable for privacy considerations (e. g. only specific users can see an object in a room) or room re-modeling. This type of diminished reality usually requires inferring the environment behind the object. Our approach resolves challenges of occlusion since the representation of the physical world and the virtual geometry share the same coordinate system.

**Move & Copy**

Objects can be moved by users or the system, e. g. to unblock the view of the environment behind it, without physically moving the object. Note that motion is not constrained by physics, meaning that it is possible to move a chair so that it appears to be levitating. Copying an object essentially creates a virtual clone of the physical geometry. Since the representation of the physical object and its copy consist of the exact same geometry and texture, it becomes difficult for users to distinguish the copy from the original. Since both operations are performed on live geometry, any changes to the source object are propagated to the moved or copied object.

#### Scale

Objects can be scaled arbitrarily, enabling users to remodel their environment at will. Scaling can be used to create world-in-miniature representations [165], or visualizations employing magic lens effects [44, 184], for example.

### 9.2.2 Appearance modification

Appearance modifications are distinct from spatial modifications in that they only alter the look (e. g. color, texture) of the environment, instead of its geometry.

#### Recolor

Recoloring objects can be performed either by replacing an object's color or blending, e. g. for applications such as ambient displays or interior design. As for most other changes, recoloring can be performed on the whole object, parts of it, or the environment. In our implementation, a target is specified by the user's selection.

#### Relight

Changing the light changes the color of the whole environment or parts of it indirectly. This enables for example dimming the room light to focus on a task, or gaming scenarios. Note that, as with any other changes, the light only changes in the virtual rendering of the room, and not in the real room.

#### Other artistic changes

Remixed Reality enables changing the appearance in various ways, e. g. for artistic purposes or gaming. This can be achieved by applying image filters such as edge enhancement, desaturation or blurring. Figure 9.4 shows examples of such changes, with edges being enhanced using a Sobel filter and colors being saturated and turned black and white, respectively.

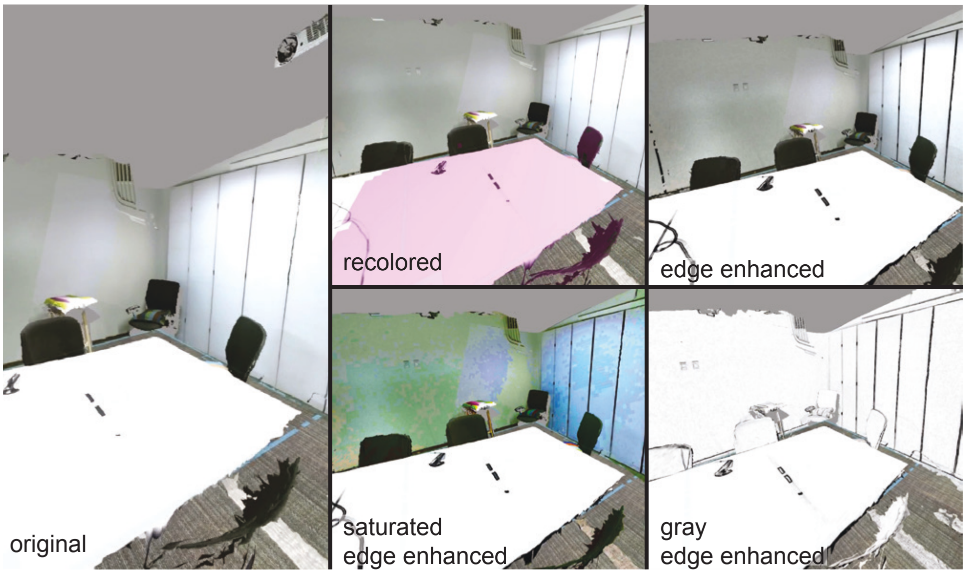


**Figure 9.4:** Appearance modifications applied to the live view.

### 9.2.3 Temporal modification

Remixed Reality also enables changing the temporal properties of an environment. Any data captured through the depth cameras can be paused, recorded and played back at any desired speed.

#### Pause time

Time is ”paused” by no longer updating the geometry that is presented to users. This does not mean, however, that they are presented with a still image of the scene, but rather a static 3D reconstruction of the environment. Users can move freely in the scene and observe the room at the instant in time when the pause command was triggered. Figure 9.5 shows an example where a user paused time during jumping. Even though users see a static scene, time in the physical world resumes normally. Once the resume-command is triggered, users effectively make a ”time jump”. Events that occurred during a pause can be recorded and played back later.

**Figure 9.5:** User ”paused” time during jumping, then walked around the table to inspect the scene.

**Playback and loop**

The representation of the physical world that is presented to users can be recorded and played back at various speeds. This allows slowly replaying short events (e. g. the jump in Figure 9.5), or fast playback of longer events such as meetings, for example. Remixed Reality also allows users to repeatedly play back events (i. e. looping).

### 9.2.4 Altering viewpoint

An advantage of Remixed Reality over conventional mixed reality approaches is the ability to dynamically alter the user's viewpoint. Since Remixed Reality features large visual coverage of the environment, users are free to choose their viewing location.

### Teleportation and arbitrary movement

Remixed Reality is naturally suited to camera and viewpoint control techniques used in games and virtual reality. We refer to discrete viewpoint changes (i. e. specifying a target position in the room and going there without physical motion) as teleportation, adopted from current virtual reality games. Furthermore, users are not constrained to walk on the ground but can also fly or walk on walls, for example. This is achieved by simply moving the virtual camera away from user's actual tracked position in the room.

### Portals

Users are not only able to see the environment from their current physical or virtual perspective, but to equip the environment with "portals." Conceptually and in our current implementation, a single portal consists of a display area and a virtual camera. If no other portal is present, this single portal acts like a mirror. By linking multiple portals (e. g. portal A shows the view of portal B and vice versa), users can view the room from multiple perspectives. Figure 9.6 shows two portals, one placed beside the user, the other behind them. By looking at the portal to their left side the user can see themselves from behind.

Portals can be viewpoint-dependent or viewpoint-independent and have been used for navigation in virtual reality (e. g. [29, 48]). Viewpoint-dependent portals essentially behave like a fishtank VR display [164], meaning that the perspective (i. e. the virtual camera's rotation) changes according to a user's perspective. Viewpoint-independent portals are similar to conventional displays with a static virtual camera. Portals can also be used to constantly monitor regions-of-interest as in the work by Lin et al. [105].

In our current implementation, those portals can not only be used for viewing but also as ways to transport objects and users, inspired by the game Portal by Valve Corporation [35]. By moving an object into a portal (i. e. triggering a collision), the object is moved "through" the portal to the location of the linked portal. Users can perform the same action themselves (i. e. walk through a portal) as an alternative to teleportation.

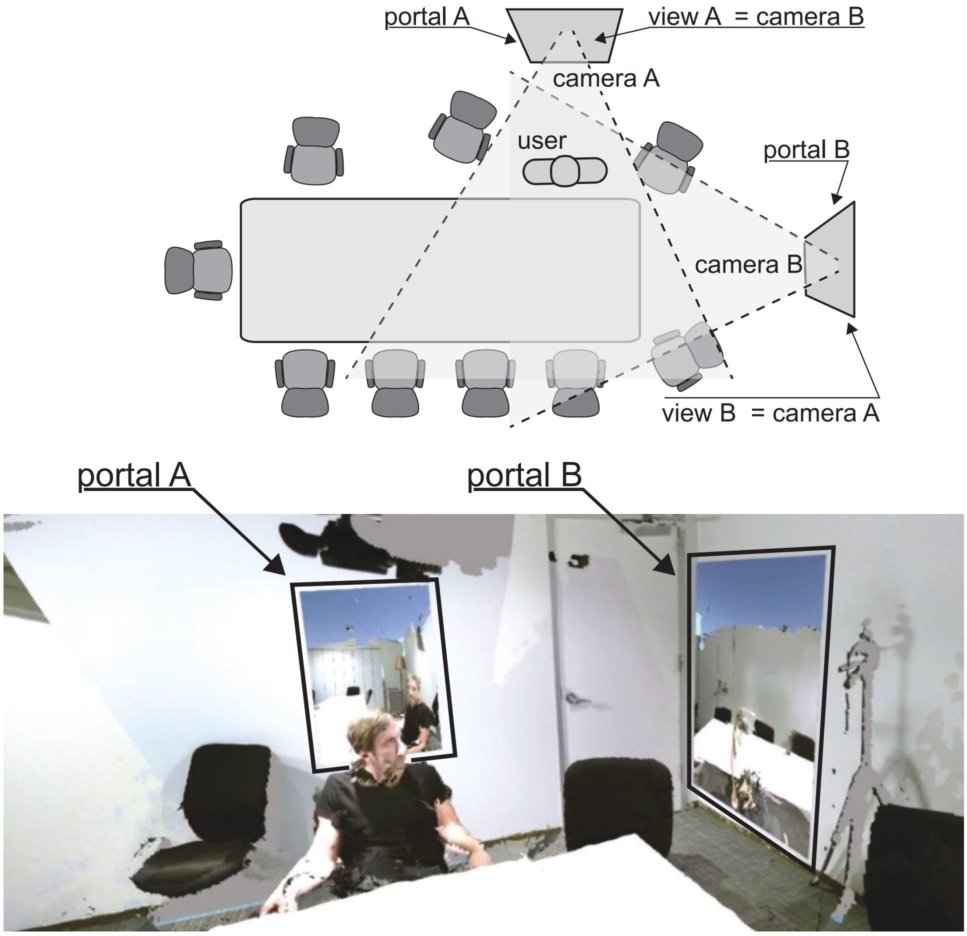


**Figure 9.6:** Portal A shows the view of portal B (side of user), while portal B shows the view of portal A (back of user).

### Changing perspectives

For certain applications, keeping a constant overview of the environment can be valuable. By default, the user's view employs a conventional perspective graphics camera. We can, however, use other camera models that feature, for example, fish-eye lenses or arbitrary distortions to give users the ability to easily enlarge their field of view, as shown in Figure 9.7. This also enables modifications of the view-port such as real-world zooming, or x-ray (i. e. viewing through objects), or taking on the perspective of another person (or even object) in the environment.

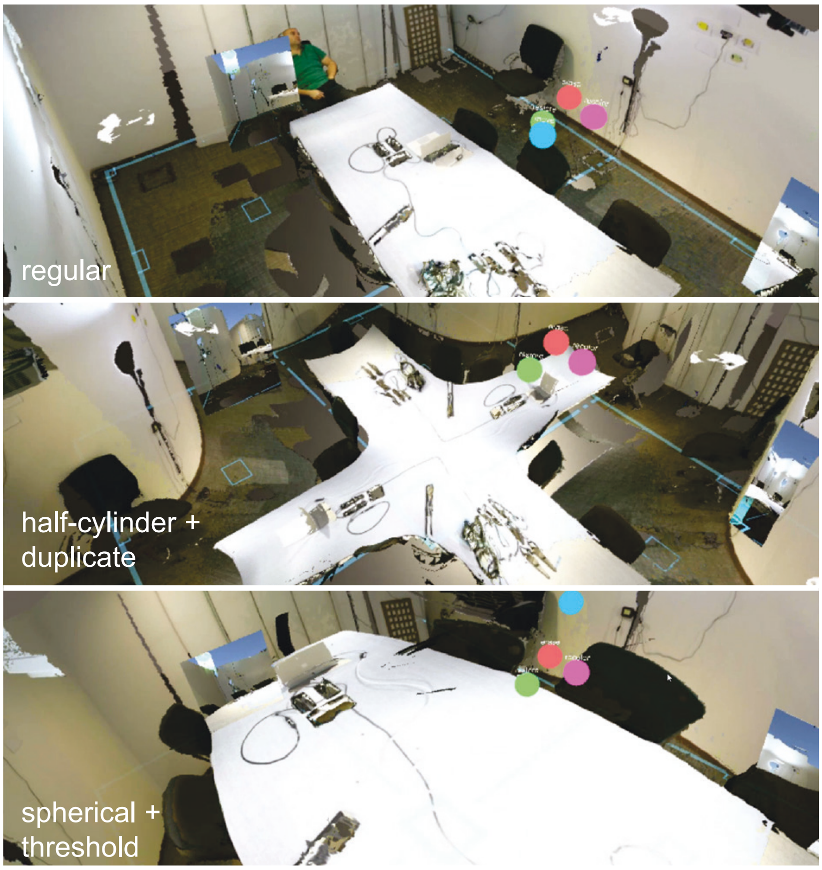


**Figure 9.7:** Users can be presented with different perspectives on the current environment. *Top* shows a perspective with a conventional camera model. Center shows a projection to a half-cylinder and all contents duplicated. *Bottom* shows a projection onto a sphere for contents within a certain distance.

### 9.2.5 Virtual content in a reconstructed reality

Mixed reality approaches allow the environment to be augmented and extended with virtual objects. Typically, virtual content overlays real objects to convey information. Remixed Reality additionally allows erasing a physical object and optionally replacing it with a virtual one. One example is demonstrated throughout many of the example images in this chapter: to fill holes in the reconstructed floor caused by an incomplete model of the room, we replaced the floor with a virtual ground plane with a texture similar to the physical floor. Since it is visually similar to the actual floor, it blends into the room nicely.

Conventional mixed reality applications aim to render physical and virtual objects with the same high visual quality, and therefore must properly handle lighting, occlusion and anchoring of objects in the environment. With our current depth camera-based implementation, object occlusion and anchoring are resolved automatically since the room's physical environment is effectively physical and virtual at the same time. Note that occlusion (i. e. areas without visual coverage) still occurs for regions that are not covered by any of the external cameras. This could be resolved by using more cameras or cameras with a wider field of view. Physical and virtual objects share the same coordinate space, making anchoring easy. Correct inter- and intra-object occlusion is achieved by presenting users with the 3D reconstruction of the physical world. Considering visual quality, our approach also requires equalizing visual quality of physical and virtual geometry. Due to the rather low resolution of the depth cameras and calibration noise, the overall visual quality of the environment suffers. Consequently, the visual quality of the virtual objects should probably be decreased as well so that they do not attract undue attention. This could be achieved by analyzing geometric noise and colors retrieved from the depth cameras and forming a model from those parameters. By applying this model as a filter to virtual objects, the gap in visual quality would be decreased. We plan to investigate this in the future.

### 9.2.6 Safety considerations

Erasing or displacing physical objects, pausing time, and teleporting lead to potential safety challenges. Users lose the ability to judge whether physical objects are in their way, which can lead to bumping into those objects. This could be resolved by more advanced "chaperoning" techniques, such as rendering erased geometry as opaque or semi-transparent when users are in their proximity. Furthermore, applying techniques such as redirected walking [150] or inverse haptic retargeting [6] (e. g. warping the environment in a way that users do not touch anything) could resolve this challenge. We plan to investigate this interesting part of research in the future, especially since this not only applies to our approach but many virtual reality scenarios.

## 9.3 Implementation

Our current implementation relies on two main components, the data acquisition and reconstruction of the environment, and displaying and interacting with the reconstructed environment. Note that the user is not equipped with a head-mounted camera (only a head-mounted display), since their position might be independent of their viewpoint. The external cameras in our approach allow easy handling of occlusion and geometry alignment. An overview of the system is illustrated in Figure 9.8.

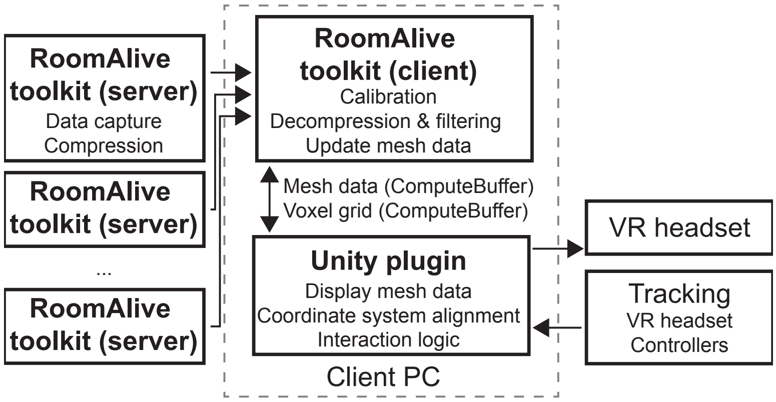


**Figure 9.8:** Overview of the Remixed Reality system.

Data acquisition and reconstruction is performed using the RoomAlive Toolkit [193]. In the example shown in this chapter, the room is equipped with eight Kinect v2 cameras. Each of the cameras is connected to a compact server PC (Intel NUC) which receives data from the depth cameras and sends the compressed depth and color data to the RoomAlive client, running on a commodity gaming PC. All PCs run Windows 10. The client performs decompression and reconstruction. Display and interaction is implemented in Unity 5.6 and SteamVR, which also runs on the client computer. A virtual reality headset (HTC Vive) and its trackers are connected to the client and used for displaying the geometry. Data exchange between the RoomAlive client application and Unity is implemented as a custom native Unity plugin. All shaders are implemented in DirectX 11.

### 9.3.1 Calibration and data acquisition

The depth cameras are calibrated using the RoomAlive calibration procedure, which was built for dynamic projection mapping. Six projectors were installed in the environment to display Gray codes onto the physical surfaces in the room. Using projectors could be omitted by performing the calibration directly on the received data using the iterative closest point (ICP) algorithm [17] for alignment, for example. The depth and color data is compressed (lossless depth compression based on [192], JPEG color compression) and sent over Ethernet to a computer running the RoomAlive client application. The eight Kinects produce approximately 400 Mbit/s of data, which is well within the limit of a regular network switch (typically 1 Gbit/s). This also means this approach would scale to more than eight cameras if the required network bandwidth is available. Data can be processed and rendered at the client in real-time at a maximum of 100 frames per second, which is higher than the frame rate of the Kinect cameras (typically 30 frames per second). The data is decompressed on the RoomAlive client and reconstructed using the camera parameters retrieved in the calibration procedure. The client also performs basic filtering and noise rejection and shares the data with Unity through a custom plugin.

### 9.3.2 Displaying live depth data

A custom Unity plugin handles data exchange between the RoomAlive client and Unity. In our current implementation, we create a Unity scene with multiple meshes for exchanging the data. The data of eight Kinects results in approximately 2 million vertices per frame. The meshes are represented as multiple compute buffers, shared between Unity and the RoomAlive client. This means that, while the data is updated constantly, Unity does not require its own copy of the data. For display, we use a HTC Vive VR headset with the base stations for tracking mounted in the room. The coordinate systems of the mesh and the VR headset are manually aligned once (adjusting an affine transformation in Unity) and the resulting matrix is stored in Unity. This allows the physical world and its geometric representation to be tightly aligned.

Note that there is no drift over time since the coordinate systems of the Kinects and the VR headset do not change over time. Calibration accuracy was in the range of few centimeters, which from our experience was sufficiently accurate for users to grab items (e. g. a remote control on the table). Accuracy could be further improved by using a semi-automatic multiple-point calibration procedure, for example. The user is represented as camera in Unity. Since the tracked space of the headset and the mesh data are aligned, user motion in the physical space is forwarded as camera motion, allowing the user to move freely in the tracked space. For teleportation, we move the camera that represents the user.

### 9.3.3 Modifying live depth data for interaction

Modifying a mesh, e. g. erasing or moving parts, typically involves modifying its underlying vertices. In our case, however, the mesh that represents the environment changes at a rate of 30 frames per second. Since there is no correspondence between the meshes over time, any change performed on the mesh at one frame would not carry over to the next frame. Retaining changes between frames would require aligning the raw vertices (e. g. using ICP or optical flow) and determining which vertices have been changed due to interaction. In our case of approximately 2 million vertices, this operation would be very computationally expensive, potentially prohibiting real-time interaction. Therefore, we rely on an underlying volumetric representation of interaction data that is maintained simultaneously. The polygonal reconstruction is used solely for display purposes. We implemented a 3D voxel grid comprised of 1.35 million voxels (150 width $\times$ 150 length $\times$ 60 height), resulting in a voxel size of approximately 3 cm in our room (4 m $\times$ 5 m $\times$ 2.5 m), shown in Figure 9.9. The voxel grid is initialized as a compute buffer shared between Unity and the native plugin and passed to all shaders. The voxel grid and the geometry from the depth cameras share the same coordinate system, meaning that each vertex has a corresponding voxel (however a voxel typically contains multiple vertices). This means that although vertices between frames do not have any correspondence, they do fall in the same voxel if there was no motion between frames. This is key to enable modifying the geometry and preserving changes over time.

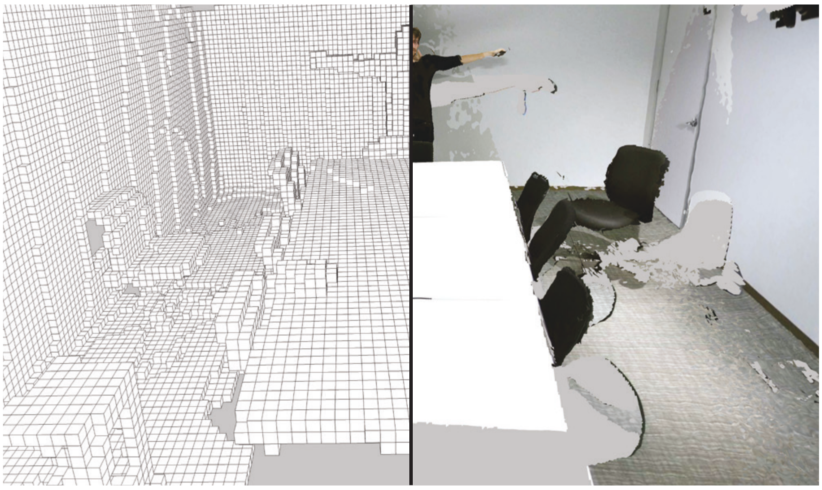

**Figure 9.9:** We store interaction data (e. g. changes in visibility, transformation) in an underlying voxel grid (*left*) that is aligned with the geometry retrieved from the depth cameras (*right*). Users do not see the voxel grid.

When the user selects geometry, this selection is performed on the voxel grid rather than the geometry. Each voxel holds flags for interaction data, i. e. if it has been selected, colored, hidden, moved or copied. In the geometry shader that generates the mesh, each vertex queries its corresponding voxel. Based on the flags that are specified in the voxel, the shader performs the corresponding interaction, e. g. translating a vertex if the move-flag is active. This process is illustrated in a 2D example in Figure 9.10. Note that in this approach all actions are performed on a per-vertex basis, requiring the examination of only a small fraction of the voxel grid. This means that the resolution of the voxel grid can be increased up to the limit of the graphics card's memory, since every interaction is basically a look-up in the voxel grid per vertex. Thus, the computational complexity only scales proportional to the number of vertices, not the number of voxels. While we currently use 1.35 million voxels for 2 million vertices, we tested this implementation with up to 50 million voxels on the same number of vertices. This only increased the time to upload the voxel grid onto the GPU (only performed once at the start of the software). Keeping voxel size constant, this means that this approach can cover a room significantly larger than the one used in our experiments.

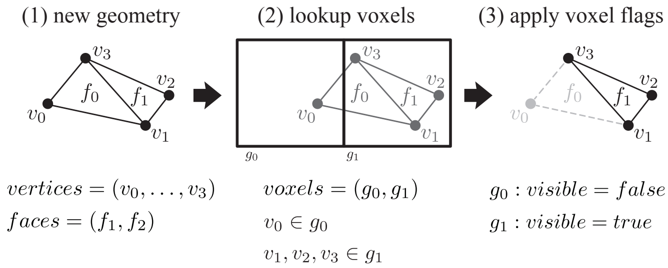


**Figure 9.10:** Workflow for modifying live mesh data. For all new geometry (*left*), we perform a lookup operation to find the voxels at the same position (*center*). Each voxel contains specific flags regarding properties such as visibility or translation. Those flags are then applied to the geometry (*right*). Vertices that fall within the left voxel are erased, i. e. discarded by the geometry shader.

### 9.3.4 User interaction with volumetric handles

Users interact with the environment using spherical selection handles. The interaction is related to the ”go-go” technique [141], and allows performing contactless selections on physical surfaces. The handles manipulate the voxel grid that is applied to the environment’s geometry (the ”erase” handle sets the corresponding voxels to ”hidden”, for example). This manipulation is performed on the GPU by passing the voxel grid to the shaders that render the handles.

## 9.4 Applications

We utilize the proposed capabilities in a variety of applications taking advantage of spatial, temporal, viewpoint and appearance modification.

#### Interior design

We created an application that allows users to remodel their room. First, users erase any objects in the room they want to replace. They can then trigger a second purely virtual room that is equipped with furniture, and teleport themselves into this room. By moving furniture into the physical space, they can see what the remodeled room would look like, for example as in Figure 9.11. By placing virtual seating furniture where physical chairs are, they can also sit ”on virtual objects”, taking advantage of the physical objects as haptic proxies (cf. [66]).

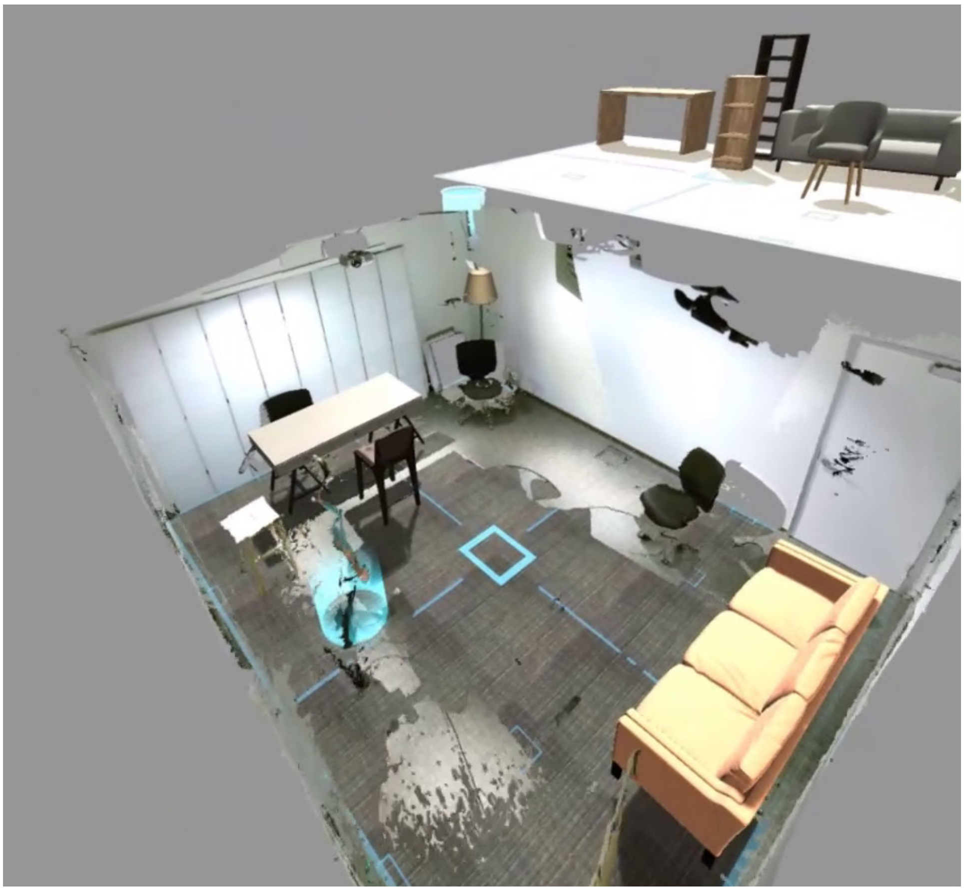

**Figure 9.11:** The user has erased the table and chairs in the room and replaced them with virtual furniture (couch, table, chair). Note the virtual second floor with furniture on the top right. Users can teleport there to retrieve virtual object.

### Relive a meeting

Because the complete physical environment in a room can be recorded, users can relive past events such as meetings. When ”watching” the meeting while in the room, users can choose to either sit at an empty chair or take the perspective of another attendee. The perspective can be changed any time, for example through teleportation. Since the data is recorded, users can also choose to ”attend” the meeting at increased speed or jump back in time to important moments.

#### Gaming

Remixed Reality can be used for gaming purposes. To demonstrate this, we created a simple shooting game. Players can dim the light in the room (virtually), and use a blaster to erase geometry in the room. The geometry spawns back after a few seconds. Combining this with other changes in appearance, for example enhanced contours, and playback of pre-recorded scenes, leads to an immersive gaming experience in the physical world without players being bound to physical constraints, e. g. they can fly rather walk on the ground.

#### Investigate posture

Users can halt time at any point. This allows applications such as investigating the posture of fast movements. As an example, by pausing time at the moment a baseball player hits a ball, common errors such as collapsing the back side or dipping the shoulder can be viewed first hand and corrected. Remixed Reality allows not only capturing and viewing this moment in 2D but as a full 3D representation that users can walk around and even virtually correct a posture through sculpting.

#### Remove distraction

By incorporating Kinect body tracking, users can choose to remove the unwanted distraction caused by other people in the room. In this mode, the system removes all users by erasing the geometry sufficiently near to the position of their tracked heads. By incorporating object tracking based on geometry rather than head position, this removal could be more fine-grained.

#### Fast viewpoint switch

When performing actions on larger objects, e. g. fastening a knot around a large box, or reaching for something that is occluded by a large object, users may need to walk around the object to see what they are doing. Remixed Reality allows quickly changing the user's perspective (currently by pushing a button on the controller), which enables them to look behind or around objects without changing their physical location.

### Extension: Remixing remote reality

While the core concept of Remixed Reality focuses on co-located interaction, users can also incorporate remote spaces into their environment or interact with a remote environment. Incorporating spatially distant locations into a single environment is performed by aligning and blending the two. Furthermore, this enables interactions such as adding a smaller representation of one environment into another, similar to concepts such as world-in-a-miniature [165] or Holoportation [138], as shown in Figure 9.12. This seamless blending (remixing) of multiple environments enables applications in telepresence or remote collaboration. Multiple blended environments are largely indistinguishable in terms of visual quality and immersion since they use the same capture and rendering processing pipeline.

**Figure 9.12:** The live view of a remote user is placed as a miniature on the table.

## 9.5 Discussion

While Remixed Reality offers advantages in terms of modifications that go beyond conventional mixed reality approaches, several challenges arise.

### 9.5.1 Limitations

**Visual quality**

The visual quality of the live 3D reconstruction presented to users depends on the resolution and positioning of the external depth cameras used by the system. Higher resolution depth cameras would partly alleviate the problem of low visual quality. Furthermore, by incorporating more advanced algorithms for live filtering and reconstruction or adding 3D scans of static parts of the environments (e. g. KinectFusion [81]), the visual quality could be improved. This, however, is challenging due to the large size of the overall reconstructed mesh. Any algorithm for visual enhancement would need to run in real time, a demanding requirement for many existing techniques. We plan to enhance the overall visual quality of our system as one of the next steps. This would further allow us to investigate which degree of visual fidelity is necessary for users to be even more immersed in the system, and potentially "forget" that they are wearing a headset.

**Infrastructure and visual coverage**

Our current implementation requires the installation of multiple depth cameras in a room to obtain large visual coverage of the environment. This, however, might not be feasible for many users. We envision that our approach will also work with different (future) camera technologies, for example through the use of a small number of 360-degree depth cameras. Future research also needs to tackle challenges when capturing surfaces that are not well captured by current cameras, such as transparent surfaces. We further plan to investigate how Remixed Reality can be implemented with fewer cameras, for example in part by using a single head-mounted camera that retains a copy of obtained geometry and updates this presentation constantly. This, however, would potentially prohibit large changes in viewpoint and lead to uncaptured areas due to occlusion.

**Haptic feedback**

Remixed Reality offers users a full, correct passive haptic experience before nearby geometry has been manipulated. This effect can be degraded as users modify their environment by erasing geometry, changing their viewport, or teleporting themselves, for example. Resolving this challenge would require techniques such as haptic retargeting [6] or other haptic proxies (e. g. [66]) to restore tactile sensation even when the physical and the virtual world no longer match.

### 9.5.2 From Mediated Reality to Artificial Reality and back

Remixed Reality aims at combining the benefits of virtual and augmented reality, effectively blending physical and virtual spaces. The ability to easily modify many aspects of an environment is enabled by presenting users with a reconstruction of the environment. Benefits of mixed reality such as viewing the actual physical reality with real haptic sensations are preserved by using a live view of the environment. Remixed Reality enables a broad range of applications and interaction that have previously only been achievable in purely virtual environments, such as teleportation or temporal changes. We believe our work opens interesting extensions and directions for future work.

In his seminal paper on Mediated Reality, Mann [115] introduced the idea of a wearable device that can capture and control incoming light rays as well as produce light rays, essentially giving users full control over what they see and perceive. We see Remixed Reality as *one instantiation of this idea*, since it affords full control over what users see by converting their environment into live reconstructed geometry. Remixed Reality allows users to seamlessly transition between different levels of virtuality, from seeing the pure physical world without augmentation, via partly altered reality to fully artificial reality. While Mann focused mostly on visual perception, we envision future versions of Remixed Reality to be capable of altering users' perception of physics (e. g. everything appears to be happening in a zero-gravity environment) and perception of time (e. g. everything moves slowly). Once users can no longer distinguish what is virtual and what is real, we can seek the most comfortable or appropriate level of mediation.

We envision such a system to be suitable for a wide range of psychophysical experiments (e. g. investigate out-of-body perception) and to allow us to create and extend the experiences Mann envisioned, from showing users what a monochromatic environment looks like, what the world looks like from the perspective of another person, what living in an upside-down world feels like, or extending users capabilities with ability to zoom and see through walls. While Mann experimented with some of those effects, we see Remixed Reality as a first step of gaining full control of what we see and being able to choose the level of augmentation we desire.

# 10

# Conclusion

With current technologies, the virtual and the physical world are largely distinct and users operate in either one world *or* the other. Users interact with digital content through displays on their computers or smartphones, for example, or interact with the *unaugmented* physical world. The main objective of this dissertation was to *explore novel ways to bridge the two worlds through means of integrated and external augmentation*. I argue that by closing the gap between the virtual and the physical world, novel interactions and experiences that are beneficial for users can be created. Specifically, removing this separation can be advantageous and allow for a closer integration of technology into users' daily lives. This consequently will make technology less obtrusive and more usable.

To achieve this, I introduced the concept of *optically dynamic interfaces*. I created multiple instantiations of optically dynamic interfaces to explore how they can be created, how they benefit users, and which opportunities they offer.

I defined optically dynamic interfaces as real-world physical objects that change the way they are perceived by users. This is achieved through integrated capabilities or external augmentation. While all manipulations are performed purely optically, the interfaces can be built from physically static or dynamic objects.

To categorize the different examples, I proposed a *model of optically dynamic interfaces*, consisting of three levels: dynamic objects, augmented objects, and augmented surroundings. The implementations this dissertation provides for each level highlight the flexibility of optically dynamic interfaces. I transformed a conventional WIMP interface, a display, into an optically dynamic interface by building it from smart materials. This transparency-controlled see-through display exhibits the beneficial properties of transparent displays in terms of awareness while retaining the benefits of conventional opaque displays in terms privacy. Building on this approach, I created physical interfaces that are able to change their perceived shape by manufacturing them from material with controllable transparency and applying origami-like folding techniques. These transparency-controlled physical interfaces change their shape on demand and provide an unobtrusive and integrated way to connect objects with the virtual world. Both examples are instantiations of dynamic objects.

I furthermore provide an initial evaluation of the usability of dynamic objects in the context of transparent displays. The study revealed that in a dual-task scenario, users exhibited greater performance with a transparent and a horizontal display when compared to conventional opaque displays in several orientations. The findings hint at the beneficial nature of optically dynamic interfaces in terms of awareness and usability. However, broader experimentation is needed to generalize these findings to different types of the proposed interfaces.

To explore the concept of augmented objects, I combined physically dynamic interfaces with spatial augmented reality. This combination gave rise to new ways of increasing the perceived resolution and speed of shape-changing interfaces. These two properties are inherently limited in shape-changing interfaces due to physical constraints and usability issues. Since not all objects are suitable for projection mapping, I presented an implementation of virtual surroundings.

Everyday objects were placed on a display and augmented *indirectly*. By displaying perspective-corrected 3D graphics *around* objects, I was able to change the perceived size, shape, color and visibility of objects.

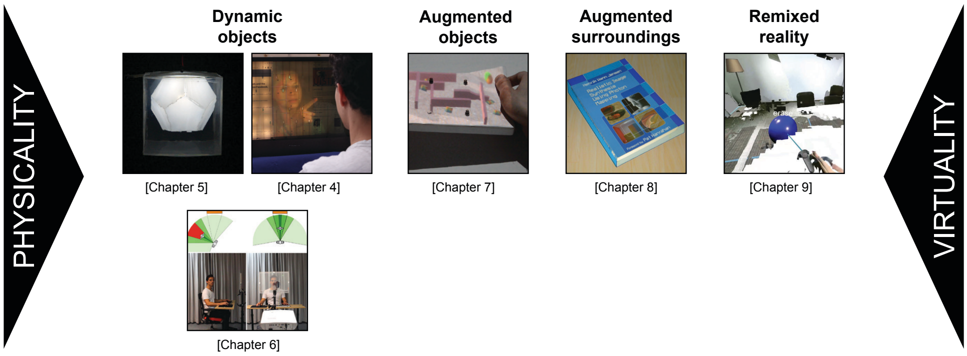


**Figure 10.1:** Summary of the presented interfaces and implementations with respect to physicality and virtuality.

While all these implementations differ in technology, manufacturing and consequently their abilities, they share a common goal: *change how humans perceive real-world objects*. This goal was approached from two different perspectives: *physicality* and *virtuality*, illustrated in Figure 10.1. I approached dynamic objects, augmented objects and augmented surroundings from the perspective of physicality. This means altering and augmenting physical objects to incorporate digital properties and capabilities. I changed their physical appearance through different technological approaches. This object-centric approach, however, is oftentimes constrained by the physical limits of the real world. These constraints are not present in the purely virtual world. To explore the perspective of *virtuality*, I created a new type of mixed reality system, termed *Remixed Reality*. While using virtual reality equipment (e. g. an immersive head-mounted display), this approach brings back properties of the physical world into this virtual environment. The environment is captured live through external cameras and users viewport is transformed to match their position in the room. Remixed Reality allows performing all changes that are possible in the virtual world, e. g. temporal, spatial or appearance changes, while users can still touch and feel the real world. This offers an experience in which everything the users see can be changed, thus their is no separation between the virtual and the physical world. The implementation allows exploring questions such as which objects should be made optically dynamic, i. e. which manipulations should be implemented in the real world, as well as the degree of augmentation we are comfortable with.

I see the instantiations of optically dynamic interfaces as a first step to explore this large space of possible applications, technologies and combinations of the different levels.

**Dynamic objects**

The resolution of the transparency-controlled see-through displays, specifically the number of transparency-controlled patches, is currently limited. I plan to increase the resolution of the transparency-control matrix to support finer control. By applying insights gained from building the transparency-controlled physical interfaces, specifically the fabrication of individual patches with engraved electrical routing, I will be able to increase the resolution of the transparency-controlled patches significantly. Tracs in the implementation presented in this dissertation had a patch size of $5 \times 5$ cm. By engraving routes directly into transparency-control layer I was able to decrease the patch size to 2 cm side length. By incorporating industrial processes, I am confident that this size could be further decreased. This will allow me to explore if it is desirable to decrease the patch size to the size of a pixel, for example, or if this fine-grained control is not needed. Additionally, working with components of higher transparency, e. g. transparent OLEDs, will give me the opportunity to evolve this setup. Furthermore, applying my concept of transparency-control to more device class (e. g. smartphones, tablets) is an interesting direction for future research.

In the context of transparency-controlled physical interfaces I plan to explore ways to create more complex 3D objects that include more layers. Currently, layering is limited by the transparency of the switchable diffuser (around 80% transparency). This makes objects with more than six to eight layers infeasible. By incorporating other smart materials, e. g. materials that can transition between transmissive, opaque and reflective, I hope to further expand the space of possible manipulations. By combining transparency-controlled physical interfaces with physically dynamic object, I hope to combine the best of both worlds. This approach is similar to my work on shape-changing interfaces and spatial augmented reality, presented in Chapter 7.

### Augmented objects

Creating augmented objects as a combination of shape-changing interfaces and spatial augmented reality allows for a more accurate and realistic representation of content. The way the device looks is closer to what the designer of the device desires. It enables natural depth cues and leaves users uninstrumented, which I believe is important. I believe that future shape-changing interfaces should incorporate pixel displays to increase the space of possible interactions and usages. In the future, I plan to explore augmented objects with technologies other than projection mapping, for example optical see-through displays as in the work by Hamasaki et al. [59]. This, however, is challenging because of a mismatch in depth cues. Virtual content shown with a head-mounted display is always at the same depth layer, in contrast to physical objects with essentially arbitrary placement. Furthermore, exploring ways to make the virtual augmentation and the physical object indistinguishable, e. g. by means of radiometric compensation as in the work by Wetzstein and Bimber [188], will be an interesting for future research.

### Augmented surroundings

Augmented surroundings present a promising approach to visually enrich objects that exhibit challenging surface properties. The approach offers advantages for some manipulations, e. g. increasing the size of objects. This is not possible with conventional spatial augmented reality approaches. For future research, it will be interesting to find ways to equalize the displayed content and the appearance of physical objects. This will, like for augmented objects, allow making virtual content and physical target objects indistinguishable. Expanding this line of research in terms of applications and potential scenarios will allow me to explore more augmentations and find the limits of this approach. Replacing the currently used conventional display with more advanced technologies such as autostereoscopic displays will be a first starting point into this direction.

### Combination

In this dissertation, I was mostly concerned with introducing the individual types of optically dynamic interfaces. There exists, however, a large space of possible combinations of the individual types. As an example, a dynamic object that changes its perceived size through transparency-control might as well be additionally augmented through projection mapping to change its color. This would allow for richer interactions and enlarge the space of possible applications. The same holds true for the combination of dynamic objects and augmented surroundings, or the combination of augmented objects and augmented surroundings. Many of the limitations of the individual types can be overcome by including other types. As an example, the ability to increase and decrease the perceived size of an object is mutually exclusive for augmented objects and augmented surroundings. Combining the two would allow to do both, increase *and* decrease the perceived size of an object. Exploring these combinations of different types of optically dynamic interfaces is a promising direction for future research which I wish to pursue.

### Evaluation

I started to evaluate optically dynamic interfaces in terms of usability in the context of dynamic objects, specifically transparent displays. I analyzed task performance and distraction and were able to identify benefits and limitations of this technology. The results gave valuable insights into users' ability of performing dual tasks with varying display configurations. However, more diverse stimuli need to be tested, potentially with real actors such as by Reetz et al. [151], instead of abstract stimuli. Furthermore, the impact of stimuli with different visual behavior (e.g. looming stimuli) needs to be included in future studies. Similarly, I probed the usability of augmented objects in an initial small-scale user experience study for the combination of shape-changing interfaces and spatial augmented reality. User feedback was positive and I was able to gather first interesting insights into user behavior and perception. For future research, I wish to expand this evaluation to all levels of optically dynamic interfaces. I believe dynamic and augmented objects are promising technologies to deliver unobtrusive notifications to users, potentially benefitting the field of peripheral interaction. Virtual surroundings offer an interesting testbed for experiments in terms of user perception. Answering the question at which point users are no longer able to distinguish virtual and physical objects is among the experiments I wish to perform. While I presented a set of enabling technologies and a framework to categorize them, I believe there exists a large space of controlled experiments and longitudinal studies that should be performed. These experiments will yield insights into the current technologies and directions for future research on optically dynamic interfaces.

# Bibliography


[1] Tomas Akenine-Möller. Fast 3d triangle-box overlap testing. In *ACM SIGGRAPH 2005 Courses*, SIGGRAPH '05, New York, NY, USA, 2005. ACM. doi: 10.1145/1198555.1198747. URL http://doi.acm.org/10.1145/1198555.1198747.

[2] Marc Alexa and Wojciech Matusik. Reliefs as images. *ACM Trans. Graph.*, 29(4):60:1–60:7, July 2010. ISSN 0730-0301. URL http://doi.acm.org/10.1145/1778765.1778797.

[3] Jason Alexander, Andrés Lucero, and Sriram Subramanian. Tilt displays: Designing display surfaces with multi-axis tilting and actuation. In *Proc. MobileHCI '12*, pages 161–170, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1105-2. URL http://doi.acm.org/10.1145/2371574.2371600.

[4] Thomas Annen, Wojciech Matusik, Hanspeter Pfister, Hans-Peter Seidel, and Matthias Zwicker. Distributed rendering for multiview parallax displays. In *Proceedings SPIE 6055, Stereoscopic Displays and Virtual Reality Systems XIII*, volume 6055, 2006. URL http://dx.doi.org/10.1117/12.650171.

[5] AntiVJ. Enghien, 2009. Retrieved September 17, 2016 from http://antivj.com/enghien/.

[6] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. Haptic retargeting: Dynamic repurposing of passive haptics for enhanced virtual reality experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1968–1979, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858226. URL http://doi.acm.org/10.1145/2858036.2858226.

[7] Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoper. Virtual Environ.*, 6(4):355–385, August 1997. ISSN 1054-7460. doi: 10.1162/pres.1997.6.4.355. URL http://dx.doi.org/10.1162/pres.1997.6.4.355.

[8] Deepak Bandyopadhyay, Ramesh Raskar, and Henry Fuchs. Dynamic shader lamps: Painting on movable objects. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, ISAR '01, Washington, DC, USA, 2001. IEEE Computer Society. ISBN 0-7695-1375-1. URL http://dl.acm.org/citation.cfm?id=582828.881324.

[9] Lyn Bartram, Colin Ware, and Tom Calvert. Moticons: Detection, distraction and task. *Int. J. Hum.-Comput. Stud.*, 58(5):515–545, May 2003. ISSN 1071-5819. doi: 10.1016/S1071-5819(03)00021-1. URL http://dx.doi.org/10.1016/S1071-5819(03)00021-1.

[10] Olivier Bau, Ivan Poupyrev, Ali Israr, and Chris Harrison. Teslatouch: Electrovibration for touch surfaces. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 283–292, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0271-5. doi: 10.1145/1866029.1866074. URL http://doi.acm.org/10.1145/1866029.1866074.

[11] Patrick Baudisch and Carl Gutwin. Multiblending: Displaying overlapping windows simultaneously without the drawbacks of alpha blending. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 367–374, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8. doi: 10.1145/985692.985739. URL http://doi.acm.org/10.1145/985692.985739.

[12] Moshe Ben-Ezra, Jiaping Wang, Bennett Wilburn, Xiaoyang Li, and Le Ma. An led-only brdf measurement device. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '08, pages 1–8, June 2008. URL http://dx.doi.org/10.1109/CVPR.2008.4587766.

[13] Hrvoje Benko, Andrew D. Wilson, and Ravin Balakrishnan. Sphere: Multi-touch interactions on a spherical display. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pages 77–86, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-975-3. doi: 10.1145/1449715.1449729. URL http://doi.acm.org/10.1145/1449715.1449729.

[14] Hrvoje Benko, Ricardo Jota, and Andrew Wilson. Miragetable: Freehand interaction on a projected augmented reality tabletop. In *Proceedings of the*

*SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 199–208, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. URL http://doi.acm.org/10.1145/2207676.2207704.

[15] Hrvoje Benko, Andrew D. Wilson, and Federico Zannier. Dyadic projected spatial augmented reality. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 645–655, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. URL http://doi.acm.org/10.1145/2642918.2647402.

[16] Amit Bermano, Philipp Brüschweiler, Anselm Grundhöfer, Daisuke Iwai, Bernd Bickel, and Markus Gross. Augmenting physical avatars using projector-based illumination. *ACM Trans. Graph.*, 32(6):189:1–189:10, November 2013. ISSN 0730-0301. URL http://doi.acm.org/10.1145/2508363.2508416.

[17] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, February 1992. ISSN 0162-8828. doi: 10.1109/34.121791. URL http://dx.doi.org/10.1109/34.121791.

[18] Mark Billinghurst, Adrian Clark, and Gun Lee. A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction*, 8(2-3):73–272, 2015. ISSN 1551-3955. doi: 10.1561/1100000049. URL http://dx.doi.org/10.1561/1100000049.

[19] Oliver Bimber and Ramesh Raskar. *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A K Peters, 2005.

[20] Oliver Bimber, Franz Coriand, Alexander Kleppe, Erich Bruns, Stefanie Zollmann, and Tobias Langlotz. Superimposing pictorial artwork with projected imagery. *IEEE MultiMedia*, 12(1):16–26, January 2005. ISSN 1070-986X. URL http://dx.doi.org/10.1109/MMUL.2005.9.

[21] Jeremy Birnholtz, Lindsay Reynolds, Eli Luxenberg, Carl Gutwin, and Maryam Mustafa. Awareness beyond the desktop: Exploring attention and distraction with a projected peripheral-vision display. In *Proceedings of Graphics Interface 2010*, GI '10, pages 55–62, Toronto, Ont., Canada, Canada, 2010. Canadian Information Processing Society. ISBN 978-1-56881-712-5. URL http://dl.acm.org/citation.cfm?id=1839214.1839225.

[22] James F. Blinn. Simulation of wrinkled surfaces. In *Proceedings of the Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '78,

pages 286–292, New York, NY, USA, 1978. ACM. URL http://doi.acm.org/10.1145/800248.507101.

[23] James F. Blinn and Martin E. Newell. Texture and reflection in computer generated images. *Commun. ACM*, 19(10):542–547, October 1976. ISSN 0001-0782. URL http://doi.acm.org/10.1145/360349.360353.

[24] Harry Blum. A transformation for extracting new descriptors of shape. *Models for the Perception of Speech and Visual Form*, 1967.

[25] Leonardo Bonanni. Ambient intelligence in everyday life - living with hyper-reality. pages 130–141. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 3-540-37785-9, 978-3-540-37785-6. URL http://dl.acm.org/citation.cfm?id=2167762.2167770.

[26] Leonardo Bonanni and Hiroshi Ishii. Perfect Red: A design exploration towards the vision of Radical Atoms, 2016. Retrieved January 18, 2018 from http://vimeo.com/61141209.

[27] Leonardo Bonanni, Chia-Hsun Lee, and Ted Selker. *Cooking with the Elements: Intuitive Immersive Interfaces for Augmented Reality Environments*, pages 1022–1025. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. ISBN 978-3-540-31722-7. URL http://dx.doi.org/10.1007/11555261_95.

[28] Matthew Brehmer, Joanna McGrenere, Charlotte Tang, and Claudia Jacova. Investigating interruptions in the context of computerised cognitive testing for older adults. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2649–2658, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2208656. URL http://doi.acm.org/10.1145/2207676.2208656.

[29] Gerd Bruder, Frank Steinicke, and Klaus H. Hinrichs. Arch-explore: A natural user interface for immersive architectural walkthroughs. In *Proceedings of the 2009 IEEE Symposium on 3D User Interfaces*, 3DUI '09, pages 75–82, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-1-4244-3965-2. doi: 10.1109/3DUI.2009.4811208. URL http://dx.doi.org/10.1109/3DUI.2009.4811208.

[30] Mark A. Changizi, Andrew Hsieh, Romi Nijhawan, Ryota Kanai, and Shinsuke Shimojo. Perceiving the present and a systematization of illusions. *Cognitive Science*, 32(3):459–503, 2008. ISSN 1551-6709. URL http://dx.doi.org/10.1080/03640210802035191.

[31] Charles Chubb, George Sperling, and Joshue A. Solomon. Texture interactions determine perceived contrast. *Proceedings of the National Academy of Sciences of the United States of America*, 87(3):9631–9635, 1990.

[32] Marcelo Coelho and Pattie Maes. Shutters: A permeable surface for environmental control and communication. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, pages 13–18, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-493-5. doi: 10.1145/1517664.1517671. URL http://doi.acm.org/10.1145/1517664.1517671.

[33] Marcelo Coelho and Jamie Zigelbaum. Shape-changing interfaces. *Personal Ubiquitous Comput.*, 15(2):161–173, February 2011. ISSN 1617-4909. doi: 10.1007/s00779-010-0311-y. URL http://dx.doi.org/10.1007/s00779-010-0311-y.

[34] Marcelo Coelho, Hiroshi Ishii, and Pattie Maes. Surflex: A programmable surface for the design of tangible interfaces. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pages 3429–3434, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-012-8. doi: 10.1145/1358628.1358869. URL http://doi.acm.org/10.1145/1358628.1358869.

[35] Valve Coorporation. Portal, 2007. Retrieved January 25, 2018 from http://www.valvesoftware.com/games/portal.html.

[36] Mary Czerwinski, Greg Smith, Tim Regan, Brian Meyers, George Robertson, and Gary Starkweather. Toward characterizing the productivity benefits of very large displays. In *(2003) Interact 2003*. IOS Press, 2003. URL http://research.microsoft.com/apps/pubs/default.aspx?id=64317.

[37] Peter Dalsgaard and Kim Halskov. 3d projection on physical objects: Design insights from five real life cases. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1041–1050, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. URL http://doi.acm.org/10.1145/1978942.1979097.

[38] Peter Dalsgaard and Kim Halskov. Tangible 3d tabletops. *interactions*, 21(5):42–47, September 2014. ISSN 1072-5520. doi: 10.1145/2658859. URL http://doi.acm.org/10.1145/2658859.

[39] Maria Danninger, Roel Vertegaal, Daniel P. Siewiorek, and Aadil Mamuji. Using social geometry to manage interruptions and co-worker attention

in office environments. In *Proceedings of Graphics Interface 2005*, GI '05, pages 211–218, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. Canadian Human-Computer Communications Society. ISBN 1-56881-265-5. URL http://dl.acm.org/citation.cfm?id=1089508.1089543.

[40] Erik D Demaine and Joseph O'Rourke. A survey of folding and unfolding in computational geometry. *Combinatorial and computational geometry*, 52: 167–211, 2005.

[41] Andrew Dillon, Lisa Kleinman, Gil Ok Choi, and Randolph Bias. Visual search and reading tasks using cleartype and regular displays: Two experiments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 503–511, New York, NY, USA, 2006. ACM. ISBN 1-59593-372-7. doi: 10.1145/1124772.1124849. URL http://doi.acm.org/10.1145/1124772.1124849.

[42] Neil A. Dodgson. Autostereoscopic 3d displays. *Computer*, 38(8):31–36, Aug 2005. ISSN 0018-9162. doi: 10.1109/MC.2005.252. URL https://doi.org/10.1109/MC.2005.252.

[43] Yue Dong, Jiaping Wang, Fabio Pellacini, Xin Tong, and Baining Guo. Fabricating spatially-varying subsurface scattering. *ACM Trans. Graph.*, 29(4): 62:1–62:10, July 2010. ISSN 0730-0301. URL http://doi.acm.org/10.1145/1778765.1778799.

[44] Steven Feiner, Blair MacIntyre, Marcus Haupt, and Eliot Solomon. Windows on the world: 2d windows for 3d augmented reality. In *Proceedings of the 6th Annual ACM Symposium on User Interface Software and Technology*, UIST '93, pages 145–155, New York, NY, USA, 1993. ACM. ISBN 0-89791-628-X. doi: 10.1145/168642.168657. URL http://doi.acm.org/10.1145/168642.168657.

[45] 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, pages 417–426, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2268-3. URL http://doi.acm.org/10.1145/2501988.2502032.

[46] Clifton Forlines, Chia Shen, Daniel Wigdor, and Ravin Balakrishnan. Exploring the effects of group size and display configuration on visual search.

In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, CSCW '06, 2006. URL http://doi.acm.org/10.1145/1180875.1180878.

[47] Karmen Franinović and Luke Franzke. Luminous matter. electroluminescent paper as an active material. In *Proceedings of the 9th International Conference on Design and Semantics of Form and Movement*, DeSForM'15, pages 37–47, 2015.

[48] Sebastian Freitag, Dominik Rausch, and Torsten Kuhlen. Reorientation in virtual environments using interactive portals. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, 3DUI'14, pages 119–122, March 2014. URL http://dx.doi.org/10.1109/3DUI.2014.6798852.

[49] Antonio Gomes and Roel Vertegaal. Paperfold: Evaluating shape changes for viewport transformations in foldable thin-film display devices. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '15, pages 153–160, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3305-4. doi: 10.1145/2677199.2680572. URL http://doi.acm.org/10.1145/2677199.2680572.

[50] Antonio Gomes, Andrea Nesbitt, and Roel Vertegaal. Morephone: A study of actuated shape deformations for flexible thin-film smartphone notifications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 583–592, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470737. URL http://doi.acm.org/10.1145/2470654.2470737.

[51] Richard L. Gregory. Knowledge in perception and illusion. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 352(1358):1121–1127, 1997. ISSN 0962-8436. doi: 10.1098/rstb.1997.0095. URL http://rstb.royalsocietypublishing.org/content/352/1358/1121.

[52] David Grimes, Desney S. Tan, Scott E. Hudson, Pradeep Shenoy, and Rajesh P.N. Rao. Feasibility and pragmatics of classifying working memory load with an electroencephalograph. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 835–844, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-011-1. doi: 10.1145/1357054.1357187. URL http://doi.acm.org/10.1145/1357054.1357187.

[53] Markus Gross, Stephan Würmlin, Martin Naef, Edouard Lamboray, Christian Spagno, Andreas Kunz, Esther Koller-Meier, Tomas Svoboda, Luc

Van Gool, Silke Lang, Kai Strehlke, Andrew Vande Moere, and Oliver Staadt. Blue-c: A spatially immersive display and 3d video portal for telepresence. *ACM Trans. Graph.*, 22(3):819–827, July 2003. ISSN 0730-0301. doi: 10.1145/882262.882350. URL http://doi.acm.org/10.1145/882262.882350.

[54] M.D. Grossberg, H. Peri, S.K. Nayar, and P.N. Belhumeur. Making one object look like another: controlling appearance using a projector-camera system. In *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '04, 2004. URL http://dx.doi.org/10.1109/CVPR.2004.1315067.

[55] Jinwei Gu and Chao Liu. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral brdf. In *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '12, pages 797–804, June 2012. URL http://dx.doi.org/10.1109/CVPR.2012.6247751.

[56] Carl Gutwin and Saul Greenberg. A descriptive framework of workspace awareness for real-time groupware. *Comput. Supported Coop. Work*, 11 (3):411–446, November 2002. ISSN 0925-9724. doi: 10.1023/A:1021271517844. URL http://dx.doi.org/10.1023/A:1021271517844.

[57] Carl Gutwin, Jeff Dyck, and Chris Fedak. The effects of dynamic transparency on targeting performance. In *In Proceedings of Graphics Interface*, GI'03, pages 105–112, 2003.

[58] Kim Halskov, Peter Dalsgaard, and Louise Bak Stolze. Analysing engaging experiences with a tangible 3d tabletop. In *Proceedings of the Conference on Advances in Computer Entertainment Technology*, ACE '14, pages 29:1–29:10, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2945-3. doi: 10.1145/2663806.2663831. URL http://doi.acm.org/10.1145/2663806.2663831.

[59] Takumi Hamasaki, Yuta Itoh, Yuichi Hiroi, Daisuke Iwai, and Maki Sugimoto. Hysar: Hybrid material rendering by an optical see-through head-mounted display with spatial augmented reality projection. *IEEE Transactions on Visualization and Computer Graphics*, PP(99):1–1, 2018. ISSN 1077-2626. doi: 10.1109/TVCG.2018.2793659. URL https://doi.org/10.1109/TVCG.2018.2793659.

[60] Mark Hancock, Miguel Nacenta, Carl Gutwin, and Sheelagh Carpendale. The effects of changing projection geometry on the interpretation of 3d orientation on tabletops. In *Proceedings of the ACM International Conference*

*on Interactive Tabletops and Surfaces*, ITS '09, pages 157–164, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-733-2. URL http://doi.acm.org/10.1145/1731903.1731934.

[61] Beverly L. Harrison, Hiroshi Ishii, Kim J. Vicente, and William A. S. Buxton. Transparent layered user interfaces: An evaluation of a display design to enhance focused and divided attention. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 317–324, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-84705-1. doi: 10.1145/223904.223945. URL http://dx.doi.org/10.1145/223904.223945.

[62] Doris Hausen, Hendrik Richter, Adalie Hemme, and Andreas Butz. Comparing input modalities for peripheral interaction: A case study on peripheral music control. In *Human-Computer Interaction – INTERACT 2013*, volume 8119 of *Lecture Notes in Computer Science*, pages 162–179. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-40476-4. doi: 10.1007/978-3-642-40477-1_10. URL http://dx.doi.org/10.1007/978-3-642-40477-1_10.

[63] Eugene Hecht. *Optics*. Pearson Education Limited, 4th international edition edition, 2013.

[64] Heejeong Heo, Hyung Kun Park, Seungki Kim, Jeeyong Chung, Geehyuk Lee, and Woohun Lee. Transwall: A transparent double-sided touch display facilitating co-located face-to-face interactions. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, pages 435–438, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2474-8. doi: 10.1145/2559206.2574828. URL http://doi.acm.org/10.1145/2559206.2574828.

[65] Jan Herling and Wolfgang Broll. Pixmix: A real-time approach to high-quality diminished reality. In *Proceedings of the 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, ISMAR '12, pages 141–150, Washington, DC, USA, 2012. IEEE Computer Society. ISBN 978-1-4673-4660-3. doi: 10.1109/ISMAR.2012.6402551. URL http://dx.doi.org/10.1109/ISMAR.2012.6402551.

[66] 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, pages 1957–1967, New York, NY, USA, 2016.

ACM. ISBN 978-1-4503-3362-7. URL http://doi.acm.org/10.1145/2858036.2858134.

[67] Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. Holodesk: Direct 3d interactions with a situated see-through display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2421–2430, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. URL http://doi.acm.org/10.1145/2207676.2208405.

[68] Masahito Hirakawa and Satoshi Koike. A Collaborative Augmented Reality System using Transparent Display. In *Proceedings of the IEEE Sixth International Symposium on Multimedia Software Engineering*, ISMSE '04, pages 410–416, 2004. doi: 10.1109/MMSE.2004.2.

[69] David Holman and Roel Vertegaal. Organic user interfaces: Designing computers in any way, shape, or form. *Commun. ACM*, 51(6):48–55, June 2008. ISSN 0001-0782. doi: 10.1145/1349026.1349037. URL http://doi.acm.org/10.1145/1349026.1349037.

[70] Oren Horev. 'Talking to the Hand' An exploration into shape shifting objects and morphing interfaces, 2006. Master's thesis.

[71] Matthias B. Hullin, Ivo Ihrke, Wolfgang Heidrich, Tim Weyrich, Gerwin Damberg, and Martin Fuchs. Computational Fabrication and Display of Material Appearance. In *EUROGRAPHICS '13 State-of-the-Art Report*, 2013. doi: 10.2312/conf/EG2013/stars/137-153.

[72] Junko Ichino, Kazuo Isoda, Ayako Hanai, and Tetsuya Ueda. Effects of the display angle in museums on user's cognition, behavior, and subjective responses. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2979–2988, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2481413. URL http://doi.acm.org/10.1145/2470654.2481413.

[73] Masahiko Inami, Naoki Kawakami, Dairoku Sekiguchi, Yasuyuki Yanagida, Taro Maeda, and Susumu Tachi. Visuo-haptic display using head-mounted projector. In *Proceedings of the IEEE Virtual Reality 2000 Conference*, VR '00, pages 233–, Washington, DC, USA, 2000. IEEE Computer Society. ISBN 0-7695-0478-7. URL http://dl.acm.org/citation.cfm?id=832288.835773.

[74] Kori Inkpen, Kirstie Hawkey, Melanie Kellar, Regan Mandryk, Karen Parker, Derek Reilly, Stacey Scott, and Tara Whalen. Exploring Display

Factors that Influence Co-Located Collaboration: Angle, Size, Number, and User Arrangement. In *Proceedings of HCI International*, HCII'05, 2005.

[75] Hiroshi Ishii and Minoru Kobayashi. Clearboard: A seamless medium for shared drawing and conversation with eye contact. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 525–532, New York, NY, USA, 1992. ACM. ISBN 0-89791-513-5. doi: 10.1145/142750.142977. URL http://doi.acm.org/10.1145/142750.142977.

[76] 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, pages 234–241, New York, NY, USA, 1997. ACM. ISBN 0-89791-802-9. doi: 10.1145/258549.258715. URL http://doi.acm.org/10.1145/258549.258715.

[77] Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. Radical atoms: Beyond tangible bits, toward transformable materials. *interactions*, 19(1):38–51, January 2012. ISSN 1072-5520. URL http://doi.acm.org/10.1145/2065327.2065337.

[78] Hiroshi Ishii, Daniel Leithinger, Sean Follmer, Amit Zoran, Philipp Schoessler, and Jared Counts. Transform: Embodiment of "radical atoms" at milano design week. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pages 687–694, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3146-3. doi: 10.1145/2702613.2702969. URL http://doi.acm.org/10.1145/2702613.2702969.

[79] Daisuke Iwai and Kosuke Sato. Limpid desk: See-through access to disorderly desktop in projection-based mixed reality. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '06, pages 112–115, New York, NY, USA, 2006. ACM. ISBN 1-59593-321-2. doi: 10.1145/1180495.1180519. URL http://doi.acm.org/10.1145/1180495.1180519.

[80] Shahram Izadi, Steve Hodges, Stuart Taylor, Dan Rosenfeld, Nicolas Villar, Alex Butler, and Jonathan Westhues. Going beyond the display: A surface technology with an electronically switchable diffuser. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pages 269–278, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-975-3. doi: 10.1145/1449715.1449760. URL http://doi.acm.org/10.1145/1449715.1449760.

[81] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, and Andrew Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 559–568, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. doi: 10.1145/2047196.2047270. URL http://doi.acm.org/10.1145/2047196.2047270.

[82] Yvonne Jansen, Pierre Dragicevic, Petra Isenberg, Jason Alexander, Abhijit Karnik, Johan Kildal, Sriram Subramanian, and Kasper Hornbæk. Opportunities and challenges for data physicalization. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 3227–3236, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6. doi: 10.1145/2702123.2702180. URL http://doi.acm.org/10.1145/2702123.2702180.

[83] Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. Roomalive: Magical experiences enabled by scalable, adaptive projector-camera units. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 637–644, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. URL http://doi.acm.org/10.1145/2642918.2647383.

[84] Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. Illumiroom: Peripheral projected illusions for interactive experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 869–878, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. URL http://doi.acm.org/10.1145/2470654.2466112.

[85] Mark W. Jones, J. Andreas Baerentzen, and Milos Sramek. 3d distance fields: A survey of techniques and applications. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):581–599, July 2006. ISSN 1077-2626. doi: 10.1109/TVCG.2006.56. URL http://dx.doi.org/10.1109/TVCG.2006.56.

[86] Yasuaki Kakehi. Transmart Miniscape [Installation], 2012. Retrieved March 20, 2016 from http://www.xlab.sfc.keio.ac.jp/?works=transmart-miniascape.

[87] Abhijit Karnik, Diego Martinez Plasencia, Walterio Mayol-Cuevas, and Sriram Subramanian. Pivot: Personalized view-overlays for tabletops. In

*Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 271–280, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1580-7. doi: 10.1145/2380116.2380151. URL http://doi.acm.org/10.1145/2380116.2380151.

[88] Abhijit Karnik, Walterio Mayol-Cuevas, and Sriram Subramanian. MUSTARD: A multi user see through AR display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2541–2550, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2208641. URL http://doi.acm.org/10.1145/2207676.2208641.

[89] A. Karpathy, S. Miller, and Li Fei-Fei. Object discovery in 3d scenes via shape analysis. In *Proceedings of the 2013 IEEE International Conference on Robotics and Automation*, ICRA'13, pages 2088–2095, 2013. doi: 10.1109/ICRA.2013.6630857. URL https://doi.org/10.1109/ICRA.2013.6630857.

[90] John Kestner, Daniel Leithinger, Jaekyung Jung, and Michelle Petersen. Proverbial wallet: Tangible interface for financial awareness. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, pages 55–56, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-493-5. URL http://doi.acm.org/10.1145/1517664.1517683.

[91] Heesun Kim, Seung A. Lee, Hyoyoung Kim, and Ji-Hyung Park. Twosome coloring: Designing a playful coloring system on dual-sided transparent surfaces. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces*, ISS '16, pages 373–378, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4248-3. doi: 10.1145/2992154.2996783. URL http://doi.acm.org/10.1145/2992154.2996783.

[92] Konstantin Klamka and Raimund Dachselt. Illumipaper: Illuminated interactive paper. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 5605–5618, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025525. URL http://doi.acm.org/10.1145/3025453.3025525.

[93] R. IL Kollarits, C. Woodworth, J. E Ribera, and R. D. Gitlin. An Eye-Contact Camera Display System for Video Phone Applications. *SID Digest*, 1995.

[94] Robert Kraut, Carmen Egido, and Jolene Galegher. Patterns of contact and communication in scientific research collaboration. In *Proceedings of*

*the 1988 ACM Conference on Computer-supported Cooperative Work*, CSCW '88, pages 1–12, New York, NY, USA, 1988. ACM. ISBN 0-89791-282-9. doi: 10.1145/62266.62267. URL http://doi.acm.org/10.1145/62266.62267.

[95] Denis Lalanne and Agnes Lisowska Masson. A fitt of distraction: Measuring the impact of distracters and multi-users on pointing efficiency. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 2125–2130, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0268-5. doi: 10.1145/1979742.1979908. URL http://doi.acm.org/10.1145/1979742.1979908.

[96] Robert S. Laramee and Colin Ware. Rivalry and interference with a head-mounted display. *ACM Trans. Comput.-Hum. Interact.*, 9(3):238–251, 2002. ISSN 1073-0516. doi: 10.1145/568513.568516. URL http://doi.acm.org/10.1145/568513.568516.

[97] Nilli Lavie. Distracted and confused?: Selective attention under load. *Trends in Cognitive Sciences*, 9(2), 2005. URL https://doi.org/10.1016/j.tics.2004.12.004.

[98] Jinha Lee, Alex Olwal, Hiroshi Ishii, and Cati Boulanger. Spacetop: Integrating 2d and spatial 3d interactions in a see-through desktop environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 189–192, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470680. URL http://doi.acm.org/10.1145/2470654.2470680.

[99] Daniel Leithinger and Hiroshi Ishii. Relief: A scalable actuated shape display. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '10, pages 221–222, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-841-4. doi: 10.1145/1709886.1709928. URL http://doi.acm.org/10.1145/1709886.1709928.

[100] Daniel Leithinger, David Lakatos, Anthony DeVincenzi, Matthew Blackshaw, and Hiroshi Ishii. Direct and gestural interaction with relief: A 2.5d shape display. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 541–548, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. URL http://doi.acm.org/10.1145/2047196.2047268.

[101] Daniel Leithinger, Sean Follmer, Alex Olwal, Samuel Luescher, Akimitsu Hogge, Jinha Lee, and Hiroshi Ishii. Sublimate: State-changing virtual and

physical rendering to augment interaction with shape displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1441–1450, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2466191. URL http://doi.acm.org/10.1145/2470654.2466191.

[102] Alex Leykin and Mihran Tuceryan. Automatic determination of text readability over textured backgrounds for augmented reality systems. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '04, pages 224–230, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2191-6. doi: 10.1109/ISMAR.2004.22. URL http://dx.doi.org/10.1109/ISMAR.2004.22.

[103] Jiannan Li, Saul Greenberg, Ehud Sharlin, and Joaquim Jorge. Interactive two-sided transparent displays: Designing for collaboration. In *Proceedings of the 2014 Conference on Designing Interactive Systems*, DIS '14, pages 395–404, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2902-6. doi: 10.1145/2598510.2598518. URL http://doi.acm.org/10.1145/2598510.2598518.

[104] Jiannan Li, Saul Greenberg, and Ehud Sharlin. A two-sided collaborative transparent display supporting workspace awareness. *International Journal of Human-Computer Studies*, 101(Supplement C):23 – 44, 2017. ISSN 1071-5819. doi: https://doi.org/10.1016/j.ijhcs.2017.01.003. URL http://www.sciencedirect.com/science/article/pii/S1071581917300034.

[105] Yung-Ta Lin, Yi-Chi Liao, Shan-Yuan Teng, Yi-Ju Chung, Liwei Chan, and Bing-Yu Chen. Outside-in: Visualizing out-of-sight regions-of-interest in a 360° video using spatial picture-in-picture previews. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 255–265, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126656. URL http://doi.acm.org/10.1145/3126594.3126656.

[106] David Lindlbauer and Andrew D. Wilson. Remixed reality: Manipulating space and time in augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5620-6. doi: 10.1145/3173574.3173703. URL https://doi.org/10.1145/3173574.3173703.

[107] David Lindlbauer, Toru Aoki, Robert Walter, Yuji Uema, Anita Höchtl, Michael Haller, Masahiko Inami, and Jörg Müller. Tracs: Transparency-

control for see-through displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 657–661, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. URL http://doi.acm.org/10.1145/2642918.2647350.

[108] David Lindlbauer, Jens Emil Grønbæk, Morten Birk, Kim Halskov, Marc Alexa, and Jörg Müller. Combining shape-changing interfaces and spatial augmented reality enables extended object appearance. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 791–802, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. URL http://doi.acm.org/10.1145/2858036.2858457.

[109] David Lindlbauer, Klemen Lilija, Robert Walter, and Jörg Müller. Influence of display transparency on background awareness and task performance. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1705–1716, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858453. URL http://doi.acm.org/10.1145/2858036.2858453.

[110] David Lindlbauer, Jörg Müller, and Marc Alexa. Changing the appearance of physical interfaces through controlled transparency. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, UIST '16, New York, NY, USA, 2016. ACM. URL http://doi.acm.org/10.1145/2984511.2984556.

[111] David Lindlbauer, Jörg Mueller, and Marc Alexa. Changing the appearance of real-world objects by modifying their surroundings. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3954–3965, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025795. URL http://doi.acm.org/10.1145/3025453.3025795.

[112] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, Eric Lecolinet, and Wendy E. Mackay. Effects of display size and navigation type on a classification task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 4147–4156, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2473-1. doi: 10.1145/2556288.2557020. URL http://doi.acm.org/10.1145/2556288.2557020.

[113] Wendy E. Mackay. Augmented reality: Linking real and virtual worlds: A new paradigm for interacting with computers. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '98, pages 13–21, New York,

NY, USA, 1998. ACM. doi: 10.1145/948496.948498. URL http://doi.acm.org/10.1145/948496.948498.

[114] Paul P. Maglio and Christopher S. Campbell. Tradeoffs in displaying peripheral information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, pages 241–248, New York, NY, USA, 2000. ACM. ISBN 1-58113-216-6. doi: 10.1145/332040.332438. URL http://doi.acm.org/10.1145/332040.332438.

[115] Steve Mann. Mediated Reality. Technical Report MIT-ML Percom TR-260, University of Toronto, 1994.

[116] Steve Mann. Mediated reality with implementations for everyday life. *Presence Connect*, August 2002.

[117] Steve Mann and James Fung. Eyetap devices for augmented, deliberately diminished, or otherwise altered visual perception of rigid planar patches of real-world scenes. *Presence: Teleoperators and Virtual Environments*, 11(2): 158–175, 2002. doi: 10.1162/1054746021470603. URL https://doi.org/10.1162/1054746021470603.

[118] Wojciech Matusik, Boris Ajdin, Jinwei Gu, Jason Lawrence, Hendrik P. A. Lensch, Fabio Pellacini, and Szymon Rusinkiewicz. Printing spatially-varying reflectance. *ACM Trans. Graph.*, 28(5):128:1–128:9, December 2009. ISSN 0730-0301. doi: 10.1145/1618452.1618474. URL http://doi.acm.org/10.1145/1618452.1618474.

[119] Michael McKenna. Interactive viewpoint control and three-dimensional operations. In *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, I3D '92, pages 53–56, New York, NY, USA, 1992. ACM. ISBN 0-89791-467-8. URL http://doi.acm.org/10.1145/147156.147163.

[120] Ajmal S. Mian, Mohammed Bennamoun, and Robyn Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1584–1601, Oct 2006. ISSN 0162-8828. doi: 10.1109/TPAMI.2006.213.

[121] Florian Michahelles, Frederic Thiesse, Albrecht Schmidt, and John R. Williams. Pervasive rfid and near field communication technology. *IEEE Pervasive Computing*, 6(undefined):94–96, c3, 2007. ISSN 1536-1268. doi: 10.1109/MPRV.2007.64.

[122] Microsoft. Hololens, 2016. Retrieved January 25, 2018 from https://www.microsoft.com/en-us/hololens.

[123] Paul Milgram and Fumio Kishino. A Taxonomy of Mixed Reality Visual Displays. *EICE Transactions on Information Systems*, E77-D(12), 1994.

[124] Viktor Miruchna, Robert Walter, David Lindlbauer, Maren Lehmann, Regine von Klitzing, and Jörg Müller. Geltouch: Localized tactile feedback through thin, programmable gel. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pages 3–10, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3779-3. doi: 10.1145/2807442.2807487. URL http://doi.acm.org/10.1145/2807442.2807487.

[125] Takashi Miyaki and Jun Rekimoto. Lidarman: Reprogramming reality with egocentric laser depth scanning. In *ACM SIGGRAPH 2016 Emerging Technologies*, SIGGRAPH '16, pages 15:1–15:2, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4372-5. doi: 10.1145/2929464.2929481. URL http://doi.acm.org/10.1145/2929464.2929481.

[126] Mathias Mohring, Christian Lessig, and Oliver Bimber. Video see-through AR on consumer cell-phones. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '04, pages 252–253, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2191-6. doi: 10.1109/ISMAR.2004.63. URL http://dx.doi.org/10.1109/ISMAR.2004.63.

[127] John Y. Moon. Augmented shadow, 2010. Retrieved September 12, 2016 from http://cargocollective.com/joonmoon/Augmented-Shadow.

[128] Ryan E. B. Mruczek, Christopher D. Blair, Lars Strother, and Gideon P. Caplovitz. The dynamic ebbinghaus: motion dynamics greatly enhance the classic contextual size illusion. *Frontiers in Human Neuroscience*, 9(77), 2015. URL http://doi.org/10.3389/fnhum.2015.00077.

[129] Miguel A. Nacenta, Mark Hancock, Carl Gutwin, and Sheelagh Carpendale. The effects of changing projection geometry on perception of 3d objects on and around tabletops. *ACM Trans. Comput.-Hum. Interact.*, 23(2):11:1–11:54, May 2016. ISSN 1073-0516. URL http://doi.acm.org/10.1145/2845081.

[130] Takeshi Naemura, Takuya Nitta, Atsushi Mimura, and Hiroshi Harashima. Virtual shadows-enhanced interaction in mixed reality environment. In

*Proceedings IEEE Virtual Reality*, IEEE VR'02, pages 293–294, 2002. doi: 10.1109/VR.2002.996545.

[131] Ken Nakagaki, Sean Follmer, and Hiroshi Ishii. Lineform: Actuated curve interfaces for display, interaction, and constraint. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pages 333–339, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3779-3. URL http://doi.acm.org/10.1145/2807442.2807452.

[132] Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. Materiable: Rendering dynamic material properties in response to direct physical touch with shape changing interfaces. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 2764–2772, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858104. URL http://doi.acm.org/10.1145/2858036.2858104.

[133] Vidhya Navalpakkam, Justin Rao, and Malcolm Slaney. Using gaze patterns to study and predict reading struggles due to distraction. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 1705–1710, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0268-5. doi: 10.1145/1979742.1979832. URL http://doi.acm.org/10.1145/1979742.1979832.

[134] Simon Olberding, Michael Wessely, and Jürgen Steimle. Printscreen: Fabricating highly customizable thin-film touch-displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 281–290, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. URL http://doi.acm.org/10.1145/2642918.2647413.

[135] Simon Olberding, Sergio Soto Ortega, Klaus Hildebrandt, and Jürgen Steimle. Foldio: Digital fabrication of interactive and shape-changing objects with foldable printed electronics. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pages 223–232, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3779-3. URL http://doi.acm.org/10.1145/2807442.2807494.

[136] Alex Olwal, Stephen DiVerdi, Nicola Candussi, Ismo Rakkolainen, and Tobias Höllerer. An Immaterial, Dual-sided Display System with 3D Interaction. In *Proceedings of the IEEE Virtual Reality Conference*, IEEE VR'06, pages 279–280, 2006. doi: 10.1109/VR.2006.25.

[137] Alex Olwal, Stephen DiVerdi, Ismo Rakkolainen, and Tobias Höllerer. Consigalo: Multi-user face-to-face interaction on immaterial displays. In *Proceedings of the 2nd International Conference on INtelligent TEchnologies for Interactive enterTAINment*, INTETAIN '08, pages 8:1–8:9, ICST, Brussels, Belgium, Belgium, 2007. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). ISBN 978-963-9799-13-4. URL http://dl.acm.org/citation.cfm?id=1363200.1363211.

[138] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L. Davidson, Sameh Khamis, Mingsong Dou, Vladimir Tankovich, Charles Loop, Qin Cai, Philip A. Chou, Sarah Mennicken, Julien Valentin, Vivek Pradeep, Shenlong Wang, Sing Bing Kang, Pushmeet Kohli, Yuliya Lutchyn, Cem Keskin, and Shahram Izadi. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 741–754, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4189-9. doi: 10.1145/2984511.2984517. URL http://doi.acm.org/10.1145/2984511.2984517.

[139] Ken Perlin, Salvatore Paxia, and Joel S. Kollin. An autostereoscopic display. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 319–326, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. ISBN 1-58113-208-5. doi: 10.1145/344779.344933. URL http://dx.doi.org/10.1145/344779.344933.

[140] Ben Piper, Carlo Ratti, and Hiroshi Ishii. Illuminating clay: A 3-D tangible interface for landscape analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pages 355–362, New York, NY, USA, 2002. ACM. ISBN 1-58113-453-3. URL http://doi.acm.org/10.1145/503376.503439.

[141] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. The go-go interaction technique: Non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, pages 79–80, New York, NY, USA, 1996. ACM. ISBN 0-89791-798-7. doi: 10.1145/237091.237102. URL http://doi.acm.org/10.1145/237091.237102.

[142] Ivan Poupyrev, Tatsushi Nashida, Shigeaki Maruyama, Jun Rekimoto, and Yasufumi Yamaji. Lumen: Interactive visual and shape display for calm

computing. In *ACM SIGGRAPH 2004 Emerging Technologies*, SIGGRAPH '04, pages 17–, New York, NY, USA, 2004. ACM. ISBN 1-58113-896-2. doi: 10.1145/1186155.1186173. URL http://doi.acm.org/10.1145/1186155.1186173.

[143] Ivan Poupyrev, Tatsushi Nashida, and Makoto Okabe. Actuation and tangible user interfaces: The vaucanson duck, robots, and shape displays. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 205–212, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-619-6. doi: 10.1145/1226969.1227012. URL http://doi.acm.org/10.1145/1226969.1227012.

[144] Kathrin Probst, David Lindlbauer, Michael Haller, Bernhard Schwartz, and Andreas Schrempf. A chair as ubiquitous input device: Exploring semaphoric chair gestures for focused and peripheral interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 4097–4106, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2473-1. doi: 10.1145/2556288.2557051. URL http://doi.acm.org/10.1145/2556288.2557051.

[145] Umar Rashid, Miguel A. Nacenta, and Aaron Quigley. Factors influencing visual attention switch in multi-display user interfaces: A survey. In *Proceedings of the 2012 International Symposium on Pervasive Displays*, PerDis '12, pages 1:1–1:6, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1414-5. doi: 10.1145/2307798.2307799. URL http://doi.acm.org/10.1145/2307798.2307799.

[146] Ramesh Raskar, Greg Welch, Matt Cutts, Adam Lake, Lev Stesin, and Henry Fuchs. The office of the future: A unified approach to image-based modeling and spatially immersive displays. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 179–188, New York, NY, USA, 1998. ACM. ISBN 0-89791-999-8. URL http://doi.acm.org/10.1145/280814.280861.

[147] Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 89–102, London, UK, UK, 2001. Springer-Verlag. ISBN 3-211-83709-4. URL http://dl.acm.org/citation.cfm?id=647653.732300.

[148] Majken K. Rasmussen, Esben W. Pedersen, Marianne G. Petersen, and Kasper Hornbæk. Shape-changing interfaces: A review of the design space

and open research questions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 735–744, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. URL http://doi.acm.org/10.1145/2207676.2207781.

[149] Majken Kirkegård Rasmussen, Timothy Merritt, Miguel Bruns Alonso, and Marianne Graves Petersen. Balancing user and system control in shape-changing interfaces: A designerly exploration. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '16, pages 202–210, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3582-9. doi: 10.1145/2839462.2839499. URL http://doi.acm.org/10.1145/2839462.2839499.

[150] Sharif Razzaque, Zachariah Kohn, and Mary C. Whitton. Redirected Walking. In *Eurographics 2001 - Short Presentations*. Eurographics Association, 2001. doi: 10.2312/egs.20011036.

[151] Adrian Reetz and Carl Gutwin. Making big gestures: Effects of gesture size on observability and identification for co-located group awareness. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 4087–4096, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2473-1. doi: 10.1145/2556288.2557219. URL http://doi.acm.org/10.1145/2556288.2557219.

[152] Jun Rekimoto. Squama: Modular visibility control of walls and windows for programmable physical architectures. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, pages 168–171, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1287-5. doi: 10.1145/2254556.2254587. URL http://doi.acm.org/10.1145/2254556.2254587.

[153] George Robertson, Mary Czerwinski, Patrick Baudisch, Brian Meyers, Daniel Robbins, Greg Smith, and Desney Tan. The large-display user experience. *IEEE Computer Graphics and Applications*, 25(4):44–51, July 2005. ISSN 0272-1716. doi: 10.1109/MCG.2005.88.

[154] Joan Sol Roo and Martin Hachet. One reality: Augmenting how the physical world is experienced by combining multiple mixed reality modalities. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 787–795, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126638. URL http://doi.acm.org/10.1145/3126594.3126638.

[155] Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. Morphees: Toward high "shape resolution" in self-actuated flexible mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 593–602, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470738. URL http://doi.acm.org/10.1145/2470654.2470738.

[156] Munehiko Sato, Shigeo Yoshida, Alex Olwal, Boxin Shi, Atsushi Hiyama, Tomohiro Tanikawa, Michitaka Hirose, and Ramesh Raskar. Spectrans: Versatile material classification for interaction with textureless, specular and transparent surfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 2191–2200, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6. doi: 10.1145/2702123.2702169. URL http://doi.acm.org/10.1145/2702123.2702169.

[157] Christian Schüller, Daniele Panozzo, and Olga Sorkine-Hornung. Appearance-mimicking surfaces. *ACM Trans. Graph.*, 33(6):216:1–216:10, November 2014. ISSN 0730-0301. doi: 10.1145/2661229.2661267. URL http://doi.acm.org/10.1145/2661229.2661267.

[158] Mel Schwartz. *Smart Materials*. CRC Press, 2008.

[159] Jean Serra. *Image Analysis and Mathematical Morphology*. Academic Press, Inc., 1983.

[160] Lavanya Sharan, Ce Liu, Ruth Rosenholtz, and Edward H. Adelson. Recognizing materials using perceptually inspired features. *International Journal of Computer Vision*, 103(3):348–371, 2013. doi: 10.1007/s11263-013-0609-0. URL http://dx.doi.org/10.1007/s11263-013-0609-0.

[161] Anirudh Sharma, Lirong Liu, and Pattie Maes. Glassified: An augmented ruler based on a transparent display for real-time interactions with paper. In *Proceedings of the Adjunct Publication of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13 Adjunct, pages 21–22, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2406-9. doi: 10.1145/2508468.2514937. URL http://doi.acm.org/10.1145/2508468.2514937.

[162] Xiaobin Shen. An evaluation methodology for ambient displays. *Journal of Engineering, Computing and Architecture*, 1, 2007.

[163] Srikanth Kirshnamachari Sridharan, Juan David Hincapié-Ramos, David R. Flatla, and Pourang Irani. Color correction for optical see-through displays using display color profiles. In *Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology*, VRST '13, pages 231–240, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2379-6. doi: 10.1145/2503713.2503716. URL http://doi.acm.org/10.1145/2503713.2503716.

[164] Ian Stavness, Billy Lam, and Sidney Fels. pcubee: A perspective-corrected handheld cubic display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1381–1390, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. URL http://doi.acm.org/10.1145/1753326.1753535.

[165] Richard Stoakley, Matthew J. Conway, and Randy Pausch. Virtual reality on a WIM: interactive worlds in miniature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 265–272, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-84705-1. doi: 10.1145/223904.223938. URL http://dx.doi.org/10.1145/223904.223938.

[166] Miriam Sturdee and Jason Alexander. Analysis and classification of shape-changing interfaces for design and application-based research. *ACM Comput. Surv.*, 51(1):2:1–2:32, January 2018. ISSN 0360-0300. doi: 10.1145/3143559. URL http://doi.acm.org/10.1145/3143559.

[167] Ivan E. Sutherland. The ultimate display. In *Proceedings of the IFIP Congress*, pages 506–508, 1965.

[168] Kishore Swaminathan and Steve Sato. Interaction design for large displays. *interactions*, 4(1):15–24, January 1997. ISSN 1072-5520. doi: 10.1145/242388.242395. URL http://doi.acm.org/10.1145/242388.242395.

[169] Tomohiro Tachi. Origamizing polyhedral surfaces. *IEEE TVCG*, 16(2): 298–311, 2010. ISSN 1077-2626. doi: 10.1109/TVCG.2009.67.

[170] Desney S. Tan and Mary Czerwinski. Effects of visual separation and physical discontinuities when distributing information across multiple displays. In *Proceedings of the Annual Conference of CHISIG, the Computer Human Interaction Special Interest Group of the Ergonomics Society of Australia*, OZCHI 2003, 2003.

[171] Chengcheng Tang, Pengbo Bo, Johannes Wallner, and Helmut Pottmann. Interactive design of developable surfaces. *ACM Trans. Graph.*, 35(2):

12:1–12:12, January 2016. ISSN 0730-0301. URL http://doi.acm.org/10.1145/2832906.

[172] John C. Tang and Scott Minneman. Videowhiteboard: Video shadows to support remote collaboration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 315–322, New York, NY, USA, 1991. ACM. ISBN 0-89791-383-3. doi: 10.1145/108844.108932. URL http://doi.acm.org/10.1145/108844.108932.

[173] John C. Tang and Scott L. Minneman. Videodraw: A video interface for collaborative drawing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 313–320, New York, NY, USA, 1990. ACM. ISBN 0-201-50932-6. doi: 10.1145/97243.97302. URL http://doi.acm.org/10.1145/97243.97302.

[174] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 4.11 edition, 2017. URL http://doc.cgal.org/4.11/Manual/packages.html.

[175] Jonas Togler, Fabian Hemmert, and Reto Wettach. Living interfaces: the thrifty faucet. In *Proc. TEI '09*, 2009.

[176] Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97 – 136, 1980. ISSN 0010-0285. doi: http://dx.doi.org/10.1016/0010-0285(80)90005-5. URL http://www.sciencedirect.com/science/article/pii/0010028580900055.

[177] Richard A. Tyrrell and Herschel W. Leibowitz. The relation of vergence effort to report of visual fatigue following prolonged near work. *Human Factors*, 32(3), 1990. doi: doi:10.1177/001872089003200307.

[178] John Underkoffler and Hiroshi Ishii. Illuminating light: An optical design tool with a luminous-tangible interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '98, pages 542–549, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-30987-4. URL http://dx.doi.org/10.1145/274644.274717.

[179] John Underkoffler and Hiroshi Ishii. Urp: A luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 386–393, New York, NY, USA, 1999. ACM. ISBN 0-201-48559-1. URL http://doi.acm.org/10.1145/302979.303114.

[180] John Underkoffler, Brygg Ullmer, and Hiroshi Ishii. Emancipated pixels: Real-world graphics in the luminous room. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 385–392, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-48560-5. URL http://dx.doi.org/10.1145/311535.311593.

[181] Pablo Valbuena. Augmented sculpture, 2008. Retrieved September 17, 2015 from http://www.pablovalbuena.com/selectedwork/augmented-sculpture-v1/.

[182] Julien Valentin, Vibhav Vineet, Ming-Ming Cheng, David Kim, Jamie Shotton, Pushmeet Kohli, Matthias Nießner, Antonio Criminisi, Shahram Izadi, and Philip Torr. Semanticpaint: Interactive 3d labeling and learning at your fingertips. *ACM Trans. Graph.*, 34(5):154:1–154:17, November 2015. ISSN 0730-0301. doi: 10.1145/2751556. URL http://doi.acm.org/10.1145/2751556.

[183] Roel Vertegaal and Ivan Poupyrev. Organic user interfaces: Introduction. *Commun. ACM*, 51(6):26–30, June 2008. ISSN 0001-0782. doi: 10.1145/1349026.1349033. URL http://doi.acm.org/10.1145/1349026.1349033.

[184] John Viega, Matthew J. Conway, George Williams, and Randy Pausch. 3d magic lenses. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, pages 51–58, New York, NY, USA, 1996. ACM. ISBN 0-89791-798-7. doi: 10.1145/237091.237098. URL http://doi.acm.org/10.1145/237091.237098.

[185] Chun Wang, Nolan T Flynn, and Robert Langer. Controlled structure and properties of thermoresponsive nanoparticle–hydrogel composites. *Advanced Materials*, 16(13):1074–1079, 2004.

[186] Yanan Wang, Shijian Luo, Yujia Lu, Hebo Gong, Yexing Zhou, Shuai Liu, and Preben Hansen. Animskin: Fabricating epidermis with interactive, functional and aesthetic color animation. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, pages 397–401, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4922-2. doi: 10.1145/3064663.3064687. URL http://doi.acm.org/10.1145/3064663.3064687.

[187] Mark Weiser. The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.*, 3(3):3–11, 1999. URL http://doi.acm.org/10.1145/329124.329126.

[188] Gordon Wetzstein and Oliver Bimber. Radiometric compensation through inverse light transport. In *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, PG '07, pages 391–399, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3009-5. doi: 10.1109/PG.2007.52. URL http://dx.doi.org/10.1109/PG.2007.52.

[189] Lance Williams. Casting curved shadows on curved surfaces. *SIGGRAPH Comput. Graph.*, 12(3):270–274, August 1978. ISSN 0097-8930. URL http://doi.acm.org/10.1145/965139.807402.

[190] Karl Willis, Eric Brockmeyer, Scott Hudson, and Ivan Poupyrev. Printed optics: 3d printing of embedded optical elements for interactive devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 589–598, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1580-7. doi: 10.1145/2380116.2380190. URL http://doi.acm.org/10.1145/2380116.2380190.

[191] Andrew Wilson, Hrvoje Benko, Shahram Izadi, and Otmar Hilliges. Steerable augmented reality with the beamatron. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 413–422, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1580-7. doi: 10.1145/2380116.2380169. URL http://doi.acm.org/10.1145/2380116.2380169.

[192] Andrew D. Wilson. Fast lossless depth image compression. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pages 100–105, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4691-7. doi: 10.1145/3132272.3134144. URL http://doi.acm.org/10.1145/3132272.3134144.

[193] Andrew D. Wilson and Hrvoje Benko. Projected augmented reality with the RoomAlive toolkit. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces*, ISS '16, pages 517–520, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4248-3. doi: 10.1145/2992154.2996362. URL http://doi.acm.org/10.1145/2992154.2996362.

[194] Xuedi Chen and Pedro G. C. Oliveira. X.pose, 2014. Retrieved January 24, 2018 from http://xc-xd.com/xpose/.

[195] Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Follmer, Clark Della Silva, and Hiroshi Ishii. Pneui: Pneumatically actuated soft composite materials for shape changing interfaces. In *Proceedings of the 26th Annual ACM*

*Symposium on User Interface Software and Technology*, UIST '13, pages 13–22, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2268-3. URL http://doi.acm.org/10.1145/2501988.2502037.

[196] Lining Yao, Jifei Ou, Chin-Yi Cheng, Helene Steiner, Wen Wang, Guanyun Wang, and Hiroshi Ishii. biologic: Natto cells as nanoactuators for shape changing interfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1–10, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6. doi: 10.1145/2702123.2702611. URL http://doi.acm.org/10.1145/2702123.2702611.

[197] Ya-Ting Yue, Yong-Liang Yang, Gang Ren, and Wenping Wang. Scenectrl: Mixed reality enhancement via efficient scene editing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 427–436, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126601. URL http://doi.acm.org/10.1145/3126594.3126601.