



HAL
open science

Tangible Interaction for Wall Displays

Emmanuel Courtoux

► **To cite this version:**

Emmanuel Courtoux. Tangible Interaction for Wall Displays. Human-Computer Interaction [cs.HC]. Université Paris-Saclay, 2023. English. NNT : 2023UPASG028 . tel-04094239

HAL Id: tel-04094239

<https://theses.hal.science/tel-04094239>

Submitted on 10 May 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Tangible Interaction for Wall Displays

Interaction Tangible pour les Murs d'Écrans

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et Technologies de l'Information et de la
Communication (STIC)
Spécialité de doctorat: Informatique
Graduate School : Informatique et Sciences du Numérique. Référent : Faculté des
sciences d'Orsay

Thèse préparée dans l'unité de recherche **Laboratoire interdisciplinaire des sciences
du numérique (Université Paris-Saclay, CNRS)**, sous la direction de
Caroline APPERT, Directrice de Recherche
et **Olivier CHAPUIS**, Chargé de Recherche HC

Thèse soutenue à Paris-Saclay, le 11 Avril 2023, par

Emmanuel COURTOUX

Composition du jury

Myriam LEWKOWICZ Professeure, Université de Technologie de Troyes	Présidente
Martin HACHET Directeur de Recherche, Centre Inria de l'université de Bordeaux	Rapporteur & Examineur
Jan BORCHERS Professor, RWTH Aachen University	Rapporteur & Examineur
Anke BROCK Assistant Professor / Inria Researcher, ENAC Toulouse	Examinatrice

ACKNOWLEDGEMENTS

Tout d'abord merci aux membres de mon jury de thèse, Myriam, Anke, Jan et Martin. Merci pour votre présence et l'intérêt que vous avez portés à mes travaux. C'est un vrai plaisir de pouvoir partager avec vous toutes ces idées et réflexions sur mes recherches passées et à venir. Sans vous, tout ça n'aurait pas été possible.

Viennent ensuite mes encadrants, Olivier et Caroline, sans qui à peu près tout ce que j'ai entrepris durant ma thèse n'aurait pas abouti. Merci de m'avoir emmené jusque-là, c'est avec évidence que je partage le mérite avec vous. Je remercie aussi bien évidemment les autres chercheurs permanents de l'équipe : Anastasia, Emmanuel et Vanessa pour leurs conseils avisés.

Un gros merci également à Romain directeur du FabLab qui, depuis le master, m'a appris tous les rouages relatifs à la fabrication assisté par ordinateur et bien plus.

Merci aussi à Olivier Gladin pour m'avoir appris à utiliser WILDER et de m'avoir aidé pour tous les problèmes techniques.

Merci aussi à mes chers collègues, ceux qui sont déjà parti et ceux qui sont encore là. Commençons par les premiers, Raphael, Eugénie et Dylan, qui m'ont accueilli au début de ma thèse. Vous avez su créer et intégrer une ambiance inégalable dans cette équipe, et sans vous c'est certain que tout aurait été plus difficile. Alors merci, mais pas seulement merci pour les innombrables coup de main que vous avez pu me donner pendant la thèse, mais surtout pour les rires à ne plus savoir quoi en faire. Enfin merci à tous ceux qui sont encore là et qui maintiennent la bonne humeur dans laquelle j'ai passé ces trois années et demi de doctorat Mehdi, Vincent, Camille, Ludo, Théo et Gaëlle.

J'aimerais aussi remercier mes proches, à commencer par mon chat, quand j'ai enfin compris que tous ces miaulements n'étaient pas simplement pour me demander à manger, mais bien pour m'encourager à avancer sur mes projets. Un grand merci également à ma compagne Tamara, pour m'avoir soutenu tout du long de ma thèse et au flot de mes humeurs changeantes. Merci aussi pour les illustrations de ma soutenance ! Je remercie également bien entendu mes parents, mes frères et toute ma famille pour leur soutien et leur éternelle curiosité sur mon doctorat, j'ai l'impression d'avoir beaucoup répondu aux mêmes questions, mais ça me faisait toujours autant plaisir de pouvoir partager ça avec vous. Je vous embrasse ! Viennent enfin mes amis. Vous avez contribué à ma thèse bien plus que vous ne pouvez le croire car, sans tous ces bons moments passés avec vous pour sortir un peu la tête de la recherche, je n'aurais pas eu une aussi bonne expérience du doctorat que celle-ci.

Finalement, je remercie le CESFO, qui m'a accueilli lors de mon premier jour dans l'équipe et qui est le meilleur lieu rapport qualité prix pour manger sur le plateau et avoir des discussions enflammées sur des sujets sans importance.

Encore merci à tous !

ABSTRACT

Wall displays immerse users in large, high-resolution information spaces. They are well suited for data analysis, as users only need to move around the physical space to explore the virtual information space displayed on the wall. They also facilitate collaboration as their large physical size can accommodate multiple users. However, designing effective ways of interacting with wall displays is challenging. Traditional input devices, such as mice and keyboards, quickly show their limitations in an environment where multiple users can interact and move freely.

HCI literature offers interesting alternatives to traditional input techniques. In particular, Tangible User Interactions (TUIs), where users rely on custom tangible objects to interact with the virtual scene, have proved efficient with different types of displays ranging from smartphones to tabletops. Tangible controllers have natural advantages such as the haptic feedback they provide that enables eyes-free manipulations. They also afford specific grasps and manipulations, guiding users on what they can do with them. Empirical studies that compare tangibles to other forms of input also report quantitative gains in regarding manipulation speed and precision in different hardware setups.

However, designing tangible controllers for wall displays is difficult. First, the large size and vertical orientation of walls must be taken into account to design tangibles with a suitable form factor. Second, users move in space. They move away to get a wider view, move closer to see details, or adjust their physical position based on other users and objects in the room. This means that tangible controllers must be usable regardless of the user's position in the room, which has some impact on design and engineering aspects. Finally, a wall display is often located in an environment that feature other devices and displays. In such cases, designing tangible controllers for a wall display requires to consider the whole multi-display environment, which constrains even more the tangibles' form factor and the underlying technologies.

My thesis work makes three contributions towards enabling tangible interaction with wall displays.

The first project, WallTokens, contributes tangibles for enabling *on surface* interaction with wall displays. WallTokens are low-cost, passive controllers that users can manipulate directly on the wall's surface. WallTokens have a mechanism that allows users to easily attach and detach them from the wall surface, so that when users are done interacting, they can leave them in place and free their hands for other purposes. We report on two studies assessing WallTokens' usability, showing that they are more precise and comfortable than bare-hand gestures to perform low-level manipulations on walls.

The second project, SurfAirs, contributes tangibles that support not only *on surface* interaction but also *distant* interaction with wall displays. We present two possible designs for versatile tangible controllers that can be used both on the wall surface when users need precision and detail, and in the air when they need a wide viewing angle. SurfAirs support both types of input, as well as smooth transitions between the two. We report on two studies that compare SurfAir prototypes with bare hand gestures for performing low-level manipulation tasks. SurfAirs outperform bare hand gestures regarding accuracy, speed and user preference.

The third project contributes a survey about the use of physical controllers to interact with a physical display. Each project is described along twelve dimensions that capture the design aspects of the controller, the properties of the display and how they communicate with each other. We contribute a Web page to explore this list of references along the different dimensions, and use it to discuss the challenges that underlie the design of tangible controllers in a multi-display environment.

RÉSUMÉ

Les murs d'écrans plongent les utilisateurs dans de larges espaces d'informations ultra-haute résolution. Ils sont bien adaptés à l'analyse de grands ensembles de données car les utilisateurs peuvent se déplacer physiquement pour explorer ce qui est affiché à l'écran. Ils facilitent également la collaboration car leur taille permet facilement d'accueillir plusieurs utilisateurs à la fois. Cependant, créer des interactions efficaces avec les murs d'écrans est un défi. Les périphériques traditionnels tel que le clavier et la souris montre vite leur limite dans un contexte où plusieurs utilisateurs interagissent et se déplacent librement.

La littérature en IHM propose des moyens d'interaction alternatifs. En particulier, l'interaction tangible, qui s'appuie sur la manipulation d'objets physiques pour interagir avec des scènes virtuelles, offre des avantages intéressants pour les murs d'écrans. La matérialité des contrôleurs offre un retour haptique qui permet de les manipuler sans les regarder. Leur forme suggère aussi la manière de les attraper et de les manipuler, guidant les utilisateurs sur leurs fonctionnalités. De nombreuses études empiriques comparant les contrôleurs tangibles à d'autres formes d'interaction montrent qu'ils procurent des gains significatifs en termes de vitesse et de précision de manipulation.

Cependant, fabriquer des contrôleurs tangibles pour les murs d'écrans est difficile. Tout d'abord, l'orientation de l'écran et sa taille doivent être prise en compte pour fabriquer des tangibles appropriés. De plus, les utilisateurs sont mobiles : ils s'éloignent pour obtenir un angle de vue plus large, se rapprochent pour voir plus de détails, ou ajustent leur position en fonction de celle des autres utilisateurs. Les contrôleurs tangibles doivent donc être conçus pour être utilisables quelle que soit la position de l'utilisateur dans la pièce. Enfin, un mur d'écrans est souvent situé dans un contexte rassemblant d'autres dispositifs (tables interactives, ordinateurs, etc.). Dans ce cas, il est nécessaire de prendre en compte l'ensemble du contexte, contraignant la forme des tangibles et les technologies sous-jacentes.

Mon travail de thèse propose trois contributions pour faciliter l'interaction tangible avec les murs d'écrans.

Mon premier projet, *WallTokens*, propose des tangibles qui permettent d'interagir sur la surface des murs d'écrans. Les *WallTokens* sont équipés d'un mécanisme qui permet aux utilisateurs de les attacher et de les détacher facilement de la surface du mur. Cela permet de les laisser en place lorsque les utilisateurs veulent libérer leur main pour d'autres tâches. Nous présentons deux études évaluant la facilité d'utilisation et l'efficacité des *WallTokens*. Nos résultats montrent qu'ils sont plus

précis et plus confortables que les interactions tactiles pour effectuer des manipulations de bas niveau sur mur d'écrans.

Mon deuxième projet, *SurfAirs*, propose des tangibles permettant des interactions avec les murs d'écrans en surface, quand les utilisateurs ont besoin de détails et de précision, mais aussi à distance quand ils ont besoin d'un grand angle de vue. Les SurfAirs permettent également une transition continue entre ces deux modes d'interaction. Nous présentons deux études qui comparent les SurfAirs avec des gestes à main nue pour effectuer des tâches de manipulation de bas niveau. Les SurfAirs sont plus performants que les gestes à main nue en termes de précision et de vitesse et les utilisateurs les préfèrent.

Le troisième projet propose une étude de la littérature sur l'utilisation de contrôleur tangible avec des écrans. Chaque article étudié est classifié selon 12 dimensions qui reflètent les aspects de la conception du contrôleur et de l'écran. Nous proposons un outil Web qui permet l'exploration de notre corpus d'articles à travers ces dimensions de classification. Nous discutons ensuite les défis qui sous-tendent la conception de contrôleurs tangibles dans un environnement multi-écrans.

CONTENTS

Acknowledgements	i
Abstract	iii
Résumé	v
Contents	vii
List of Figures	ix
1 Introduction	3
1.1 Research question	8
1.2 Thesis Outline	10
2 Related Work	11
2.1 Benefits of Tangible Controllers	11
2.2 Benefits of Vertical Display Surfaces	12
2.3 Tangibles for Interacting on the Surface	14
2.4 Interacting on the Surface... and from a Distance	16
2.5 Physical Controllers for Interacting with Wall Displays	19
2.6 Summary	21
3 On-Surface Tangible Interaction with Walls	23
3.1 Introduction	24
3.2 WallTokens	25
3.3 Walltokens vs Touch Gestures	32
3.4 Applications	40
3.5 Limitations	43
3.6 Conclusion and Future Work	44
4 On- and Off- Surface Tangible Interaction with Walls	45
4.1 Introduction	45
4.2 Controllers for Hybrid Input with Wall Displays	47
4.3 <i>SurfAirs</i>	49
4.4 <i>SurfAirs</i> vs bare hand for mid-air input	54
4.5 <i>SurfAirs</i> vs bare hand for both on surface and mid-air input	61
4.6 Limitations	69
4.7 Conclusion and Future Work	70

5	Tangible Interaction for Multi-Display Environments	73
5.1	Introduction	73
5.2	Survey Scope and Methodology	74
5.3	Survey Dimensions	77
5.4	Survey Exploration Tool	88
5.5	Designing for Multi-Display Environments	90
5.6	Limitations	94
5.7	Conclusion and Future Work	94
6	Conclusion and Perspectives	97
6.1	Summary of Contributions	97
6.2	Future Work	99
	Bibliography	103

LIST OF FIGURES

1.1	Examples of large interactive displays.	3
1.2	Examples of tangible user interfaces. On the left, a standalone TUI used to interact with vocal messages. In the center, a TUI used to provide haptic feedback in VR on user faces. On the right, a TUI used as a controller for tablets. Images source: https://vimeo.com/19930744 , [227] and [41].	4
1.3	Examples of different tangible form factors. On the left and in the middle, generic cubic or spherical controllers. On the right, the controller is designed as a human brain proxy. Images source: [79], [64] and [17].	5
1.4	Tangible use in the non-digital on horizontal or non horizontal surfaces. On the left, an antique depiction of the senet game, played on tabletop. Second left, people exchanging paper documents during a meeting. Second right, an investigation board used for crime investigation. Right, a drafting table with an attached ruler. Images source: https://fr.wikipedia.org/wiki/Senet , https://www.pexels.com/photo/colleagues-exchanging-papers-8369218/ , https://www.pexels.com/fr-fr/photo/gens-main-stylo-crime-8369520/ and https://digitaltmuseum.se/021027069178/ritbord	6
1.5	Tangible controllers used on a display surface. On the left, tangibles are used on a horizontal surface in a board game context. On the right, tangibles attached on a vertical display surface to educate students about physics principles. Images source: [211] and [204].	7
1.6	Large vertical screens used with other displays. On the left, tablets are used to control the cursors on the wall. In the middle, the user can position a tangible on a map displayed on a tabletop to display a <i>street view</i> of that location on a vertical display. On the right, users equipped with headsets to augment a wall display. Images source: [43], [51] and [108].	9
2.1	Proxemic interactions for public vertical displays. Both images illustrate how the concept of proxemic interaction enables displaying personal information on a public display. The closer to the display a user is, the more personal the information displayed is. Images source: [50] and [215].	13
2.2	On the left, the Pointables technique [14] enables the manipulation of objects that are not within arms reach. On the right, the Talaria technique [183] makes it possible to perform a continuous movement of a large amplitude by taking some distance from the wall during a drag-and-drop action. Source: [14] and [183].	17

2.3	Examples of instrumentation for users to interact with mid-air bare hand gestures. On the left, 5mm infrared markers attached on the user hand to track precise gestures. On the right, users are equipped with a glove, a hat and a jacket with 10mm infrared markers attached to them to track their body, right hand, head and selected fingers. Images source: [147] and [215].	18
2.4	Examples of tangible controllers for vertical displays. On the left, physical controllers are attached to a handheld device that can be used in a remote control like way to interact with a wall from a distance. On the right, tangibles are used directly on the surface of a wall display. Images source: [110] and [217].	19
3.1	(Left) Two users interacting with <i>WallTokens</i> on a wall display. Each <i>Wall-Token</i> can be manipulated on the surface or attached to it. (Right) Close up of a user attaching a <i>WallToken</i> to the wall.	24
3.2	(Left) A schematic representation of a <i>WallToken</i> 's components, and a real token. (Right) The 9-token final set tested in the Recognition Experiment. Real tokens are laid on a surface textured with 1 cm squares.	26
3.3	Footprints for the nine tokens in our final set. Pink tokens' footprints form equilateral triangles. White tokens are derived from pink tokens by pulling one vertex from 30 mm. Red tokens are derived from pink tokens too by pulling two vertices from 15 mm each in order to make the basis 30 mm larger.	29
3.4	Recognition Experiment: main steps of a trial.	30
3.5	The task under the two input conditions in experiment <i>Walltokens vs Touch Gestures</i> .	33
3.6	(top) Completion time by input \times rotation; (bottom) Completion time by input \times size. Error bars represent the 95% confidence interval. <i>Token</i> is faster than <i>Touch</i> for each rotation and size condition. However, the difference is not significant for the -90° rotation condition as shown in the bar charts ('****' < 0.001, '***' < 0.01, '**' < 0.05, 'ns' otherwise).	36
3.7	Distribution of grades (on 5-point Likert scales) for each input along the dimensions: Comfort, Mental Demand, Occlusion Issues, Perceived Performance and Physical Demand (positive assessments are green and on the right).	39
3.8	Enriching the TUIO protocol to dispatch token-related events.	41
3.9	Application demos: picture classification, map navigation, artistic performance.	42

4.1	Interacting with a <i>SurfAir</i> to drag an object across the wall. 1) The user on the left adjusts the zoom factor in the inset window by rotating the <i>SurfAir</i> on the surface. 2) He then initiates a drag-and-drop action to move that window by sliding the <i>SurfAir</i> on the surface. In order to get a wide viewing angle on the scene and to cover a large distance quickly, he steps backward. Since the <i>SurfAir</i> can transition from surface control to air control, the user can move freely in space while carrying on his drag-and-drop action. 3) He drops the window close to the other user without having to enter her personal physical space.	46
4.2	Interaction states and transitions for a hybrid controller.	48
4.3	<i>SurfAirs</i> are custom physical controllers that users can manipulate both on the surface and in the air. In this scene, three users interact with <i>SurfAirs</i> : i) the user on the left uses a <i>SurfAir</i> as a surface controller, sliding and rotating it to pan and zoom in the inset window; ii) the user in the middle interacts from afar as she needs a wide view to perform a large lasso selection (in red); and iii) the user on the right is adjusting the position of the area to be magnified. He picked the viewport's proxy on the wall, and then stepped back to perform a large movement towards North-West without disturbing other users.	50
4.4	<i>SurfAir</i> prototypes. (Top) <i>Torch</i> supports a torch-like grip. (Bottom) <i>Handle</i> supports a door handle-like grip.	52
4.5	Infrared marker positioning for tracking the position of a <i>SurfAir</i> and the on/off state of its switch for the <i>Handle SurfAir</i> (Left) and <i>Torch SurfAir</i> (Right).	53
4.6	Bare-hand input is implemented as a raycast that follows the direction of the user's index. (Left) Typical posture for Switch=off. (Right) SideTrigger posture for Switch=on.	55
4.7	Experimental task in the <i>Docking</i> phase. Participants have to drag and rotate the <i>Modulus</i> , a blue square of 40 cm side, into the <i>Stimulus</i> , an orange square positioned at 90 cm of the <i>Modulus</i> . A square's orientation is indicated by a green mark in one of the corners. In this example: direction = NE and rotation = +90°.	56
4.8	Pointing time by input \times width.	57
4.9	Error rate by input \times width.	58
4.10	Docking time by input \times rotation.	60
4.11	Surface to mid-air transition techniques: <i>Flick</i> and <i>Grab</i> . With the <i>Flick</i> technique, users slide on the surface before lifting their hand/controller off the surface. With the <i>Grab</i> technique, users activate the switch (Side-Trigger posture or button press) before lifting their hand/controller off the surface.	63

4.12	Task time by input for each task.	66
4.13	Results of Anova tests for task time.	66
5.1	Examples of project illustrating our exclusion criteria. On the left, a human-shaped avatar giving vitals output only. In the center, a controller used on the street without additional display. On the right, unmodified smartphones used to display additional information on tabletop. Images source: [75], [29] and [125].	76
5.2	Years distribution of our corpora. In grey the papers from our query (C_{ACM}) that we excluded; in light blue the papers of (C_{init}) that are not in the query; and in steel blue (C_{filter}).	77
5.3	Display Size	78
5.4	Display Orientation	79
5.5	Display - Controller Distance	80
5.6	Display - Controller Communication	81
5.7	Controller Form Factor	82
5.8	Controller Input/Output	83
5.9	Controller Set	84
5.10	Controller Modularity	85
5.11	Controller Transparency	86
5.12	Based on Existing Controller	86
5.13	User Posture	87
5.14	User Mobility	88
5.15	The Web tool to explore our corpus of papers. The left panel features one facet per dimension, and the main panel on the right lists the references in our corpus. A reference is represented by its textual description and a series of icons that represent the values it has along each dimension.	88
5.16	Pictograms used to represent values along the <i>DisplaySize</i> dimension. (a-c) Pictograms for projects that work with Small, Medium or Large displays. (d) Pictogram for a project that considers both Small and Large displays. (e) Pictogram for a project whose the value of <i>DisplaySize</i> is Unknown.	89

INTRODUCTION

In our data driven era, interacting with large volumes of data has become more common but is still challenging. This is partly due to the complexity of the data sets but also because of the large information space they require. A simple solution to address this space issue is to use bigger displays. Such displays mainly take the shape of high resolution interactive tabletops or walls that tend to move users away from the usual desktop computing context (Figure 1.1). First, users are no longer seated alone in front of their individual screen but rather share a large display surface with others. Second, the interaction space is usually extended, meaning that users can move in space around the display. Those new, large and multi-user spaces thus require designing novel ways of interacting as they create much more complexity than a simple context where a stationary user works with an individual display.

Even though large displays are more common today, HCI researchers are still working on the multiple research questions they raise. Among others, an important research question is to design ways of using concurrently a large shared screen. Users need to share not only the digital space of the screen itself but also the physical space around the screen. For instance, they might need to collaborate and share data or, on the opposite, divide the space and keep some privacy. To perform this, it is usually necessary for the system to identify users, which is a research question by itself. Moreover, as users can move in space and interact from a distance with the digital surface, the system must support an interaction space that consists of both the surface



Figure 1.1: Examples of large interactive displays. On the left, a wall display and on the right a tabletop. Images source: [181] and [114].



Figure 1.2: Examples of tangible user interfaces. On the left, a standalone TUI used to interact with vocal messages. In the center, a TUI used to provide haptic feedback in VR on user faces. On the right, a TUI used as a controller for tablets. Images source: <https://vimeo.com/19930744>, [227] and [41].

and the space around it. In particular, it means that users should ideally have consistent and continuous means to interact both on the surface and in the air around it. These are a few examples of the broad research questions that the HCI literature keeps investigating in the domain of large interactive surfaces.

A promising way to address the challenges that we raise above is to rely on tangible objects to interact with large displays. Design and interaction relying on tangible user interfaces (TUIs) is an extensive and very active research domain with a vast literature. TUIs cover a more or less large set of interactions and devices. The consensus is to consider any user interface where users rely on their physical environment to interact with a digital system as a TUI. Some of them are well known due to the predominant role they actually play. We can cite the computer mouse, invented by D. Engelbart in 1963, or game controllers, the first one called "Paddle" introduced by the game "Tennis for Two" in 1958. Although those input devices have proved very versatile for controlling a wide range of applications or games, they show their limits in the novel contexts of use that mobile devices, large displays or even Virtual Reality headsets have opened. These new contexts of use, where users are not anymore comfortably seated at their desk, have created novel opportunities and challenges for designing TUIs. Challenges are not only related to technology, but also and foremost to interaction and design regarding the form factor tangibles should have and the role they should play in the interaction.

Even if a branch of the HCI literature takes interest in fully custom-made TUIs to create stand-alone appliances [58, 155] as Bishop's marble answering machine [25] illustrated in Figure 1.2-(left), most of tangibles are used in conjunction with existing interactive systems. In both cases, the shapes or form factors that a TUI can take are extremely diversified (Figure 1.2): from very generic geometries like cubes [22, 40, 79] or spheres [64, 176] to realistic proxies [17, 157] (Figure 1.3 - right). The form factor of a TUI is usually a key property in the interaction paradigm it supports. They can even take the shape of wearable devices [35, 218] or piece of cloth [119, 172] and can also be designed to provide specific haptic feedback using users' own body [68]. From flat



Figure 1.3: Examples of different tangible form factors. On the left and in the middle, generic cubic or spherical controllers. On the right, the controller is designed as a human brain proxy. Images source: [79], [64] and [17].

surfaces to mid-air interaction, the context of use also heavily influences the design of effective TUIs. Some TUIs involve static objects that can be put on desks [16, 55] or that can be suspended [174], while others are mounted on existing devices like smart phones [234] or VR headsets [205, 227] and some are even actuated like robots [168, 173] or float in the air thanks to vibrations [153, 155].

Tangible objects can also play different roles in TUIs. Sometimes they are used to represent data [73, 152, 177], to give visual feedback [75] or to play an aesthetics role [98, 132]. However, most often, tangible objects are used to act as controllers [85, 110, 138]. Those are just a few examples of form factors, contexts and roles that TUI projects can cover, but they highlight the broad scope of TUIs' literature. As manipulating physical objects is part of our everyday life, a TUI often creates affordances as to how to use it (*i.e.*, users' experience with artefacts from their physical environment suggests a mode of operation with tangible objects in the TUI). Beyond this, the physical cues in a TUI bring different advantages. First, the haptic feedback provided by tangible objects can make the system more powerful regarding different aspects. For instance, it can profit to immersion in a VR environment in multiple ways [67, 157, 200], but can also enable eyes free control which can reach a critical importance in certain situations such as driving a car [19, 237]. Second, the physicality of TUIs has proven to allow more precise and faster manipulations in many different contexts [53, 69, 85, 213, 219]. Finally, researchers also found that using TUIs in a multi-user setup can encourage specific collaboration strategies [5, 66] and increase the awareness of what other users are doing [45]. Those advantages have led the HCI community to investigate TUIs in many different domains such as education [59, 141, 201], accessibility [1, 33, 56, 61, 120], health [7, 131, 140], art [34, 164, 231] or social interactions [28, 188, 220].

Within this wide variety of TUIs and contexts, my thesis focuses on TUIs for large vertical displays with a specific interest in tangibles used as physical controllers. However, using a tangible controller in conjunction with an active display brings an important additional question: how to implement communication between the controller



Figure 1.4: Tangible use in the non-digital on horizontal or non horizontal surfaces. On the left, an antique depiction of the senet game, played on tabletop. Second left, people exchanging paper documents during a meeting. Second right, an investigation board used for crime investigation. Right, a drafting table with an attached ruler. Images source: <https://fr.wikipedia.org/wiki/Senet>, <https://www.pexels.com/photo/colleagues-exchanging-papers-8369218/>, <https://www.pexels.com/fr-fr/photo/gens-main-stylo-crime-8369520/> and <https://digitaltmuseum.se/021027069178/ritbord>.

and the display? In such a case, the controller must be recognized and/or tracked to be used as an input/output with an accuracy that is sufficient enough for the type of interaction that it should support. The HCI literature already explored many different ways to tackle this question. Some of the recognition system used with TUs rely on wireless communication [137, 201], RFID tags [78, 96], vision-based mid-air tracking [65, 79], magnetic fields [126, 130] or marker based recognition [114, 238]. Identifying a controller allows not only to tie it to specific uses, giving the ability to create a set of tangible specialized tools [41], but also, if each users has his own set of tangibles, it provides a straightforward solution to identify different user interacting with the same system [217]. This latter property makes tangible controllers particularly useful for large display interaction because it avoids to instrument users with sometimes cumbersome and/or long to configure trackable pieces of equipment [147]. However, even if the final result is a physical object that users manipulate with simple actions, designing tangible controllers for a given setup is more complex than it can seem. In particular, its form factor must be thoroughly thought. Of course, the technical requirements to make it communicate with the display surface and the usage the tangible will be assigned to are the main challenges to solve. However, considerations regarding weight, size and comfort of use during manipulations are key to usability and must be kept in mind all along the creation process of a tangible controller. All of these results in a set of physical constraints on the controller's structure that make its design a non-trivial task.

Using tangible objects while sitting or standing nearby a table is a common habit for most people (Figure 1.4). We can think of lunch times where people pass plates around or meetings where participants exchange paper documents. Board games, which are heavily spread across the world, are also a context where users play with tangible objects on a table. One common point in those games is the physical tokens that players manipulate to give a visual feedback on the state of the game. Those non-

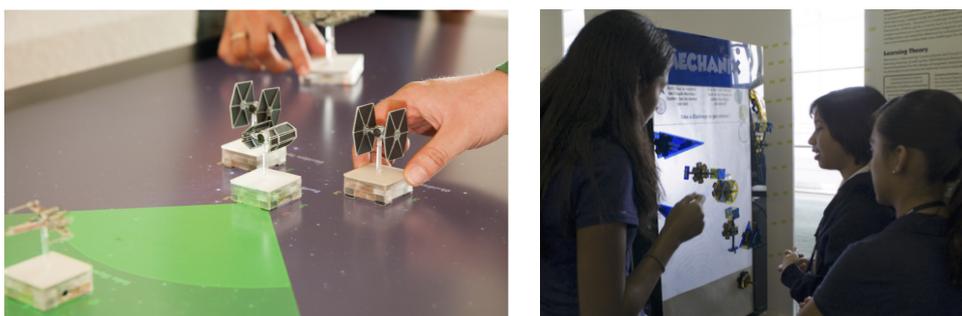


Figure 1.5: Tangible controllers used on a display surface. On the left, tangibles are used on a horizontal surface in a board game context. On the right, tangibles attached on a vertical display surface to educate students about physics principles. Images source: [211] and [204].

digital uses have inspired many projects since the first active tabletop systems became available. Additionally to the analogies they provide, tangible controllers used with tabletops are a good fit as they profit from the fact that the screen is horizontal and flat. It is easy for users to free their hands when they stop interacting as they can just leave in place the tangible they were holding or manipulating (Figure 1.5). However, these simple yet very useful advantages that tabletops offer disappear when surfaces are no longer horizontal. Vertical surfaces like walls suffer from the effect of gravity.

Nevertheless, the non-digital world also features contexts where people do use tangible objects on passive vertical surfaces. Investigation boards, with pins and threads or architect desk but also white boards with magnets, are all examples where physical objects are used on non-horizontal surfaces. All of them have in common the fact that the objects need at one point to be attached to the surface. But when it comes to vertical active screens, pins and magnets (Figure 1.4), which are often used in the non-digital world, cannot be directly transposed to the digital world. Projection-based displays are a solution to remedy this, as the image can be projected on magnetic [126] or porous material, but such solutions typically do not have a resolution as high as an active display and suffer from ambient luminosity problems. This might be a reason why the literature about tangible interaction is abundant when it comes to tabletops, but get scarcer when it comes to vertical displays. Moreover, wall displays and tabletops share a large interaction space but are different by nature due to their screen orientation. Additionally to the problems it creates for tangible interaction, this simple property strongly impacts the way they are used, therefore creating the need to design for each of them specific interaction techniques. While tabletops allow users to gather and move around the horizontal surface, wall displays' verticality enforce users to face them. Users can then get close to it and profit from the high resolution to perform precise manipulations or observe specific data, or take some distance to get a wider view of their workspace, but they always have to face them. To summarize,

creating tangible interaction for wall displays requires not only to compensate the disadvantages verticality brings, but also to ensure that it will not impair the unique interaction space they offer. The goal of this thesis is to provide novel and efficient ways to enable tangible interaction in a wall display environment.

1.1 Research question

The main research question of my thesis is the following: **How to enable tangible interaction on wall displays?** A main aspect to consider when designing interaction with a wall display is the multi-layered space they open to users. On or close to their surface, wall displays can be used for tasks that require precision or a high level of details. From a distance, wall displays give users a wider view of their workspace. Ensuring means to interact in these two spaces (close and distant) but also a fluid transition between them is key for users to take advantage of both spaces depending on the task at hand. Moreover, walls are often used in large rooms equipped with many devices which can be used in combination with the wall to design powerful multi-display setups as in [109], or [65]. These considerations motivate the three more specific research questions of my thesis that I detail below:

- **How to enable *on surface* tangible interaction on a wall display?**

On surface controllers are physical objects that users can manipulate in contact with the digital surface (Figure 1.5). Typical manipulations consist in sliding or rotating such controllers on the surface. However, as mentioned above, using tangibles as on surface controllers for wall displays requires to find a way to leave them in place. On the one hand, it gives the possibility for users to free their hands for other purposes. On the other hand, it makes it possible to indicate a specific location in the digital space with a tangible, which contributes to blending the space between the physical and the digital world. Designing on surface tangibles for wall displays thus involve some technical aspects to continuously track and uniquely identify those tangibles (to enable the creation of a set of tangibles), and design aspects to offer a user experience where tangibles can be both easily manipulated on surface and left in place at will.

- **How to enable continuous tangible interaction from the wall surface to the air?**

Finding a way to enable on surface tangible interaction on a wall display is not enough. To take full advantage of a wall's possibilities, users must also be able to move and interact in the physical space in front of the wall. One of the main challenges that this raises is the ability to track a tangible both on the surface, in the air and during



Figure 1.6: Large vertical screens used with other displays. On the left, tablets are used to control the cursors on the wall. In the middle, the user can position a tangible on a map displayed on a tabletop to display a *street view* of that location on a vertical display. On the right, users equipped with headsets to augment a wall display. Images source: [43], [51] and [108].

the transition between the surface and the air. Ensuring this on surface and mid-air interaction with the same tangible controller is difficult in terms of design as it requires to provide consistent manipulations between the surface and the air as well as enabling a seamless transition between both. The design difficulty does not only come from the definition of interaction techniques, but also from the identification of a good form factor for the tangible, as users need to be able to manipulate it both when in contact with a surface and in the air.

- **How to design a multi-display/multi-purpose tangible controller for interactive rooms that feature more than just a wall?**

A wall display is often used in a multi-display context (Figure 1.6). The wall itself needs a large room where different devices are often available as well. These other devices can take the form of a traditional desktop workstation [139] but also of mobile displays [110], tabletops [65] or AR headsets [109]. Those devices usually complement one another to offer a rich interactive ecosystem where users can take advantage of each device's specificities depending on their needs. Bringing those different devices together in a continuous workspace requires seamless transitions which can be achieved through tangible interactions. However, those devices each have specific characteristics that have an impact on the design of tangible controllers. As having a specialized tangible for each device breaks the notion of continuity between the devices, I believe that it would be interesting to discuss the design of multi-device/multi-purpose tangibles, versatile enough to interact with multiple heterogeneous devices. Designing such a controller is then particularly challenging as it needs to allow a high degree of adaptability to take into account the diversity in such a complex ecosystem.

1.2 Thesis Outline

This thesis is structured into six parts including the introduction and the conclusion. After a related work chapter, I dedicate one chapter to each of my contributions that respectively address the research questions described above.

- First, we describe the WallTokens project that offers a solution for on-surface tangible interaction with vertical displays. A WallToken is an easy to fabricate and passive token whose footprint is recognized on a tactile surface. It is equipped with a push-handle that controls a suction cup. This makes it easy for users to switch between sliding the token or attaching it to the wall. We describe how to build such tokens and how to recognize them on a tactile surface. We report on a study showing the benefits of WallTokens for manipulating virtual objects over multi-touch gestures. This project is a step towards enabling tangible interaction in a wall display context.
- Then, we propose to extend the interaction from the surface to the air with SurfAirs. SurfAirs are physical controllers that users can manipulate on screen (surface input), in the air (mid-air input), and transition from the surface to the air during a single manipulation (hybrid input). We report on two user studies that compare SurfAirs' performance with bare-hand input for both mid-air and hybrid input. Participants prefer and perform better with SurfAirs.
- Finally, based on a survey that we conducted across four ACM conferences, we give an overview of the projects that use tangible controllers with physical displays over a span of 27 years. This survey was motivated by the desire to understand how we could design tangible controllers for multi-display environments containing a wall display. We propose 12 design dimensions to characterize the controllers, the displays and how they communicate, and provide a Web tool to navigate our corpus of references along these dimensions. We then conclude with an analysis of our corpus to discuss the challenges that underlie the design of tangible controllers in a multi-display environment.

RELATED WORK

My thesis work investigates the use of tangible controllers with wall displays. This section, divided into five parts, presents the related work on the topic. The first two parts cover general related work about the benefits of both tangible interaction and vertical displays. The other three parts cover related work that is more specific to the precise research questions this thesis studies. The third part reviews projects about *on surface* tangible interaction for both horizontal and vertical displays. The fourth part focuses on previous work about hybrid interaction, *i.e.*, solutions that combine on surface and mid-air control, in general (*i.e.*, for various types of display surfaces) while the fifth part focuses on physical controllers that have been used for interacting with vertical displays. We will revisit some of this related work at the end of this manuscript, where we discuss the challenges that designing tangible controllers raise when they are considered in environments that combine a wall display with other displays.

2.1 Benefits of Tangible Controllers

For Collaboration

The HCI literature has proposed applications that rely on tangibles for collaborative interaction in many domains such as art [114], education [5], design [86] or document editing [224]. There are also empirical results that support tangibles' advantages in such contexts. For example, in Schneider *et al.*'s study [193], pairs of apprentices in logistics had to design a warehouse on a tabletop using either multitouch interaction or tangibles. In such a high-level, problem-solving task, tangibles were preferred over touch gestures because they offered a better support to the exploration of alternatives. Antle *et al.* also showed that tangibles can be beneficial to the implementation of specific design strategies that enforce or encourage exchanges between users [5, 66]. Finally, Cherek *et al.* report on a study where participants were playing a game on

a tabletop and had to react to others' actions while playing [45]. When performed with tangibles, actions were better detected than with virtual objects, suggesting that tangibles increase awareness of others' actions, which is key to collaboration [60, 82].

For Precise, Space-multiplexed or Eyes-free Control

Tangibles have also been studied at a lower level, demonstrating their efficiency for fine-grained manipulations of virtual objects. In their seminal study, Fitzmaurice and Buxton [69] show that enabling *space-multiplexed* input with multiple specialized tangibles offers more precision than a *time-multiplexed* input device such as a mouse. Later, Tuddenham *et al.* [206] compared the performance of multi-touch and tangible input when interacting with virtual objects displayed on a tabletop. They considered both a simple docking task and a pursuit task that involves four objects as in Fitzmaurice and Buxton's study. In these studies, participants were faster and more accurate with tangibles than they were with multi-touch manipulations. The advantages of tangibles have been further supported by other studies. Voelker *et al.* [213] report on a study where they compared physical knobs and virtual knobs (manipulated with one or two fingers). They found that physical knobs outperform virtual ones in terms of both speed and accuracy, and that one-finger virtual knob performance degrades relatively more when used eyes-free than the other two types of knob (whose performance remained mostly unchanged). Hancock *et al.* [85] propose tangibles with a mounted trackball that provides an additional control dimension. They report on an experimental task that consists of manipulations with multiple degrees of freedom (DOF). Compared to touch gestures, tangibles did not significantly differ for 5-DOF tasks, but were much faster for 3-DOF tasks. Participants also felt more precise when controlling a parameter with tangibles, in particular for the data exploration task that they studied. However, Hancock *et al.* also mention the occlusion issues that can occur with tangibles. Finally, Tangible Tiles [219] is a system that provides a collection of transparent tokens for manipulating digital images on a tabletop. In a task where pairs of participants had to explore images to find hidden features, manipulations with tangible tiles were still less efficient than manipulations of paper images, but they were more efficient than single-touch manipulations.

2.2 Benefits of Vertical Display Surfaces

For Orientation-dependent or Multi-Scale Information Spaces

Some studies have observed the effect of display orientation on users' experience. First, Morris *et al.* [156] have studied the impact of orientation for additional displays in an office environment for single users, and have reported that horizontal displays

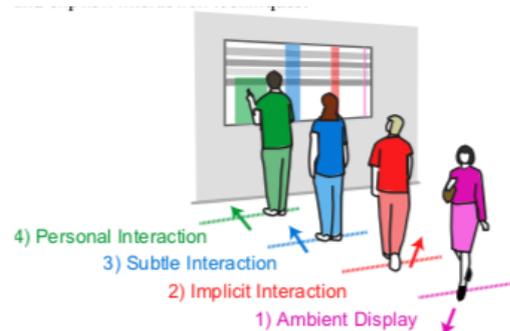


Figure 2. Four interaction phases, facilitating transitions from implicit to explicit, public to personal, interaction

Figure 2.1: Proxemic interactions for public vertical displays. Both images illustrate how the concept of proxemic interaction enables displaying personal information on a public display. The closer to the display a user is, the more personal the information displayed is. Images source: [50] and [215].

can cause ergonomic discomfort for some tasks such as reading long documents in comparison with vertical ones. A vertical orientation is thus preferable in some cases, particularly when the information space has a reference orientation. Inkpen *et al.* [97] support such a dependency between the type of task and the relevance of a given display's orientation. In their study, participants were faster using a vertical display but also find it more tiring. Their participants also really liked the perspective given by the vertical display by moving toward or away from it. This particularity of movement in space opens a design space for proxemic interactions. In proxemic interactions, users' position relative to the display impacts what is actually displayed on the screen. A good example is Ballendat *et al.*'s media player in [13]. In this project, the interface is designed to react to both implicit and explicit interactions. For instance the interface gets more detailed as the user gets closer to it, implicitly inviting to interact with touch Figure 2.1, right. Output can also be modified for privacy purpose as seen in Vogel *et al.* [215] and Coenen *et al.*'s [50] projects. Designed for use in a public space, those projects divide the screen space to provide different level of information to different users. For instance, in [50] which is illustrated on the left of Figure 2.1, one user's personal information is displayed on the bottom display that is visible to that user only as he is close enough to the display to see that information while his body hides it to other, more distant users. However, proxemics can also be used to control input as in Jakobsen *et al.*'s information visualization project [107]. Here, the user's position in space is directly used as input to navigate graph, maps or charts by panning or zooming on the data. In their study, participants "*liked the idea of mapping physical space to data space*" and found that "*using body movement was intuitive or natural*". Changing the zoom level according to user distance seemed useful as it reduced participants effort while allowing smoother interaction compared to mouse control.

For Collaboration

In a collaborative context, Rogers and Lindley [187] conducted a study where teams of three participants had to solve a trip planning task. They compared three conditions: a desktop+mouse control condition, and two larger (~1×1 meter) shared display conditions with a single pen controller that users have to exchange. The two shared display conditions differ in the display orientation: horizontal (table) or vertical (wall). Overall, the horizontal condition seemed to better promote collaboration between participants. A possible reason why users may have preferred the horizontal condition is the ease of both exchanging the pen and switching between paper-based and digital activities. In the vertical condition, participants complained about the fact that there was *no obvious place to put the pen down*. Interestingly, in Potvin *et al.*'s study [179] where each participant had their own pen and could keep holding it, there was no clear difference between the horizontal and vertical orientations for completing a design task collaboratively. These results suggest that tangible artifacts can be cumbersome when working with a vertical display if users cannot easily put them down somewhere. When multiple users collaborate on the same display, the question of how users share the physical and virtual space become crucial. Azad *et al.* [9] showed in their study that, in a collaborative context, the on-screen territoriality of vertical displays is similar to the one on tabletop. Jakobsen and Hornbaek [102] showed that this similarity might be linked to the display size. Their study showed that larger display requires less negotiations for pairs of users as they mostly shared the space evenly. However, when the number of users increases as in the fast-paced game context given in the Miners project by VonZadow *et al.* [217], users tend to ignore social cues and other's interactions. Nonetheless, authors discuss the fact that those awareness and communication problems are strong when users are up-close to the wall but the observations might be different with distant interactions.

2.3 Tangibles for Interacting on the Surface

In 1997, H.Ishii and B.Ulmer present their Tangible bits project which features [99] two interesting prototypes: the metaDESK and the transBOARD. While the metaDESK is horizontal and looks like an interactive tabletop, the transBOARD is a vertical active whiteboard allowing to record users gestures while drawing or writing on it. This seminal work on tangible interaction already envisioned tangible uses both on horizontal and vertical surfaces. However, this project already shows the limitations of tangible when used in combination with vertical surfaces. While the metaDESK provides multiples phicons and tools, the transBOARD is limited to physical cards which can be attached to the surface. This review section focuses on *On*

Surface physical controllers, *i.e.*, controllers that are in contact with the surface, and the impact of surface orientation on their design and use.

Horizontal Displays

With the advent of interactive tabletops, the metaphor with tables from our everyday life quickly inspired using tangible objects to interact with digital tables. The literature in HCI presents multiple projects about tangibles for tabletop interaction. One key challenge to interact with tangible controllers on display surfaces is the recognition technology used. Relying on a vision-based tabletop system, Jorda *et al.*'s Reactable enabled music creation on interactive tabletops with tangible controllers. Those controllers are identified using reactTIVision [18] custom fiducial markers tracked by an infrared camera. Waldner *et al.* [219] also proposed tangible controllers for active tabletops recognized thanks to markers. Those tangibles are transparent, revealing what is displayed underneath. Thanks to their transparency and generic form factor, they can be used as different interactive tools such as an eraser, a magnifier or a magnet. Recognizing tangibles on a tabletop surface can also be done through capacitive technology, by associating a specific configuration of *contact points* to a specific token. While Morales *et al.*'s TouchTokens are made to create a specific configuration of fingers when users hold them on a surface, Chan *et al.*'s Capstones [41] and Voelker *et al.*'s PUCs [212] use conductive materials to create footprints. However, Capstones relies on user touching them to create conductivity while PUCs are standalone, cleverly using the screen itself as ground. Other projects propose either using magnetic fields with Hall sensors [128, 222] or RFID markers [96, 185] to identify multiple markers on horizontal surfaces. The common point of all those projects is the possibility to create as many unique tangibles as the number of capacitive footprints the technology can discriminate. Designing multiple tangibles creates a set of tools to multiplex input in space, which is beneficial to contexts that involve multiple users (multi-user) or multiple tools (multi-tool). However, users are limited by the number of tangibles they can manipulate as they have only two hands and they might have to free their hands for other purposes. On horizontal displays, they can just leave the tangibles in place but, as discussed in the next section, this quickly becomes a problem with vertical displays.

Vertical Displays

Whatever the technology considered, there is no obvious way of making tangibles work as they do on horizontal surfaces with a vertical surface. As discussed above, the main issue to address is that of gravity, which prevents users from dropping tangibles without them falling to the floor.

As mentioned in the introduction, locomotion is key to interaction with wall displays. Users interact with the wall both from up close and from afar [3]. This means

that the design of tangibles for wall displays should consider (at least) the following two themes in Hornecker and Buur's framework of Tangible Interaction [95]: *tangible manipulation* to act on digital information through manipulation of material objects, and *spatial interaction* to take into account users' physical navigation in front of the wall. Carrying multiple controllers in this situation might quickly become cumbersome, hence the need for users to leave them somewhere meaningful.

Distant actions for interacting with a wall display can be performed with mid-air gestures (e.g., [167]) and with physical objects such as personal devices (e.g., [43, 218]) or custom-made objects that are manipulated in the air [17]. However, when it comes to interaction within arms' reach, *surface* tangibles have rarely been considered. The Miners project [217] makes use of tangible+touch interactions for a multi-player game on a wall display. Such interactions proved very engaging for users. However, the use of tangibles remains underexplored, as each player manipulates a single token that is associated with a specific action. Furthermore, each player must always hold the token in their hand. The Vertibles project [90] relies on micro suction cups to attach tangible controllers on a vertical surface. A camera and a projector are used to track the controllers and display information on them. However, the Vertibles are stuck once attached making surface interactions such as slides or rotations impossible. The Geckos project [126] brings a promising solution with a projected screen on a magnetic surface to enable the use of magnet based controllers. Those controllers are recognized thanks to their magnetic footprint and can be easily moved on the display surface. Nonetheless, this solution is unadapted to active high resolution display as their surface can hardly become magnetic. My first project contribution aims at enabling an interaction style where users can not only manipulate tokens on a vertical surface but also *leave them in place* on that surface. This makes it possible for users to attach controllers at locations of interest, to switch between different tangibles or to simply free their hands when needed.

2.4 Interacting on the Surface... and from a Distance

As explained above, wall displays profit from a large but complex interaction space. Interacting on the display surface is not enough and users need to move along but also away from the display. This section focuses on HCI projects that investigate hybrid control that spans the surface and the volume around the surface. The scope of projects in this section is not restricted to tangible interaction and wall displays but rather to the hybrid nature of interaction independently from the type of input and display.

To address issues related to touch input on small-sized screens, several research prototypes (e.g. [32, 37, 113, 121]) augment handheld devices with sensors to track hand input in the air around the device. But the advantages of extending touch to the



Figure 2.2: On the left, the Pointables technique [14] enables the manipulation of objects that are not within arms reach. On the right, the Talaria technique [183] makes it possible to perform a continuous movement of a large amplitude by taking some distance from the wall during a drag-and-drop action. Source: [14] and [183].

air around or above the screen are not limited to small devices. Marquardt *et al.* [143] propose such a *continuous interaction space* above a tabletop, and list the many interaction techniques that it enables to, *e.g.*, interact from up-close or afar, grab out-of-reach objects or perform high-precision manipulations. Prior to the conceptualization by Marquardt *et al.*, projects such as SecondLight [100] had investigated the detection of users' hand and objects beyond the tabletop's surface from a technological standpoint and proposed actual tabletop prototypes. For example, Hilliges *et al.* use this continuous interaction space to enable intuitive manipulations of virtual objects [91]. Banerjee *et al.*'s Pointables [14] project pursue this idea of interacting above a tabletop. Using bi-manual interaction combining touch + mid-air gestures (Figure 2.2-left), they propose an unobtrusive in place manipulation to reach and manipulate distant target on the tabletop. Extended interaction spaces can also be very useful in multi-display environments such as LightSpace [228] where users need both *local power* for interacting with the current display and *remote powers* for interacting with distant displays [162].

Empirical studies show that physical navigation is key when working with large information spaces [11, 12]. It is valued over virtual navigation [104], and even more so for difficult tasks [135]. Hybrid input can also facilitate collaborative work. On the one hand, touch input can help switch between different collaboration styles [103] and handle concurrent access [105]. On the other hand, touch input can also cause physical conflicts or hide content from other users [89, 105]. This advocates for enabling interaction both up close and from afar. Some systems implement hybrid input for wall displays, enabling interaction through either direct touch or mid-air gestures. For instance, Schick *et al.* [190] rely on RGB cameras and computer vision methods to *extend touch* with a raycast along the user's arm. In a similar spirit, Jakobsen *et al.* extend a multitouch wall display with optical tracking to support mid-air raycast [106].



Figure 2.3: Examples of instrumentation for users to interact with mid-air bare hand gestures. On the left, 5mm infrared markers attached on the user hand to track precise gestures. On the right, users are equipped with a glove, a hat and a jacket with 10mm infrared markers attached to them to track their body, right hand, head and selected fingers. Images source: [147] and [215].

They report on studies that compare touch and mid-air input with contrasted observations. Participants preferred touch to interact with small objects, but tended to choose mid-air input over touch for large-scale manipulations or when they needed a large viewing angle. Like the Pointable technique [14] for tabletops or the MirrorTouch public display [158], the wall display in Jakobsen *et al.*'s studies does not integrate touch and mid-air input. Users can rely on either touch or mid-air, but they cannot, for instance, initiate a movement with a sliding gesture on the surface and continue with that same movement in the air like the Talaria technique does [183]. Moreover, while projects such as Liu *et al.*'s Gunslinger [136] or Langner *et al.* [123] divides touch and mid-air interaction between both hands, Talaria profit from uni-manual interaction (Figure 2.2-right). This is particularly interesting as it avoid preempting one of the user hands to enable continuity between the touch and the mid-air interaction spaces. My second project focus on those hybrid input and allows to go beyond surface manipulations using physical controllers by implementing the concept of *Off-Surface Tangibles*, which has been sketched by Cherek *et al.* in the context of tabletops [44]. Those tangible controllers are tracked both on the surface and in the air, and enable interactions that span across the two.

Such hybrid input is particularly important for wall displays as users move physically in front of the display, however all the systems mentioned above rely on bare-hand, device-free techniques. Using bare hand gestures has the great advantage of leaving users' hands free, but it also has downsides. In particular, it suffers from a lack of haptic feedback and requires users to learn specific hand postures to perform actions as simple as mode switching [197] or clicking [216]. Even for something as elementary as pointing, switching between absolute and relative mode is key to cope

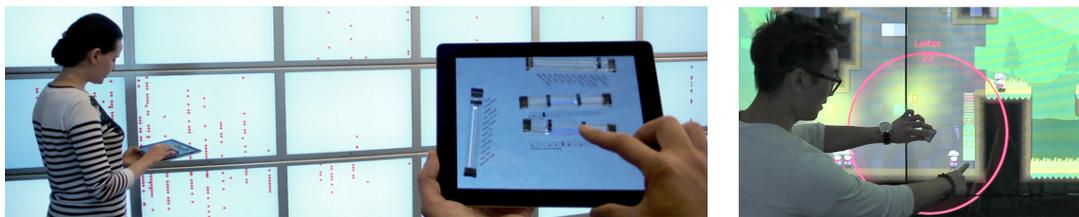


Figure 2.4: Examples of tangible controllers for vertical displays. On the left, physical controllers are attached to a handheld device that can be used in a remote control like way to interact with a wall from a distance. On the right, tangibles are used directly on the surface of a wall display. Images source: [110] and [217].

with the inherent lack of precision and instability of raycasting [160]. In addition to the difficulty of learning and performing postures such as ThumbTrigger [80, 216], SideTrigger [14] or Multirays [147], bare-hand input potentially conflicts with users' movements that are not intended to be interpreted by the system. This is particularly true when users move in space and discuss with each other in a collaborative context. Lastly, projects relying on bare hands gestures [14, 136, 147] or proxemics [13, 107, 215] often use vision based tracking. This requires to instrument the user with sometimes cumbersome or potentially difficult to calibrate equipment (Figure 2.3). Using tangible controllers in such cases allow to shift the tracked markers on the object itself avoiding those difficulties. Moreover, when they want to stop the interaction, users can simply leave the tracked object somewhere without having to remove the tracking equipment they might be wearing. When they want to resume interaction, they just take the tangible controller back while, with a wearable equipment, this might involve a new calibration procedure.

2.5 Physical Controllers for Interacting with Wall Displays

Several research prototypes complement or replace bare-hand input with an off-the-shelf input device. The device can be a basic input device (such as a mouse [104, 167] or a multi-touch trackpad [135]) or it can be a personal handheld device [43, 123, 149, 167, 203]. Relying on an additional device enables efficient implementation of indirect pointing through *e.g.*, an acceleration function and a mode switch between absolute and relative cursor control [149, 166]. In addition to acting as a pointing device, the screen of a handheld device can be used to recognize simple touch actions such as double tap or slide gestures [123]. The device's screen can even host some UI components to invoke commands and adjust parameters in the form of software components [43, 203] or physical components [110] (Figure 2.4-left).

Rather than relying on conventional devices, some projects fall in the category of Tangible User Interfaces with tailored, *ad hoc* tangibles. Tangible User Interaction (TUI) means interacting with digital information through the physical environment. What TUI encompasses is debatable. For example, taken literally, a mouse is a physical object in the environment and in that sense can be seen as a tangible interaction device. However, the HCI community tacitly agrees that TUI does not encompass traditional workstation and input devices. This can be achieved by giving a physical shape to digital information [99], by making everyday objects play an active role in the digital environment [225] or by having one or several physical, specialized controllers that can multiplex input or output in space [69, 114, 206]. This latter approach is particularly relevant for environments that involve a large interaction space. Moreover, the feedthrough provided by the manipulation of physical artefacts can help increase group awareness and facilitate collaboration [163].

Regarding mid-air interaction, tangibles have received more attention for immersive environments [10, 24, 52, 64, 65, 74, 118] than for wall displays. In the specific context of wall displays, tangibles have been almost exclusively proposed for interaction on the display itself (Figure 2.4-right). The seminal pick-and-drop technique [184] allows users to tap an object with a pen to pick it, and then tap again to drop it elsewhere. As one pen is associated with one user, pick-and-drop multiplexes input in space, enabling several users to perform concurrent manipulations. In the *Seconds matter* project [72], Fraser *et al.* study the transition from off-screen space to on-screen space for pen input on a digital whiteboard. In their system, two users (say U_1 and U_2) collaborate synchronously but are located in two distant sites. The location of U_1 's pen relative to U_1 's whiteboard is displayed on U_2 's whiteboard so that U_2 can anticipate where interaction will take place and thus better synchronize their own actions with that of U_1 . Several projects have proposed other types of tangibles for surface manipulations on vertical displays. For example, in Miners [217], users rely on touch input combined with small tangibles in the form of tokens that are similar to TouchTokens [154]. A tangible in the Miners game identifies a user. When in contact with the wall, it delineates an area where touch input is associated with that user. Some tangibles can also be attached to the wall [90, 126] and used as tangible widgets in the spirit of what SLAP widgets enable on tabletops [223]. WallTokens [53] are tangibles that can be both attached to, or slid over, the surface. They can thus be used as either widgets or controllers. Empirical observations reveal the advantages of such tangibles over touch gestures in terms of comfort and speed when manipulating virtual objects displayed on a surface, either horizontal [85, 206] or vertical [53].

Relying on a physical controller for interacting with wall displays can address issues related to bare-hand input. Physical controllers can be organized into two categories: personal devices (i.e. tablets or smartphones) or *ad hoc* tangibles. While using a personal device to interact with a wall display enables powerful interaction tech-

niques, it also means constantly holding this personal device. In addition to the fatigue this can cause, users do not have their hands free, which can be annoying when interacting with other artefacts as well as when performing bi-manual actions on the wall itself. Also, as mentioned earlier, such device-based interaction is indirect and does not integrate well with direct touch interaction on the wall, that users like to rely on [106].

A couple of research projects equip smartphones with additional sensors to recognize a contact between the smartphone and an external touchscreen [175, 191], enabling users with *phone touch* input. However, the form factor of a smartphone makes such direct *phone touch* actions typically limited to basic taps. On the opposite, tangibles that are tailored to wall displays can support richer surface manipulations on the wall itself such as sliding and rotating, which can even be combined with touch actions [53, 217]. As discussed above, those custom tangibles can even be designed so that users can attach them to the wall to free their hands for other actions [53, 90, 126].

2.6 Summary

This chapter is a review of different projects linked to my thesis research question. However, while the literature demonstrate that both large vertical displays and tangible user interfaces improve users' experience in multiple contexts, the intersection of these two domains is rarely explored. The main challenge resides in combining both large wall displays' wide interaction space and the haptic advantages of tangible controllers without impairing one or the other.

ENABLING *ON SURFACE* TANGIBLE INTERACTION WITH WALL DISPLAYS

As detailed in the previous chapter, the interaction space with a wall display is not limited to its surface. However, as a first step, we started this PhD work with an in-depth investigation of *on surface* interaction for wall displays. While many projects have investigated tangible interaction on the surface of active horizontal displays, very few projects have already investigated *on surface* tangible interaction on active vertical displays. The only projects that addressed this topic actually rely on attached static controllers [90, 204] that feature quite constrained manipulations such as turning a knob. In this project, our goal is to contribute surface tangible controllers that users can actually slide and rotate on a vertical display as they can usually do on horizontal displays.

For this purpose, we designed *WallTokens* which are tangibles that can be easily and comfortably attached or detached from the vertical display surface. A *WallToken* is an easy-to-fabricate tangible. It is a passive token whose footprint is recognized on a tactile surface. It is equipped with a *push-handle* that controls a suction cup. This makes it easy for users to switch between sliding the token or attaching it to the wall. We describe our design and fabrication process as well as our approach to track and recognize such tangibles. We report on a study showing the benefits of *WallTokens* for manipulating virtual objects over multi-touch gestures. We finally illustrate their use with three demo applications, and make our development framework available to the community.

This chapter is based on a full paper published at CHI '21 [53]. Supplementary material for this project, including a video, is available at <https://walltokens.lisn.upsaclay.fr/>.



Figure 3.1: (Left) Two users interacting with *WallTokens* on a wall display. Each *WallToken* can be manipulated on the surface or attached to it. (Right) Close up of a user attaching a *WallToken* to the wall.

3.1 Introduction

Large, ultra-high resolution displays (*wall displays*) make it possible for one or several users to interact with large volumes of data. Unlike horizontal screens such as tabletops, vertical displays make it possible for an audience to observe a scene from roughly the same perspective, which is particularly important when that scene has a preferred orientation – *e.g.*, text documents or maps. Users can also explore that scene at different levels of detail through physical navigation (stepping close to see the details, stepping back to gain an overview). However, designing efficient interaction techniques for large vertical displays is particularly challenging. Distant interaction can rely on mid-air gestures [167] or personal devices used as remote controllers [43, 110]. Close interaction can rely on touch gestures (*e.g.*, [134]). The latter have the advantage of leaving users’ hands free of any controller, but they also lack precision and hardly scale to concurrent interactions from multiple users.

Surface tangibles can enrich interaction from close distance with a large surface. As opposed to tangibles that are tracked in the air (with *e.g.*, an external optical system), surface tangibles get tracked by the system when they are in contact with the surface. They enable precise manipulations [69, 85, 206], support eyes-free control [110, 213], can encourage specific collaboration strategies [5, 66] or increase awareness of others’ actions [45]. Such tangibles have been considered to interact with horizontal surfaces, *e.g.*, [114, 154, 223], or with specific supporting structures [209]. But they cannot easily be used with large vertical surfaces as multiple tangibles become cumbersome when they cannot be left on the display.

In the non-digital world, people work with tangibles on vertical passive surfaces. Whiteboards with magnets, investigation boards with pins and threads, architect desks with drawing tools are all examples of tangibles on vertical or inclined surfaces.

However, when it comes to vertical surfaces that actively emit light such as LCD or LED screens, the use of tangibles raises issues. Simple tools just fall, while pins and magnets damage electronics in screens.

In this chapter, we introduce *WallTokens* as a means to prototype tangible interaction for vertical surfaces (Figure 3.1). A *WallToken* is a light, passive tangible that is mounted on feet in order to generate a multi-touch pattern (*i.e.*, its footprint) when put in contact with the wall. A *push-handle*, located at the token's center, controls an additional foot which is equipped with a spring and terminated by a suction cup. Pushing this handle brings the suction cup in contact with the surface, attaching the token to the wall. Once attached, the token can be manipulated as a rotary knob, or left as is to free the user's hands for any other action. Users can easily detach a token with a simple lever movement that uses any of the peripheral feet as a pivot. This pulls the suction cup away from the wall, which makes the central foot contract back. Users can thus attach and detach tokens at will, making it easy to switch between different tokens or between barehand gestures and token manipulations.

In this chapter, we detail the design and fabrication process of *WallTokens*, which relies on 3D printing and basic supplies (*i.e.*, springs and suction cups) only. We describe how well they get recognized on our wall display, which is equipped with an infrared frame for detecting multi-touch input. In a study, we compare the performance of tangibles and multi-touch gestures for manipulating virtual objects, showing that participants' experience is better with tangibles than with multi-touch gestures. We finally report on our prototyping experience with a sample of applications that we developed on top of our library for handling token input.

Our main contributions are:

- a solution to enable tangible interaction for vertical displays;
- a low-cost approach to the prototyping of such tangibles;
- a study in which tokens performed better than multitouch gestures for manipulating virtual objects.

3.2 WallTokens

WallTokens are tangibles that are low-cost and easy-to-fabricate, making them ideal for prototyping. They consist of basic supplies (spring, suction cup and felt) and 3D printed parts assembled together. They are passive, designed to be interacted with on multi-touch surfaces. Like some previous projects (*e.g.*, [41, 110, 122, 154]), each *WallToken* is mounted on feet that generate a multi-touch pattern when in contact with a tactile surface. A pattern is specific to a token, making this token recognizable with a pattern matching algorithm.

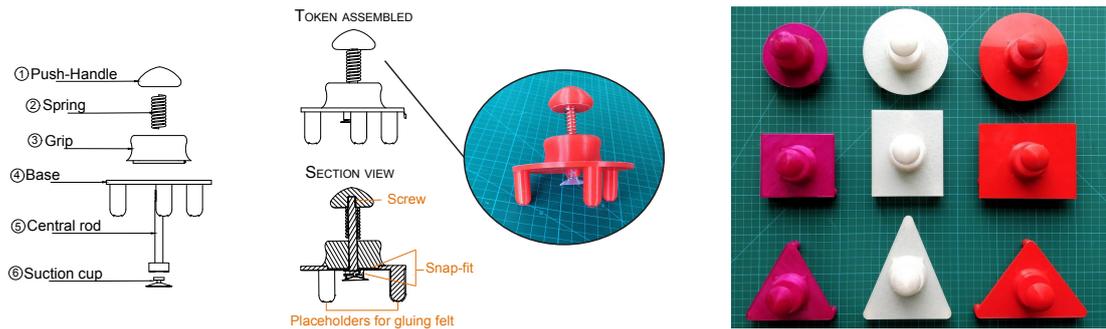


Figure 3.2: (Left) A schematic representation of a *WallToken*'s components, and a real token. (Right) The 9-token final set tested in the Recognition Experiment. Real tokens are laid on a surface textured with 1 cm squares.

Fabrication

A *WallToken* consists of several modular components that are then assembled together by means of screwing and interlocking. Figure 3.2-left details these different components.

- The **base** ④ is the main component. It consists of a plate with three feet underneath (*i.e.*, a single 3D printing job). When in contact with the wall, the three feet will generate the multi-touch pattern (*i.e.*, the token footprint) that is specific to the token. Each foot is 25 mm tall and 12 mm wide. In order to avoid any scratch on the surface, each foot is carved with a placeholder where a piece of felt can be glued (each placeholder is a 4 mm side square of 1 mm depth). The plate is 3 mm thick. Its shape can vary (*e.g.*, square, circle, triangle as illustrated in Figure 3.2-right). The plate not only gives the token a visually identifiable shape, it also prevents users' fingers from getting too close to the screen and thus interfering with the token footprint during the recognition process.
- The **grip** ③ is the knob where users put their fingertips to hold and manipulate the token. It is connected with the base thanks to a snap fit system. The grip is a 20 mm high cylinder of 4.85 mm radius at its base. Its contour is slightly curved to make it comfortable.
- The **push-handle** ① is the top of the token. It is under the palm of users' hand when they hold the token. It is 15 mm long from the base to the tip, and the radius of its circular base is 16 mm. Users push this handle with their palm when they want to attach the token to the surface. To detach a token, users pull its grip. This creates a lever effect around one of the token's feet, making this action easy to perform.

- The central rod ⑤ is a 61.5 mm tall stick that connects the suction cup, which can attach the token to the surface, to the push-handle. By default, the suction cup is not in contact with the surface. The bottom of the central rod is designed as a placeholder where the suction cup can be snapped. The top of the central rod ends with a thread on which the push-handle can be screwed.
- The suction cup ⑥ is a standard 20 mm diameter one, which generates a suction force of approximately 130 newtons. When the central rod is at its default height, the suction cup does not touch the screen, remaining 15 mm away from it.
- The spring ② maintains the central rod at its default height when the suction cup is not attached to the wall. It has 10 coils and is made of steel (AISI 304L stainless steel). Its length is 25 mm when free, 7 mm when compressed. Its inner diameter is 0.7 mm, and its outer diameter is 0.9 mm.

WallTokens are low-cost, yet robust. They do not require any electronics, but only passive materials. Apart from the suction cup, the spring and the felt, individual components are fabricated with a 3D printer using PLA or Tough PLA filament.¹ The assembly time for a token is less than a minute once the felt has been glued under the feet. We ran some informal tests to assess how long tokens can stay on the wall with such a design. Our tests revealed that they do not fall off for at least twelve hours when the wall is off, and for at least three hours when it is on (heat has an impact on how well the suction cups stick). *WallTokens* also proven quite robust against repetitive manipulations. In particular, we used the same unique token for the experiment that we report in [section 3.3](#). However, in case a component gets broken, the modularity of the fabrication process makes it easy to replace the damaged part only.

Finally, modularity also makes it easy to test different token appearances, thus compensating for the lack of flexibility tangibles usually suffer from when it comes to customizing their appearance (*e.g.*, their shape or color) [193]. For example, interaction designers can test different *bases* during the design phase of an application. Having designed tokens ourselves, we also found modularity very convenient for deciding on ergonomic details. In particular, we have tested several alternatives for the *push-handle* and *grip* components before choosing the ones that were the most comfortable during manipulations. SDF files for fabricating the different components are available as supplemental material. Others can easily use them as is, or edit them to test alternative shapes for the base, grip and push-handle.

¹We used Ultimaker 3/3X 3D printers.

Recognition

As mentioned above, we rely only on multi-touch tracking in order to recognize a token based on its footprint using the pattern matching algorithm from [154]. A token set is represented by a collection of templates, with one template per token. Then, when at least three contact points occur simultaneously,² they are processed with the pattern matching algorithm in order to identify the closest template. Relying on such a recognition strategy means that all tokens' footprints should be different from one another within a token set.

When a token is in contact with the wall, any contact point occurring inside the triangle formed by the token feet triggers the detection of the attached state of that token. The token's state remains set to Attached until a token-up event occurs. Such an event occurs as soon as two token feet are lifted off the surface.

Although *WallTokens* could be made conductive to work with a capacitive surface, very large tactile surfaces usually rather rely on optical tracking for detecting touch input. We designed *WallTokens* to interact with such a very large wall display, which is made of tiled ultra-thin bezel screens. It is equipped with a PQlabs infrared touch frame, which is located 4.5 mm in front of the screens. Detailed specifications for such a technology are not available. We thus had to conduct a series of empirical tests to assess 1) the minimal diameter of a foot to be detected as a touch point (12 mm), 2) the minimal height that feet should have to keep the token base and the suction cup when in its default state out of tracking range (25 mm), and 3) the minimal distance between two feet to avoid getting them merged as a single touch point ($D_{min}=5$ mm).

The tests mentioned above also revealed occlusion issues related to the infrared technology, which impacted our strategy for designing tokens' footprints. A foot, which is both aligned with a second foot along the x-axis and with a third foot along the y-axis, is not detected. In order to minimize the chances for such an issue to occur, the three feet of a token always form an isosceles triangle. Figure 3.3 illustrates feet configurations for the nine tokens in our set. The smallest footprint is a 58 mm side equilateral triangle. It corresponds to the smallest footprint that ensures a minimal distance of D_{min} between two feet when they get projected on the x- or y-axis regardless of the token orientation.

The position of the central rod is set in order to optimize the detection of token states (Attached or Detached). State detection accuracy results from a trade-off between maximizing the distances from the central rod to the tokens' feet and avoiding the potential occlusion issues mentioned above. Depending on the token's feet relative placement, a good compromise for positioning the central rod is either the center of the circumscribed circle of the triangle formed by the token's feet (pink and white tokens in Figure 3.3), or the center of the inscribed circle (red tokens in Figure 3.3).

²Contact points should occur within 200ms, and should be close enough to the recognizer's best match.

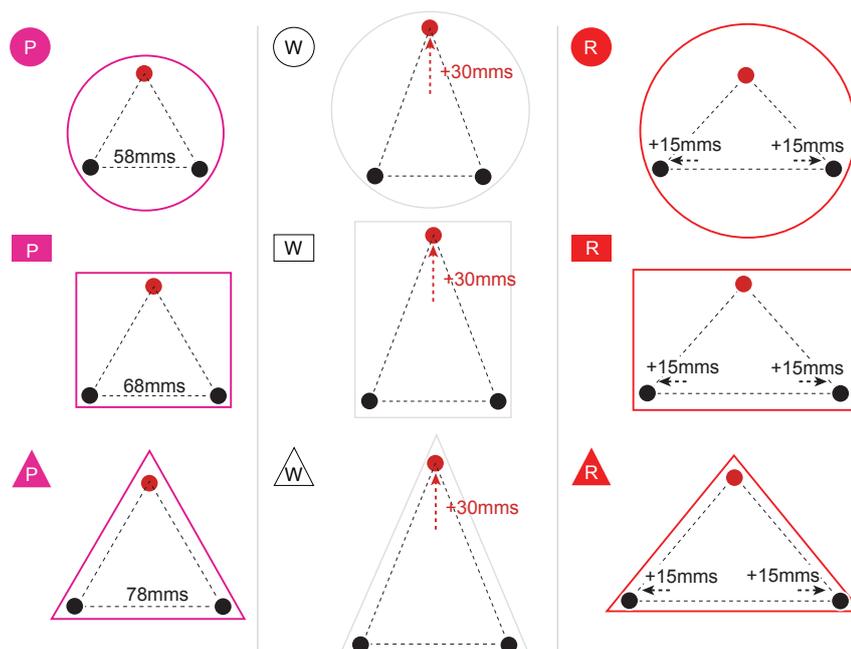


Figure 3.3: Footprints for the nine tokens in our final set. Pink tokens’ footprints form equilateral triangles. White tokens are derived from pink tokens by pulling one vertex from 30 mm. Red tokens are derived from pink tokens too by pulling two vertices from 15 mm each in order to make the basis 30 mm larger.

We ran a small-scale experiment in order to validate *WallTokens’* design. We both consider recognition accuracy of tokens’ identity and detection accuracy of their state (Attached or Detached). In order to get ecological observations, our experimental task collects measures in the context of *pick-and-drop* interactions (*i.e.*, users put a tangible controller in contact at two locations consecutively) and of *detach-then-attach* interactions (*i.e.*, users move a tangible mark from one location to another).

Participants

Because of the COVID-19 pandemics, only the three authors of this submission participated in this experiment: one woman (40 year-old) and two men (27 and 53 year-old).

Apparatus

The experiment runs in full screen mode on a wall-sized display (75 ultra-thin bezel screens tiled in a 15×5 grid, resulting in a total surface of $5m90 \times 1m95$ for a resolution of $14\,400 \times 4\,800$ pixels), driven by 10 workstations³ and equipped with a multi-touch

³Dell workstations with a 3.7 GHz Quad Core Intel Xeon CPU, a NVIDIA Quadro K5000 GPU and 32 GB RAM running Linux.

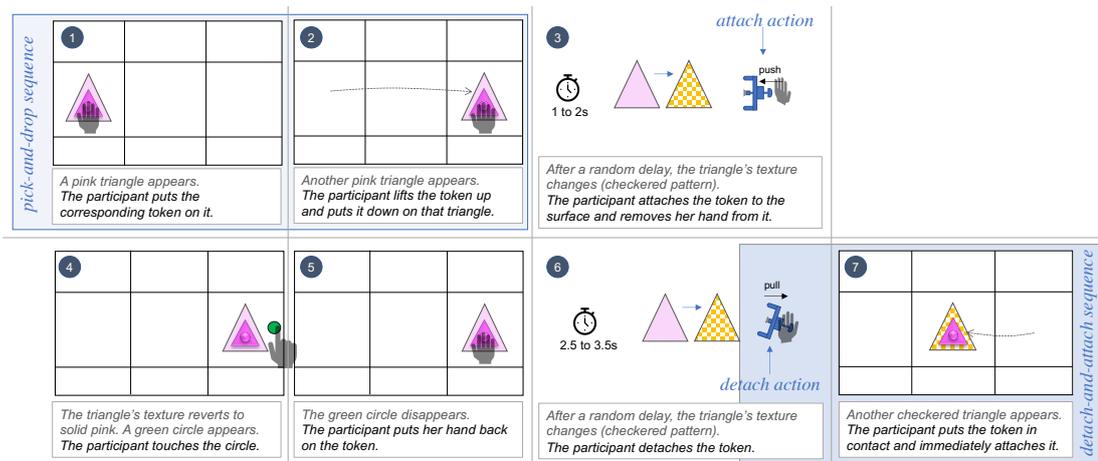


Figure 3.4: Recognition Experiment: main steps of a trial.

PQLabs frame connected to a workstation. The experimental software was developed using Unity 3D (v. 2018.3). The whole setup is orchestrated by a laptop workstation.⁴

Task

As Figure 3.4 illustrates, our experimental task consists of several steps in order to challenge the recognizer in different contexts. **1** A colored shape stimulus is displayed on the wall at 1.25 m above the floor in front of the user. Participants have to grab the token whose shape and color match that of the stimulus (e.g., the pink triangle token in Figure 3.4) and put it in contact with the wall at the stimulus' location. This makes the stimulus disappear, and another stimulus appear 60 cm to the right. **2** Participants have to put the token in contact with the wall at this new location. **3** The stimulus' texture turns into a checkered pattern after a 1-to-2 s random delay. As soon as the texture changes, participants have to attach the token to the wall as fast as possible. **4** - **5** In order to make sure that the token is actually attached to the wall, participants have to release the token to touch a green circle that appears close to the token. They then have to put their hand back on the token to be ready to detach it. **6** The stimulus' fill texture turns into a checkered one after a 2.5 to 3.5 s random delay, instructing the participant to detach the token from the wall as fast as possible. **7** A final stimulus appears 30 cm to the left. It is filled with a checkered pattern as soon as it appears, indicating to participants that they should place the token in this location and attach it as fast as possible.

⁴MSI GE72 2QF laptop with a 2.90 GHz Intel Core I5 CPU, a NVIDIA GeForce GTX970M GPU and 16 GB RAM running Windows 7.

Participant	<i>RecognitionAccuracy</i> (%)	<i>StateAccuracy</i> (%)	<i>AttachTime</i> (ms)		<i>DetachTime</i> (ms)
	① + ② + ⑦	③ + ⑦	③	⑦	⑥
A1	100 ± 0.0	100 ± 0.0	628 ± 14	852 ± 46	797 ± 41
A2	95.4 ± 4.6	98.6 ± 2.7	742 ± 55	1197 ± 106	944 ± 41
A3	98.1 ± 2.5	100 ± 0.0	674 ± 47	880 ± 43	770 ± 26
Mean	97.8 ± 2.6	99.5 ± 0.9	681 ± 65	976 ± 218	837 ± 107

Table 3.1: Results of the recognition experiment.

Design and procedure

Factor and Design. The only factor is *TokenTarget*. It can be one of the nine tokens of our final set (Figure 3.2-right, Figure 3.3): *RedCircle* (R), *RedRectangle* (R), *RedTriangle* (R), *WhiteCircle* (W), *WhiteRectangle* (W), *WhiteTriangle* (W), *PinkCircle* (P), *PinkRectangle* (P), *PinkTriangle* (P). During the experiment, each participant has to complete one block per *TokenTarget*, each block consisting of four replications of the experimental task described above. We use three different presentation orders for the nine blocks, one per participant. This design results in 108 completed tasks in total (3 participants × 9 *TokenTarget* × 4 replications).

Measures. The token recognized by our algorithm is logged at steps ①, ② and ⑦, allowing us to compute accuracy of token recognition (*RecognitionAccuracy*) over $108 \times 3 = 324$ observations. Regarding state detection, the green circle appears after step ③ only if our algorithm actually detects the Attached state. In case it does not, the current task is canceled (with already recorded measures ignored), the operator counts an error (*StateDetectionError*) and restarts the task. *StateDetectionError* is incremented each time such an error occurs. The experiment software additionally logs two time measures: *AttachTime*, which is the time between the first texture change and the Attached state detection that follows (step ③), and *DetachTime*, which is the time between the second texture change and the token up event that follows (step ⑥).

Procedure. Participants stand about 45 cm in front of the wall. The nine tokens are available on a table to their left. The session of tasks begins with a series of nine practice tasks, one task per token. Participants are allowed to redo the practice set until they feel comfortable enough. They then perform the 36 measured tasks.

Results

Table 3.1 reports the overall recognition accuracy, as well as a break-down per participant (A1, A2, and A3) (means and 95%-CIs).

Overall, seven token recognition errors occur among 324 measures. Two occurred at step ① (both for participant A2), five at step ⑦, and none at step ②. There was

no systematic pattern regarding the confusion errors. This prevents us from drawing conclusions regarding the design of the tokens, as errors might as well result from the touchframe itself, which delivers noisy input on some rare occasions. Regarding state detection, there was only one error among 216 measures. This error occurred with the smallest token (PinkCircle), which happened for A2 at step ③.

Time measures suggest that attaching and detaching a token can be performed in less than a second and that it can take an expert user as little as 500 ms to attach a token already in contact with the surface (time at ③ includes reaction time, and time at ⑦ additionally includes a movement), and 650 ms to detach a token (⑥ includes reaction time). Comparing time for attaching at step ③ with time for detaching suggests that it is easier to attach a token than to detach it. This matches our initial impressions.

We acknowledge the limited ecological validity of these observations as they come from the authors themselves. These results cannot be generalized to average users. However, they give an indication of the performance envelope [62], *i.e.*, the performance that expert users can reach.

3.3 Walltokens vs Touch Gestures

WallTokens can act as controllers on a wall display for, *e.g.*, manipulating virtual objects or adjusting parameter values. We believe that they can be an efficient alternative to touch gestures when interacting with a wall from close distance. In this second experiment, we evaluate the performance of *WallTokens* when they are used as controllers on a wall display.

We hypothesize that *WallToken* manipulations are more efficient controllers on a wall display than touch gestures are. First, previous studies have shown that users are more accurate with tangibles than with bare finger input [85, 214]. We thus hypothesize that users will be faster with *WallTokens* than with touch gestures (H_1). Second, we hypothesize that a *WallToken* is more comfortable than touch gestures are for continuous manipulations (H_2). This is not only because of the felt under their feet that reduces friction with the wall, but also because of the possibility for users to reposition their fingers on the token while manipulating it. Finally, we also hypothesize that *WallTokens'* relative advantage will be lower when manipulating small virtual objects which are under their base than when manipulating large virtual objects (H_3). This is because tangibles cause more occlusion than bare hand gestures.

Participants

Twelve volunteers (9 men and 3 women), aged 24 to 44 year-old (average 28, median 25.5), participated in the experiment.

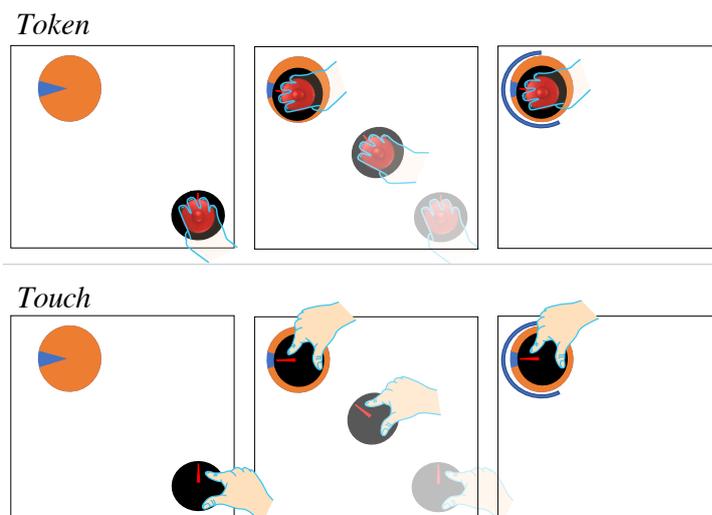


Figure 3.5: The task under the two input conditions in experiment *Walltokens vs Touch Gestures*.

Apparatus

The apparatus is the same as the one described in Section 3.2.

Task

Participants had to perform a docking task, where they had to manipulate a virtual object (*Modulus*) to make its position and orientation match that of a target placeholder (*Stimulus*). They had to perform such a task using either *WallToken*-based manipulations or common multi-touch gestures (Figure 3.5).

The task starts with the two objects displayed on screen: the *Modulus* as a black circle, and the *Stimulus* as an orange circle. Depending on the condition, the participant has to put his fingers or the token on the *Modulus* and drag it over the *Stimulus*. The task ends when the *Modulus* has been maintained for 1500ms inside the *Stimulus* with its orientation (indicated by a red line) matching that of the *Stimulus*. As soon as the position and orientation conditions are met, a blue ring starts to fill up. The ring is full when both conditions have been maintained for 1500ms (dwell), ending the task. The experiment software allows for some tolerance in both orientation and position. The difference in orientation between the *Stimulus* and the *Modulus* should be less than 10° and the distance between their centers should be less than 1cm.

Contrary to Tuddenham *et al.*'s study [206], which also compares multi-touch with tangibles, our experimental task involves a single tangible. This is because we do not advocate for the use of *WallTokens* for spatial multiplexed input where one user would manipulate several objects concurrently with frequent switches between

multiple *WallTokens*. The cost of repetitive attach and detach actions would be too much of an overhead. In a wall display context, we rather envision the use of tangibles as controllers for longer interactions. For example, when each user has their own tangible, or when interactions with a given token are performed in sequence. Our experimental task operationalizes such interactions.

Design and procedure

Factors. Our experiment involves the following three primary factors:

- $\text{input} \in \{\textit{Token}, \textit{Touch}\}$. The *WallToken* used in the *Token* condition was a red 9.4 cm-diameter circle. We chose a circle shape so that the amount of occlusion does not depend on the token's orientation. In the *Touch* condition, users perform 2-finger gestures to control both position (*i.e.*, middle of the 2-finger segment) and orientation (*i.e.*, orientation of the 2-finger segment).
- $\text{size} \in \{\textit{Small}, \textit{Medium}$ or $\textit{Large}\}$. This controls the relative size of the *Modulus* relative to that of the token, and thus the amount of occlusion caused by the token (*i.e.*, the smaller the object, the greater the occlusion caused by the token). *Small* (resp. *Large*) means that the *Modulus*' diameter is half (resp. twice) that of the token, and *Medium* means that the *Modulus* and the token have the same size.
- $\text{rotation} \in \{0^\circ, -90^\circ, 90^\circ$ and $180^\circ\}$ corresponds to the *Stimulus*' orientation.

The fourth factor, *Direction*, is a secondary one that we introduced for ecological purposes. It specifies how the *Stimulus* is displayed relative to the *Modulus* when the task starts. It can take the following four cardinal directions: *NW*, *NE*, *SW* or *SE*. When the task starts, the *Modulus* is displayed within users' arm reach (within a 39 cm square at 1.05 m height), and oriented along the y-axis. Its precise location depends on the value of *Direction* in order to ensure that both objects are displayed in the same screen. This is to avoid users having to cross bezels between the wall's screen cells, as they are an artifact of the specific wall prototype we use in this experiment. The distance between the *Modulus* and *Stimulus* is always 30 cm, but their relative positions depend on *Direction*.

Design. Trials are blocked by input. Each block contains three sub-blocks, one per size. Each sub-block consists of 32 trials, *i.e.*, each *Direction* \times *rotation* combination, and is replicated twice. Presentation order of block is counterbalanced across participants, and presentation order of series of sub-blocks is counterbalanced across participants and input conditions. Within a sub-block, trials are presented in a random order. This design results in 2304 trials in total: 12 participants \times 2 input \times 3 size \times 4 *Direction* \times 4 *rotation* \times 2 replications.

Measures. We collect the following measures: 1) *Time*, the task completion time (*i.e.*, the timer starts as soon as the token or two fingers touch the wall and stops when the 1.5 s dwell ends); 2) *clutchActions*, the number of times the token or participants' fingers leave the wall during the task; and 3) the *Modulus*' position and orientation at each input event.

Procedure. Participants first sign a consent form. They then stand in front of the wall at a distance of about 45 cm from it. The operator gives instructions for completing a task.

Each input block is preceded by three practice sub-blocks, one per size, with each sub-block containing four trials (*i.e.*, four direction \times rotation conditions randomly taken out of the sixteen combinations). Participants can request to do the practice session again if they do not feel comfortable enough. They then complete the 97 logged tasks of the first input block. Before proceeding to the second block, they have to sit and rest their arm until they do not feel any more fatigue. In all cases, the break cannot be shorter than one minute.

At the end of the experiment, participants fill in a questionnaire where they grade on a 5-point Likert scale each input along the following aspects: physical demand, mental demand, comfort, performance and occlusion. They then have to circle their preferred input technique for each size condition and overall. The operator also collects participants' informal feedback during this debriefing phase. The whole procedure lasts about 45 minutes.

Results

Completion Time. We first analyze the effect of our primary factors⁵ on completion time (*Time*) using a repeated measures ANOVA for the model input \times rotation \times size, and Bonferroni-Holm corrected paired post-hoc t-tests. Figure 3.6 illustrates our results.

First, input has a significant effect ($F_{1,11} = 35.5$, $p < 0.001$, $\eta_G^2 = 0.17$), with all participants being consistently faster with *Token*, and *Token* being about 26% faster than *Touch* on average. This result supports H_1 (*i.e.*, users are faster with *WallTokens* than with touch gestures).

We also observe a significant effect of rotation on *Time* ($F_{3,33} = 56.2$, $p < 0.001$, $\eta_G^2 = 0.33$). Unsurprisingly, participants were significantly faster in the translation-only condition (0°) than in all other conditions (-90° , 90° and 180° , all p 's < 0.001). However, there is no significant difference between conditions -90° , 90° and 180° (p 's > 0.4).

⁵The ecological factor direction has no effect on *Time* and no interaction effect with any of the other three primary factors.

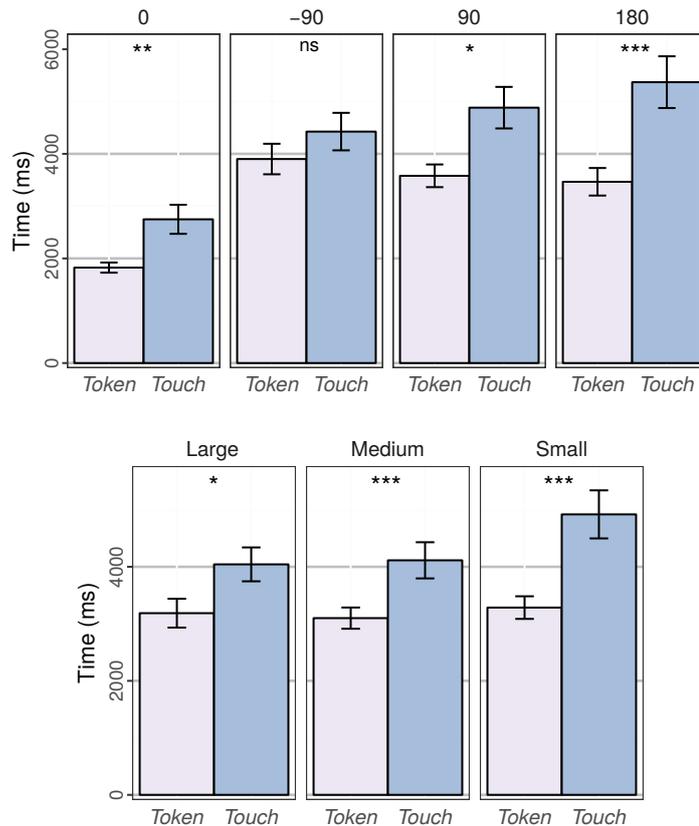


Figure 3.6: (top) Completion time by input × rotation; (bottom) Completion time by input × size. Error bars represent the 95% confidence interval. *Token* is faster than *Touch* for each rotation and size condition. However, the difference is not significant for the -90° rotation condition as shown in the bar charts (‘*’ < 0.001, ‘**’ < 0.01, ‘*’ < 0.05, ‘ns’ otherwise).**

Interestingly, there is a significant input × rotation interaction on *Time* ($F_{3,33} = 6.78$, $p = 0.001$, $\eta_G^2 = 0.04$). As Figure 3.6-(top) illustrates, the -90° , 90° and 180° conditions are not significantly different in the *Token* condition, while participants have been significantly faster with -90° rotations than with 180° rotations in the *Touch* condition ($p = 0.016$, and no significant difference between pairs $\langle 90^\circ, 180^\circ \rangle$ and $\langle 90^\circ, -90^\circ \rangle$). This is consistent with previous experiments (experiment 2 in [235], and lateral condition in [93]) where participants were faster for clockwise rotations (-90°) than for counterclockwise rotations (90°) with touch gestures. This might be because of a lower cost of movement planning [171] for right-handed users for clockwise rotations than for counterclockwise rotations. In comparison, *WallTokens* are less sensitive to differences in orientations. This results in *Token* being faster in all conditions, but not significantly in the -90° condition ($p = 0.072$).

Finally, there is a significant effect of size on *Time* ($F_{2,22} = 7.79$, $p = 0.003$, $\eta_G^2 = 0.03$), with participants being significantly slower with *Small* than they were with both *Medium* and *Large* (p 's < 0.015). This suggests that occlusion caused performance issues for *Small* targets. However, contrary to what we hypothesized, the interaction input \times size is not significant ($F_{2,22} = 2.29$, $p = 0.124$, $\eta_G^2 = 0.01$). Even more surprising, not only were participants significantly faster with *Token* than with *Touch* for all size conditions ($p < 0.001$ for *Small* and *Medium*, $p = 0.014$ for *Large*), but they seem to have been even more relatively faster when they had to manipulate small-sized objects (Cohen's $d = 1.39$ for *Small*, $d = 0.99$ for *Medium* and $d = 0.91$ for *Large*). These results reject H_3 : occlusion issues are not more detrimental with *WallTokens* than they are with touch.

Clutching. Our second measure, *clutchActions*, gets incremented each time either the fingers or the token lose contact with the wall during a task (*i.e.*, they clutch to adopt a more comfortable posture in order to keep on controlling). As collected data do not follow a normal distribution, we use paired Wilcoxon signed rank tests for statistical analyses.

The average number of clutch actions is significantly lower in the *Token* condition than in the *Touch* condition (0.11 ± 0.04 vs 0.88 ± 0.16 , $p < 0.001$), and this difference is consistent across different rotation conditions (p 's < 0.003) and size conditions (p 's < 0.002).

Interestingly, *clutchActions* hardly varies between the different size levels in the *Token* condition (p 's > 0.41), while it significantly grows when size decreases in the *Touch* condition (from 0.68 ± 0.25 to 1.11 ± 0.33 with significant difference between *Small* and *Medium* and *Large*, p 's < 0.027). This increasing need for repositioning their fingers with small objects might explain the relative disadvantage of touch gestures compared to tokens that gets higher in the size=*Small* condition. Similarly, there is no significant difference in *clutchActions* between the different rotation levels for *Token*, while some differences are significant for *Touch*. *clutchActions* is significantly lower for 0° (0.29 ± 0.18) than for the other rotation levels (p 's < 0.004 , 0.84 ± 0.30 for -90° , 1.05 ± 0.39 for 90° , and 1.32 ± 0.36 for 180°), and -90° has also significantly less *clutchActions* than 180° ($p = 0.009$). Here again, this seems to be related to time performance as 0° and -90° are the conditions where participants performed best for *Touch*. Overall, the need for finger repositioning in the *Touch* condition seems to have a negative impact on users' performance. In comparison, *WallTokens* allow users to reposition their fingers individually on the token while keeping it in contact with the wall, allowing greater fluidity in control.

Integrity & Simultaneity. A movement that affects several dimensions (here translation and rotation) is integral if the movement can concurrently modify the value of the different dimensions [101]. Integrity gives an indication of the fluid-

ity in the control of several dimensions, and is thus interesting for comparing touch gestures and token manipulations at a fine-grained level. There is no standard way to measure integrality. Here, we adapt the method from [101] with recommendations from [146] by considering as integral a portion of movement where both dimensions change simultaneously to get closer to their target values. Our *Integral* measure is thus the percentage of movement time during which the differences in position and orientation between the stimulus and the modulus decrease each by more than a given threshold th , or these differences are both very small (*i.e.*, the movement is *stable* as is typically the case during clutching actions and small adjustments at the end of the movement). The steps for computing *Integral* are as follows:

- we consider the movement from the trial start time to the first time the docking conditions are met (dwell start time), and segment it into 10 ms intervals. For each interval, we compute the difference in position and orientation, Δ_{pos} and Δ_{orient} (normalized in $[0, 1]$);
- we smooth data to remove sensor noise (using R's `smooth.spline` function with default parameters);
- we classify each interval as: (i) *integral*: $\Delta_{pos} \geq th$ and $\Delta_{orient} \geq th$; (ii) *stable*: $-th < \Delta_{pos} < th$ and $-th < \Delta_{orient} < th$; (iii) *separate*: neither integral nor stable;
- then, we compute the percentage of stable and integral intervals among all intervals.

With $th = 0.001$ (0.1% of the movement amplitude), the average values for *Integral* are $41.9\% \pm 1.3$ for *Token* and $29.8\% \pm 1.2$ for *Touch*. The difference is significant ($p < 0.001$), as differences per positive rotation condition are. This suggests that *WallTokens* enable manipulations that are more fluid than touch gestures do, allowing users to manipulate position and orientation in an integral movement.

Qualitative Feedback. Figure 3.7 reports participants' qualitative feedback by input. In the final questionnaire, they had to give grades for: comfort, mental demand, occlusion issues, perceived performance and physical demand. We compare these 5-point grades with paired Wilcoxon signed-rank tests.

Token receives better scores regarding comfort ($p = 0.002$) and physical demand ($p = 0.008$). This supports H_2 (*i.e.*, *WallTokens* are more comfortable than touch gestures are for continuous manipulations). *Touch* actually causes more friction and clutching actions than *Token* does. Furthermore, touch detection through optical technology (such as the infrared frame on our wall) can cause more discomfort than *e.g.*, capacitive screens as users may have to adopt hand postures that prevent other fingers from entering the tracking range of the frame. Perceived performance is consistent with actual performance, with *WallToken* outperforming *Touch* ($p = 0.014$). Conversely, grades

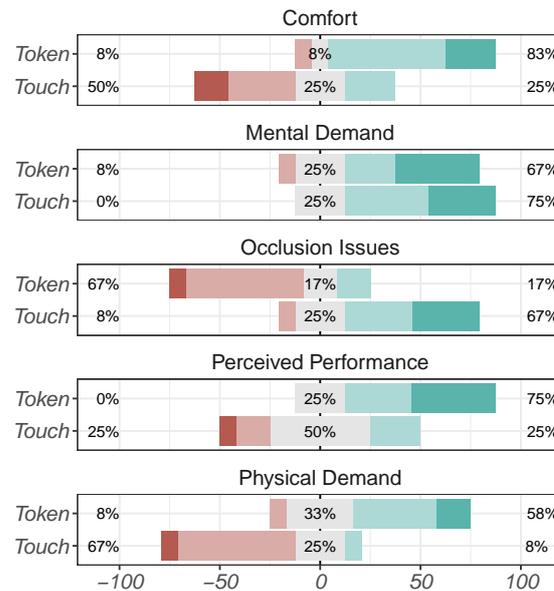


Figure 3.7: Distribution of grades (on 5-point Likert scales) for each input along the dimensions: Comfort, Mental Demand, Occlusion Issues, Perceived Performance and Physical Demand (positive assessments are green and on the right).

regarding occlusion issues are in line with our hypothesis H_3 , with participants reporting tokens to be causing more occlusion than touch gestures ($p = 0.001$). However, they are not consistent with quantitative observations as *WallTokens* did not perform worse than touch gestures with small objects. Finally, participants did not grade any of the two input techniques as mentally demanding ($p = 1$).

Summary. Overall, participants were faster with a *WallToken* controller than with bare hand gestures for manipulating a virtual object displayed on the wall. Participants found touch gestures less comfortable than token-based manipulations, the latter enabling more fluid and integral movements. However, tokens also have some drawbacks in comparison with touch gestures. First, although users' performance did not degrade more with tokens than with touch gestures when the amount of occlusion increases, participants still found occlusion more hindering with tokens than with bare hand manipulations. We are currently working on the design of *WallTokens* with a translucent base in order to address such issues. Second, the touch resolution of our research equipment might be lower than that of smaller, commercial touch devices. The large touch frame, which has been custom-built for our wall display, does not

come with detailed specifications. However, our empirical tests reveal the following limitations: 1) distinct touch points should be distant from at least 5mm in order not to get merged into a single point, and 2) accidental touch events can get triggered because of other fingers that are too close to the wall (less than 4.5mm). It would be interesting to replicate our experiment with other hardware setups.

3.4 Applications

Tangibles can play a lot of roles when interacting with a digital surface. They can materialize users, data and actions to act as identifiers, containers, filters, queries, commands, etc. Many of these different roles are particularly relevant when interacting with a wall display. First, as our experiment above suggests, tangibles can act as controllers when interacting with a wall within arm's reach. They can act as specialized controllers, freeing users from having to learn complex touch gestures. They can also be used for moving content between distant locations using pick-and-drop interactions [184, 219]. Second, tangibles are particularly useful in collaborative contexts. For example, associating each user with a specific tangible is a low-cost and robust way of identifying individual users (as in, *e.g.*, [224]). This user identification method avoids relying on external optical systems, which are usually costly and vulnerable to occlusion issues that frequently occur in an environment where users physically move around. Moreover, tangibles avoid direct touch with the surface, and can thus limit potential sanitary issues when multiple users interact with it. Finally, wall displays are very large surfaces that raise challenges regarding workspace management. Tangibles can help in that regard as they can be used to mark specific positions. For example, they can be used for tagging personal areas as opposed to shared areas, or for bookmarking positions of interest.

Development Framework

To facilitate the development of applications involving *WallTokens*, we have developed a framework based on TUIO⁶ [117], a widely used protocol for programming Tangible User Interfaces. Client libraries for TUIO are available in most programming languages (C++, C#, Java, *etc.*). They connect to an input server and dispatch events according to the TUIO protocol, *i.e.*, in the form of messages that consist of one action among {down, update, up} and the object that triggered this action. The protocol proposes three types of objects: *TuioCursor* (*e.g.*, a touch point along with its position); (ii) *TuioObject* (*e.g.*, a tangible along with its position and orientation); and (iii) *TuioBlob* (*e.g.*, an elliptic shape that often corresponds to the contact area of a *TuioCursor* or a *TuioObject*).

⁶<https://www.tuio.org/>

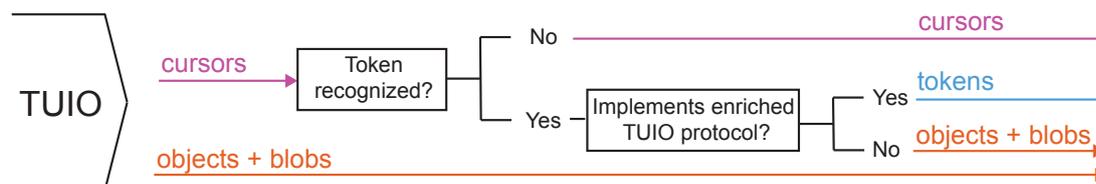


Figure 3.8: Enriching the TUIO protocol to dispatch token-related events.

We developed a C# library that enriches the standard TUIO protocol with a new type of objects, `TuioToken`. The library connects to a standard TUIO server (e.g., the PQLabs driver that runs our wall touch frame). As [Figure 3.8](#) illustrates, it runs the `WallTokens` recognizer each time at least three `TuioCursor-down` events occur simultaneously. In case a token is recognized, it dispatches a `TuioToken-down` event. Otherwise, it passes on the three initial `TuioCursor-down` events. Then, any `TuioCursor-update` event from a cursor that has been recognized as being part of a token is turned into a `TuioToken-update` event until the three cursors leave the surface (which triggers a `TuioToken-up` event). Any `TuioToken` event consists of its action type (`{down, update, up}`), as well as the token’s identity, contour shape, color and state. In addition, each time a cursor down (resp. up) event occurs within the envelope defined by the three token feet, the library dispatches a `TuioToken-update` event to communicate the token’s Attached (resp. Detached) state.

Any C# object can implement the `TuioWallTokenListener` interface to listen to such `TuioToken` events with dedicated callback methods (`addTuioWallToken`, `updateTuioWallToken` and `removeTuioWallToken`). In order to also support any programming language, we developed a `WallToken` TUIO server that any application can connect to in order to receive enriched TUIO events. As illustrated in [Figure 3.8](#), this server can either dispatch `TuioToken` events as described above, or it can downgrade these events into `TuioObject` and `TuioBlob` events, or even simple `TuioCursor` events. Such a downgrade makes it possible to run any existing TUIO-based application with `WallTokens` without modifying its source code. For instance, the `Map` application described in the next section was anterior to the `WallTokens` project. It was developed in Java with input based in part on touch `TuioCursors`. In order to enable `WallToken` input, we first ran the server in downgraded mode so that the `WallTokens` are considered as simple `TuioCursors`. This enabled `WallToken` input without writing any single line of code. We then progressively added callbacks specific to `TuioTokens` in order to handle events such as token rotations and attachments.



Figure 3.9: Application demos: picture classification, map navigation, artistic performance.

Demo Applications

We used the framework described above to develop three demo applications (Figure 3.9): a picture classification application (*Picture*); a map application (*Map*); and an artistic demo (*Art*). The *Picture* application displays a collection of pictures for users to classify by grouping them spatially. Large vertical displays are actually good at supporting the classification, grouping and comparison of visual components [103, 135, 182]. The *Map* application is a multiscale interface that allows users to pan & zoom a large map and instantiate multiple DragMags [221] to magnify specific areas. Finally, the *Art* application displays a water texture on the entire wall and lets users generate waves interactively, creating an aesthetically-pleasing rendering. This art demo was inspired by the ReactTable [114], which combines tangibles and a tabletop into an electronic instrument for collaborative musical performances.

Rather than providing a full description of each of these applications, we list below some of the roles that *WallTokens* play in them:

- **WallTokens** as data containers. In the *Picture* (resp. *Map*) application, a *WallToken* can be used to *pick-and-drop* [184] a picture (resp. a DragMag) from one location to the other. *Pick-and-drop* interactions are particularly important with large displays as dragging an object over a long distance quickly causes discomfort.
- **WallTokens** as controllers. In the *Map* application, sliding a *WallToken* pans the map while rotating it adjusts the zoom factor. Depending on the *WallToken*'s location, such pan & zoom operations apply to the whole map or are restricted to a specific DragMag. In the *Art* demo, users can attach a *WallToken* to the wall to make water drops fall on the large water surface, and then rotate it to adjust the frequency at which drops fall.
- **WallTokens** as cursors. In the *Picture* application, users can perform *drag-and-drop* operations to adjust a picture's position. In the *Art* demo, users can drag a *WallToken* to cut through the water surface and generate waves.

- **WallTokens** as bookmarks or pins. Attaching a *WallToken* to the wall in the *Picture* application lets users change how they populate selections. Once attached to the wall, a finger swipe gesture that is initiated on a picture and oriented toward a *WallToken* actually adds the picture to the selection that is associated with that token. In the *Map* application, attaching a *WallToken* to a *DragMag* locks it. This is useful to, e.g., prevent any other user from interacting with a *DragMag* or simply acts as a salient cue that facilitates later access to it. As mentioned above, in the *Art* application, users can also pin a *WallToken* to create a source of water drops.
- **WallTokens** as identifiers. In the *Picture* application, each *WallToken* is associated with a distinct selection. Users can add or remove a picture from a selection by tapping it with the associated *WallToken*. Performing a zigzag sliding gesture with a *WallToken* clears the associated selection. As a *WallToken* is uniquely identified, multiple selections do not conflict with each other, making it easy to manage several selections concurrently. In the *Map* application, users perform a tap with a *WallToken* on a *DragMag* in order to pair them. Once paired, the *WallToken* acts as a proxy to the *DragMag*, allowing users to manipulate it from anywhere on the wall.
- **WallTokens** as collaborative tools. In the *Picture* application, *WallTokens* can be used to implement a range of multi-user contexts. In a competitive context, each user can have their own set of tokens in order to compare individual performance. In a collaborative context, users can work on different regions towards a shared goal, exchanging tokens if needed. As they are uniquely identified, *WallTokens* also enable concurrent *pick-and-drop* operations from multiple users. A *WallToken* can even be duplicated to enable multi-user pick-and-drop where one user picks a picture and the other drops it [133].

We chose this specific set of demo applications as they highlight the unique advantages of large vertical displays: in the *Picture* application, users can step away from the wall in order to view many pictures at once; in the *Map* application, several users can concurrently work with a map displayed in its preferred orientation; and in the *Art* application, the performer can share their creative space with a potentially large audience.

3.5 Limitations

Expressive power. In our study, *WallTokens* had advantages over touch gestures in terms of efficiency and comfort. But a single *WallToken* has three degrees of freedom only (2D translation and rotation), while touch gestures can control more degrees of freedom (e.g., pinch to scale). When more degrees of freedom are needed, a combi-

nation of touch gestures and tangibles could be interesting (*e.g.*, holding a finger still, close to the token, to switch between rotate and scale modes, or sliding a finger towards or away from the token to control scale). This could avoid users experiencing difficulties when performing multi-touch gestures for the integral control of several dimensions, when they want to act on one dimension without affecting the others [161]. We could also think of more elaborate token designs with *e.g.*, an adjustable cursor as in [206], to increase the number of degrees of freedom that a token can control.

Passive tokens. Our *WallTokens* are fully passive so as to make them compatible with any active display technology. Their mechanism relies on basic supplies only (suction cup and spring). As people are familiar with such supplies, we expect *WallTokens* to afford their manipulations. One first limitation of suction cups is that they work only on smooth surfaces. But this should not be a major issue with most display technologies, as these generally emit light and thus have flat, smooth surfaces to avoid any diffraction effect. A second limitation is that *WallTokens* require an additional action with the push-handle in comparison with approaches based on magnets on a projection-based whiteboard [126]. However, this extra action, which we estimate to take less than 1s in our first experiment, makes *WallTokens* harmless to any active screen technology as opposed to magnets that would damage electronics in screens.

3.6 Conclusion and Future Work

We contribute *WallTokens*, which enable tangible user interfaces on vertical displays. *WallTokens*' low-cost design combined with our development framework make it possible to prototype applications that involve multiple tokens. Not only can users manipulate those tokens on the wall, they can also attach them to, and detach them from it at will. This opens up a design space for applications based on multi-token input that could only run on horizontal surfaces until now (*e.g.*, [112, 114]). As users do not have to keep on holding tokens, our contribution also makes tangible input easy to combine with other input channels such as finger touch or pointing devices on vertical displays. Future work includes the design of such interaction techniques that actually combine tangibles with other modalities. We also plan to refine the design of *WallTokens* to track their position and orientation when users hold them in the air, as recently proposed with the concept of *Off-Surface Tangibles* [44], in order to make it possible for users to rely on *WallTokens* for distant interactions with the wall.

OPENING A CONTINUUM BETWEEN ON SURFACE AND OFF SURFACE TANGIBLE INTERACTION WITH WALL DISPLAYS

While my first project focuses on proposing *on surface* interaction with wall displays, it does not provide a solution for distant interaction. This chapter fills this gap in order to make users take the full benefit of the large space in front of the wall. Some projects in the literature propose solutions that combine *on surface* and *off surface* interactions on tabletops [14] or vertical displays [123, 136]. However, they either rely on bi-manual or multi-modal interactions. In contrast, the solution that we propose focuses on enabling *on surface* and *off surface* with a single, uni-manual tangible controller. This approach actually supports an interaction paradigm where the wall surface and its surroundings are seen as a single and seamless interaction space in order to facilitate continuous actions within it.

This chapter presents *SurfAirs*, my second project. *SurfAirs* are physical controllers that users can manipulate on screen (*surface input*), in the air (*mid-air input*), and transition from the surface to the air during the same, continuous manipulation (*hybrid input*). We report on two user studies that compare *SurfAirs*' performance with bare-hand input for both mid-air and hybrid input. Participants prefer and perform better with *SurfAirs*.

This chapter is based on a full paper accepted for publication at CHI '23 [54]. Supplementary material for this project, including a video, is available at <https://surfairs.lisn.upsaclay.fr/>.

4.1 Introduction

Large vertical surfaces such as high-resolution wall displays allow users to work with a very large information space. Such displays are particularly suited to, *e.g.*, interacting with geographic data [12, 53], performing visual analysis [110, 123, 203], sorting large

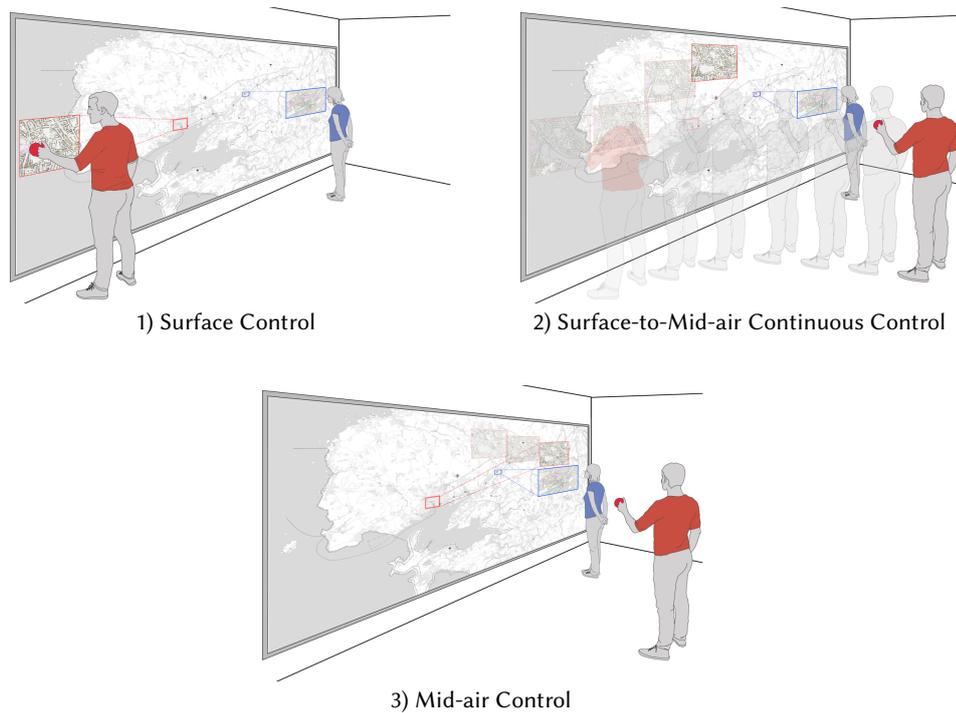


Figure 4.1: Interacting with a *SurfAir* to drag an object across the wall. 1) The user on the left adjusts the zoom factor in the inset window by rotating the *SurfAir* on the surface. 2) He then initiates a drag-and-drop action to move that window by sliding the *SurfAir* on the surface. In order to get a wide viewing angle on the scene and to cover a large distance quickly, he steps backward. Since the *SurfAir* can transition from surface control to air control, the user can move freely in space while carrying on his drag-and-drop action. 3) He drops the window close to the other user without having to enter her personal physical space.

collections of data [103, 134], or even playing games collaboratively [217]. Large vertical surfaces enable two levels of interaction. Users can come close to the display to see details, but they can also step back to get a wider view of the workspace [12, 103]. The interaction space is thus not limited to the display surface itself but also encompasses the physical space in front of it, where users should be able to move freely. Designing input techniques that effectively work across this large interaction space is challenging, however.

Bare-hand input may be considered the most intuitive interaction technique for interacting with wall displays. Users can rely on mid-air gestures [147, 167, 216] to interact from afar, and on direct touch gestures on the wall to interact up close. But while bare-hand input has the advantage of keeping users' hands free, it also raises important challenges in terms of interaction design. In particular, as the design space of

hand movements is unstructured yet constrained and user-dependent, designing bare-hand input that is steady, accurately recognized and that does not collide with regular hand movements is very difficult. Even an action as simple as clicking is challenging to design and implement [14, 216]. As a result, interaction with wall displays often involves a handheld device that serves as a remote controller (e.g. [43, 110, 149, 203, 218]). While this offers a good solution for indirect control, it requires users to always hold a device and it does not integrate smoothly with direct touch interaction on the wall display.

Taking inspiration from projects with tabletops such as [114, 213, 223], some systems rely on tangible controllers to interact with vertical displays [53, 90, 126, 217]. Recent empirical results suggest that such tangible controllers are even more efficient and more comfortable than touch gestures for manipulating virtual objects displayed on the wall [53]. When their design makes it possible for users to attach them to the surface [53, 90, 126], users can free their hands at will. They can also easily switch between multiple tangibles.

However, existing tangible controllers are limited as they support interaction either on the display surface itself or in the air, but not both. This is a strong limitation with vertical displays that afford movements in space to interact either up close or from afar. In this paper, we contribute *SurfAirs*, a new generation of controllers for vertical displays that support various interaction styles. As Figure 4.1 illustrates, users can manipulate them on screen (*surface input*), in the air (*mid-air input*) and transition from the surface to the air during a single manipulation (*hybrid input*). Like WallTokens [53] or Geckos [126], a *SurfAir* can be attached to, and detached from the vertical surface, making it easy for users to free their hands or grab another *SurfAir*.

After a review of related work, we present our contributions: 1) a characterization of hybrid input on wall displays; 2) the prototyping of physical controllers that support such hybrid input; 3) a user study that evaluates their performance for mid-air interaction only; and 4) a user study that evaluates their performance for both surface and mid-air input as well as for transitioning between these two types of input. In both studies, we compare *SurfAirs* with bare-hand input as a baseline. Participants prefer and perform better with *SurfAirs*.

4.2 Controllers for Hybrid Input with Wall Displays

Hybrid controllers can treat the surface and the air in front of it as a *continuous interaction space* [143]. Marquardt *et al.* have introduced this concept in the context of tabletops, demonstrating how such an extended interaction space can not only improve existing interaction techniques but enable novel techniques as well. In this section, we look at hybrid interaction from a lower-level perspective, describing it at the input device level (*i.e.*, the physical controller, not the interaction technique). We

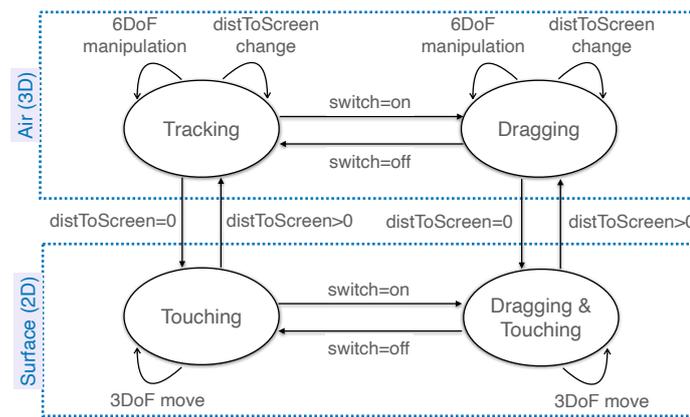


Figure 4.2: Interaction states and transitions for a hybrid controller.

list the low-level properties that *hybrid* (touch+air) brings over *touch only*. To quantify the expressive power of a hybrid controller, we also characterize it in terms of input states and transitions in the spirit of Buxton’s 3-state model [38]. As opposed to a surface controller that typically supports only two states (Out of Range and Touching), a hybrid controller is much more expressive. Figure 4.2 details the four states and multiple transitions within and across those states.

Distance-independent Touch. The most fundamental property that hybrid controllers bring over surface controllers is the ability for users to interact from afar and benefit from a large viewing angle. Enabling users to *touch from afar* requires tracking the position of the controller in the air (3D) in order to implement some raycast from the controller to the wall display (Tracking state in Figure 4.2-Right). It also requires mounting a switch on the controller to change from the Tracking state to the Dragging state without having to touch the wall. The addition of a switch has the positive side-effect of enriching surface interaction as well. When already in contact (Touching), users can activate the switch to enter an additional Dragging & Touching state.

Ideally, users should be able to smoothly transition from the Touching to the Dragging state (and vice versa) to perform continuous interactions while being able to adjust their physical position in space. This typically happens in the case of quasi-modal interactions such as drag-and-drops or area-based selections (rubber band or lasso). It can be because such interactions span a large distance but require precision as well. For instance, users can start interacting on the surface to precisely select a location or object and then step back to get a wider viewing angle and end their interaction far from where it started. Or, reciprocally, they can have to come in contact at the end of their interaction to benefit from the guidance of the surface to drop an object at a precise location. Transitioning from on surface to the air is also sometimes

necessary because of physical constraints. For instance, regions of a large display that are too low or too high can be uncomfortable or impossible to reach while interacting on surface. Similarly, in a multi-user context, some regions of the display might not be available for on-surface interaction without disturbing other users. In such cases, being able to transition from the Touching to the Dragging state enables users to adapt their position and viewing angle without interrupting their ongoing interaction.

From 2D control to 3D control. In the case of a surface controller that must be kept in contact with the surface, users can manipulate that controller along a limited number of Degrees of Freedom (DoF). The surface provides haptic support that facilitates and improves the precision of surface manipulations but it also limits manipulations to translations on the surface itself (2 DoFs) and rotations along the axis that is orthogonal to the surface (1 DoF). On the opposite, when in the air, the controller can be positioned and oriented in 3D, enabling less steady but richer, 6-DoF manipulations (3D translation + 3D rotation). Interaction designers can use all of these degrees of freedom to enable full 3D manipulations. They can also use only a subset of these degrees of freedom to cope with some limitations of surface controllers for 2D manipulations. For example, a surface controller usually does not support scaling operations as the physical object cannot be stretched or compressed [64]. A hybrid controller can enable modal input to activate different virtual actions with the same physical action on the controller. For instance, rotating or scaling a virtual object could be achieved by either rotating the controller on or above the surface.

Distance-to-screen as a parameter. In addition to the absolute 3D positioning of the controller, interaction designers can use the position of the controller relative to the screen as an input parameter. This distance can be used as a *discrete* parameter to create a personal layer to enable interaction with personal data only when close enough to the screen [215], or even a series of discrete layers as in [87]. The distance to the screen can also be used as a *continuous* parameter. For example, distance can control the precision at which an action is performed in the spirit of high-precision sliders [32, 143].

4.3 *SurfAirs*

This section describes the *SurfAir* controllers that we prototyped to implement the concept of hybrid input. As illustrated in Figure 4.3, users can interact with *SurfAirs* from different distances: on the surface itself or at varying distances that provide different viewing angles on the scene.



Figure 4.3: *SurfAirs* are custom physical controllers that users can manipulate both on the surface and in the air. In this scene, three users interact with *SurfAirs*: i) the user on the left uses a *SurfAir* as a surface controller, sliding and rotating it to pan and zoom in the inset window; ii) the user in the middle interacts from afar as she needs a wide view to perform a large lasso selection (in red); and iii) the user on the right is adjusting the position of the area to be magnified. He picked the viewport’s proxy on the wall, and then stepped back to perform a large movement towards North-West without disturbing other users.

Design Requirements

Our main design requirement is to allow users to perform manipulations from close and afar as well as manipulations over varying distances. Our goal is to build a controller that users can *manipulate consistently on the surface and in the air*. We discard solutions that are based on a combination of finger-based input for surface interaction and controller-based input for distant interaction for the sake of a continuous interaction space where users can perform precise and comfortable input. Indeed, a touch+controller solution would either require users to perform some interaction with their non-dominant hand, thus losing precision, or it would require them to move the controller from one hand to the other, thus breaking the continuity of interaction. To address these issues, we target a physical controller that users can manipulate with their dominant hand alone and in a consistent way for different types of input. Such a controller should be both *usable as a surface controller* so that users can perform precise manipulations by taking advantage of the guidance provided by the surface (as the user on the left in [Figure 4.3](#)) and *usable as a mid-air controller that supports eyes-free manipulation* so that users can interact from a distance without having their attention divided between the controller and the action that is taking place on the distant wall screen (as the user in the middle in [Figure 4.3](#)).

In addition to support for hybrid input, designing controllers for large vertical screens entails two additional requirements. First, large vertical displays are often used in a multi-user context. The system should thus support the *concurrent use of multiple controllers*. This not only makes it possible for users to work collaboratively but also for them to use multiple controllers as they would use multiple specialized tools. Second, the controllers should *comply with verticality*. In particular, users should be able to leave them on the surface to either free their hands for other purposes (*e.g.*, switching to another controller or answering a phone call) or just leave the controllers in place for later use.

Design and Fabrication Process

We adopted a modular fabrication process with several components that we assemble together by means of screwing and interlocking. This modular approach makes *SurfAirs* customizable to some extent. During our iterative design process, it had the advantage of making it fast and easy to test and refine the different parts of *SurfAirs*. In our case, it was particularly useful to test different handle designs, but it could also make it easy for customizing their appearance. For instance, creating a set of tokens that have different appearances simply requires designing bases with varying shapes and combine them with a given handle design.

When designing our *SurfAirs*, we first took inspiration from the simple suction cup mechanism for attaching and detaching a tangible on a vertical display that has been recently introduced in the WallTokens project [53]. Like a WallToken, a *SurfAir* is mounted on a base that features three feet and a central suction cup. By default, the suction cup is not in contact with the screen, so that users can slide the *SurfAir* on the surface without experiencing any friction. When they want to attach the *SurfAir* to the surface, they push its handle. This has the effect of bringing the suction cup in contact with the surface to stick the *SurfAir* in place.

A *SurfAir*'s handle is more elaborate than the simple door-knob-like handle of a WallToken. In particular, it features a switch. This requires combining several components, which leads to a different balance in terms of weight. In order to make the whole controller stable when users move, push and pull it, we designed a robust connection between the base and the handle involving rail-guided cylinders (Figure 4.4). Thus, the *SurfAir* remains steady when the user pushes the handle to attach it to the surface. Most of the design process then focused on finding a good trade-off between comfort when pointing and stability when activating the switch. In particular, not all fingers should be involved in the grip's handle so that users can easily free one of them to manipulate the switch. This led us to consider two types of grip.

Our first design, *Torch*, builds on the metaphor of a torch light (Figure 4.4-Top). The thumb is not necessarily involved in the grip and is independent enough to activate a switch. As illustrated in Figure 4.4-Bottom, the cylinder-shape of a *Torch*'s handle

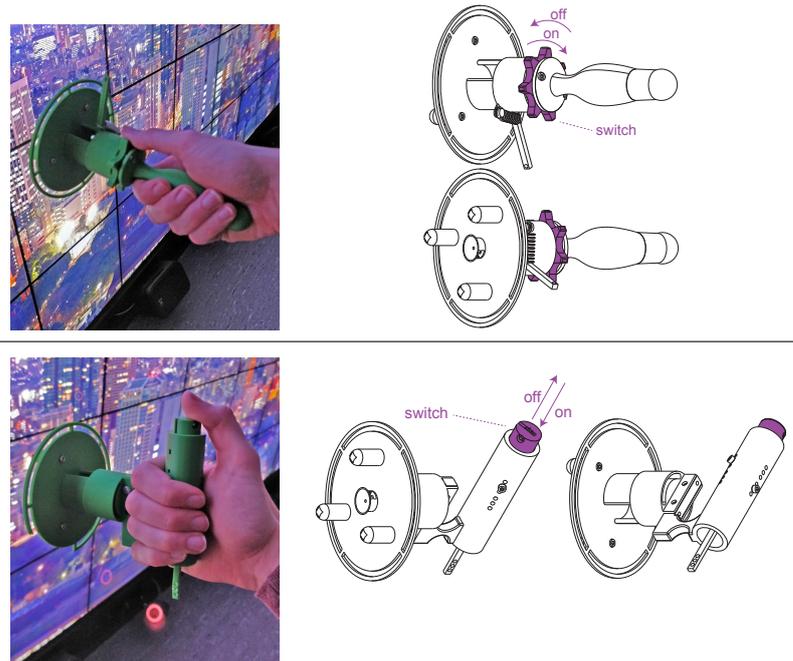


Figure 4.4: SurfAir prototypes. (Top) Torch supports a torch-like grip. (Bottom) Handle supports a door handle-like grip.

is completely symmetric and can be grasped in any orientation. While this gives the user flexibility in how they grab and hold the controller, it also means that the switch must be such that it can be activated regardless of how the user grabs the controller. To ensure that the switch can be activated in any orientation, we designed it as a gear rather than a simple push button. This gear is connected to the base with a spring so that users can rotate it with their thumb tip to switch on or to switch off. The effort that is necessary to activate this gear-like switch is minimal as it requires only a rotation of small amplitude ($\sim 15^\circ$) for the system to detect a change in switch state reliably. The direction of the force that users apply is around the main axis of the controller, which likely minimizes unintentional deviations of the controller when activating the switch. This *Torch* design has thus the advantage of being orientation-independent. In particular, it affords two types of grip. In the air, it seems natural to hold the controller as a torch but, when on the surface, the long handle can also afford a pen-like grip.

Our second design, *Handle*, affords a door-handle-like grip (Figure 4.4-Bottom). Here again, the thumb is not necessarily involved in the grip. We can take advantage of its independence from the other fingers [83, 170] to push a simple spring-mounted button that is located on one end of the handle. Contrary to the *Torch* design, the *Handle* design is orientation-dependent, and affords the same grip for both on-surface and mid-air interaction.

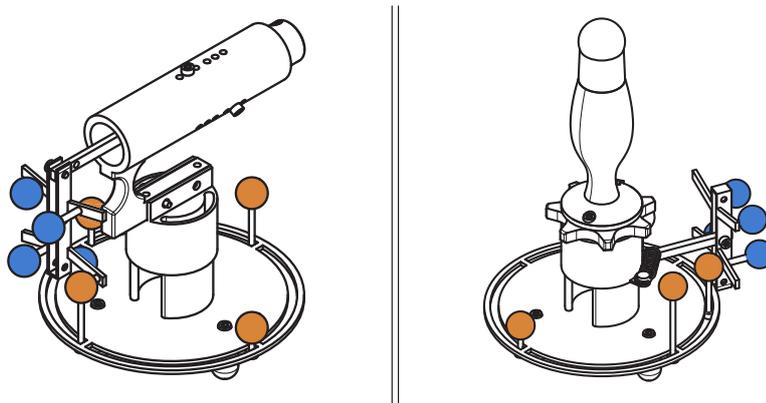


Figure 4.5: Infrared marker positioning for tracking the position of a *SurfAir* and the on/off state of its switch for the *Handle SurfAir* (Left) and *Torch SurfAir* (Right).

Tracking and Recognition

We tested our *SurfAirs* on a wall display equipped with a multi-touch PQLabs[®] frame. The room also features a Vicon[®] motion-capture system with 20 cameras. We chose this motion-capture system not only because it is optimized for our experimental setup (room and display) but also and mainly because it is able to track both *SurfAirs* and bare-hand input, eliminating the tracking technology as a confounding factor in the studies that we report in the following sections.

Like several tangibles for on-surface interaction (e.g., [53, 212, 223]), a *SurfAir* generates a multi-touch pattern when in contact with the tactile surface. Each *SurfAir* is mounted on three feet whose spatial configuration is specific so that a *SurfAir* can be recognized when in contact with a multi-touch surface using the simple pattern-matching algorithm described in [53, 154].

As illustrated in Figure 4.5, *SurfAirs* are equipped with two constellations of infrared markers that we can track with the motion-capture system in the air. The controller's base features a rail where infrared markers can be attached in flexible configurations to define the first constellation (orange-colored markers in Figure 4.5). We rely on this constellation to track the 3D position and orientation of a *SurfAir*, from which we can obtain a raycast. The second constellation of four markers is attached to a movable part of the *SurfAir* (blue-colored markers in Figure 4.5). This constellation's relative position to the base depends on whether the user activates the switch or not. This allows us to recognize the state of the switch (on or off) based on the distance between the two constellations of markers.

4.4 *SurfAirs* vs bare hand for mid-air input

Before evaluating hybrid input, we test the performance of our *SurfAirs* for performing elementary pointing and docking tasks from a distance. We run a comparative study between our two *SurfAir* prototypes and assess their performance using bare-hand input as a baseline. Although this first study does not involve input on the surface itself, our ultimate goal, which we test in the following study, is to support hybrid input. Bare-hand input is therefore the most relevant baseline to consider as it is the only single-handed input technique in the literature that allows for both surface and mid-air interaction while also enabling smooth transitions between those two [183].

In this experiment, participants perform pointing and docking tasks using a controller, which can be one of the two *SurfAirs* or their bare hand, depending on the condition. In all cases, users can point thanks to a raycast originating from the controller/hand, and rotate by adjusting the orientation of the controller/hand. They can select by clicking using either the *SurfAir*'s switch or by changing their hand posture. In the bare-hand condition, we use the *SideTrigger* gesture [14, 106] for clicking (Figure 4.6). Our general hypothesis is that *users will perform better with SurfAirs than with bare-hand input* ($H_{general}$). Although the *SideTrigger* gesture [14, 106] has proven good enough in recent studies to achieve remote selections, we hypothesize that participants will feel *more confident* when activating a mechanical switch than when switching between hand postures ($H_{confidence}$). This is not only because a button constrains the possibilities to two states only, but also because it is less subject to variations across users. In comparison, the space of hand postures is much less constrained and more user-dependent. As a result, the system's recognition and tracking performance will likely be better with *SurfAirs* than with bare-hand input. Second, a *SurfAir* provides haptic support which should help users be more stable when clicking and thus enable *more precision* in selection actions ($H_{precision}$). Finally, we expect the haptic feedback provided by a *SurfAir* to also reduce the sensation of fatigue in comparison with maintaining hand postures ($H_{fatigue}$).

Design and Procedure

Participants. Nine volunteers (8 men and 1 woman), all right-handed, aged 24 to 44 year-old (average 29, median 26), participated in the experiment.

Apparatus. The experiment runs in full screen mode on a cluster-driven wall-sized display (75 ultra-thin bezel screens tiled in a 15×5 grid, resulting in a total surface of $5m90 \times 1m95$ for a resolution of $14\,400 \times 4\,800$ pixels). The experimental software was developed using Unity 3D (version 2018.3).

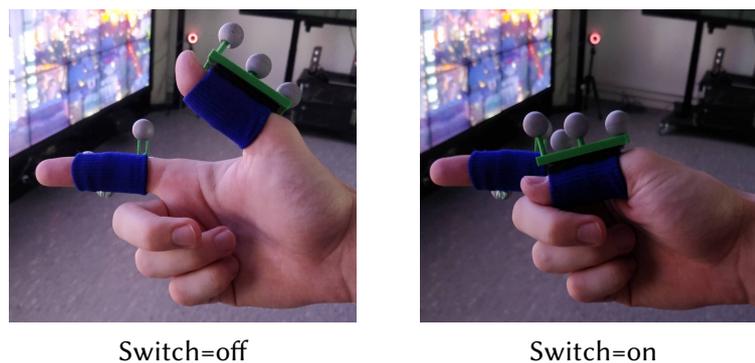


Figure 4.6: Bare-hand input is implemented as a raycast that follows the direction of the user’s index. (Left) Typical posture for Switch=off. (Right) SideTrigger posture for Switch=on.

As illustrated in Figure 4.6, we track hand postures using the exact same hardware and approach as the one we use for *SurfAirs*. In bare-hand input conditions, participants wear two finger sleeves equipped with constellations of infrared markers. We use the distance between these two constellations to recognize the switch between the two hand postures.

General Procedure and Design. We follow a within-subject design for primary factor input = {*Hand*, *Handle*, *Torch*}. The experiment consists of two phases: *Point&Click* and *Docking*, always presented in this order and separated by at least 24, and at most 48, hours. In both phases, trials are blocked by input, and the presentation order of these input blocks is counterbalanced across participants using a Latin Square.

Before starting the experiment, participants have to sign a consent form after the operator has explained the general procedure and goal of the experiment.

At the beginning of each input-block, participants have to put their right foot on a marker on the ground placed at a distance of 2 meters from the center of the wall. The operator then introduces the input technique to be used in this block. In the specific case of *Hand*, the block starts with a calibration procedure in order to account for the variability across different hand anatomies. The operator then explains the task before completing a series of sub-blocks. The first sub-block is for training purposes. As detailed below, the number of measured sub-blocks depends on the phase.

Between each input-block, participants must sit and rest their arm until any feeling of fatigue disappears. During this break, they fill in a questionnaire where they have to rate the input condition that they have just used along the following aspects, with 5-point Likert scales: easiness, confidence, physical demand and mental demand. In

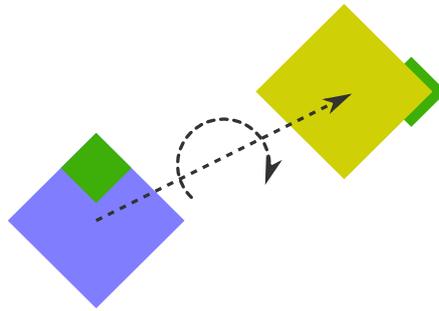


Figure 4.7: Experimental task in the *Docking* phase. Participants have to drag and rotate the *Modulus*, a blue square of 40 cm side, into the *Stimulus*, an orange square positioned at 90 cm of the *Modulus*. A square's orientation is indicated by a green mark in one of the corners. In this example: direction = NE and rotation = +90°.

the *Point&Click* phase, they rate the click gesture and the pointing action separately before giving a final performance rating for the entire task.

At the end of each phase, the participants fill in a global questionnaire where they rank the three different input techniques along the following aspects: physical demand, mental demand, cumbersomeness and preference. The operator also collects participants' informal feedback. Each phase lasts about 1 hour.

Phase 1: *Point&Click*. The first phase is a classic pointing experiment. It consists of clicking a series of 8 circular targets of the same size. The distance between two successive targets in a series is constant. A click outside a target is counted as an error but participants have to continue the task until they successfully select the target. Each input-block consists of 3 sub-blocks, each featuring 6 series of pointing tasks: 2 Dist \times 3 width. Following Jota *et al.*'s recommendations for raycast-based techniques [115], we use angular size and angular distance. Each participant experiences two values for Dist ($\{20^\circ$ and $90^\circ\}$)¹ and three for width ($\{1.8^\circ$, 3.6° and $5.4^\circ\}$).² The presentation order of the 6 series within a sub-block is random. The first sub-block is used for practice.

This design results in 2268 measured pointing tasks in total: 9 participants \times 3 input \times 3 width \times 2 Dist \times 2 sub-blocks \times 7 pointing tasks (the first pointing task is ignored as the cursor's initial location is not controlled).

Phase 2: *Docking*. The second sub-experiment is a docking experiment. As illustrated in Figure 4.7, participants have to manipulate a virtual object (the *Modulus*) to make its position and orientation match that of a target placeholder (the *Stimulus*). A

¹86cm and 490cm on screen.

²For the 20° angular distance, the 1.8° target is 8cm wide, the 3.6° target is 16cm wide and the 5.6° target is 24cm wide. For the 90° angular distance, the 1.8° target is 15cm wide, the 3.6° target is 30cm wide and the 5.6° target is 45cm wide.

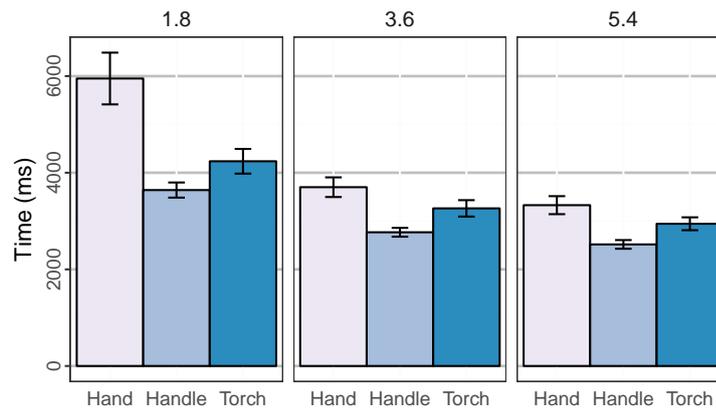


Figure 4.8: Pointing time by input × width.

trial starts with the two objects displayed on screen: the *Modulus* as a blue square in the middle of the screen, and the *Stimulus* as an orange square. The *Stimulus* is placed at constant distance of the *Modulus* (90 cm) in one *Direction* among the following: *NW*, *NE*, *SW* or *SE*. Participants have to drag and rotate³ the *Modulus* over the *Stimulus*. As soon as the position and orientation conditions are met, a blue ring starts to fill up. The ring is full when both conditions have been maintained for 1000 ms (dwell), effectively ending the trial. The experiment software has some tolerance in both orientation and position: 10° in orientation and 5cm (about 1°) in distance. The initial difference in orientation between the *Modulus* and *Stimulus* (rotation factor) is either 0°, -90° (counter-clockwise), 90° (clockwise) or 180°. Each input-block consists of a training sub-block and three measured sub-blocks, each featuring 16 trials (4 *Rotation* × 4 *Direction*) presented in a random order.

This design results in 1296 measured docking tasks in total: 9 participants × 3 input × 4 rotation × 4 direction × 3 sub-blocks.

Results: Point & Click

Pointing Time. We remove 17 points from our collection of 2268 data points: 10 outliers (6 *Hand*, 3 *Torch*, 1 *Handle*) and 7 trials in which participants experienced tracking issues (5 *Torch*, 2 *Hand*). We then run a repeated-measures factorial ANOVA with Greenhouse-Geisser correction for sphericity, and Bonferroni-Holm corrected paired post-hoc t-tests. Figure 4.8 illustrates our results. In all our bar plots, an error bar represents the 95% confidence interval relative to all the data points collected in the corresponding condition.

³To enable wide-angle rotations while remaining within a reasonable range of motion, the *Modulus* rotates twice as fast as the hand or *SurfAir*.

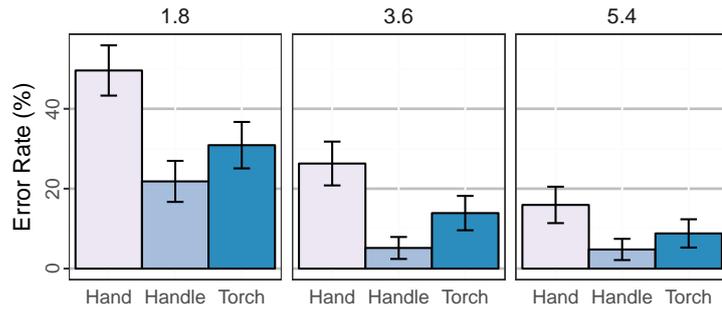


Figure 4.9: Error rate by input × width.

Our primary input factor has a significant effect on pointing time ($F_{1,2,9,7} = 17.5$, $p = 0.001$, $\eta_G^2 = 0.22$), and post-hoc tests show that *Handle* is faster than both *Hand* ($p = 0.006$, $d = 1.48$) and *Torch* ($p = 0.007$, $d = 0.84$), and that *Torch* is faster than *Hand* ($p = 0.007$, $d = 0.81$).

As expected, the ANOVA test also reveals significant effects of both width ($F_{1,1,8,5} = 33.4$, $p < 0.001$, $\eta_G^2 = 0.33$) and Dist ($F_{1,8} = 33.7$, $p < 0.001$, $\eta_G^2 = 0.12$) on pointing time, with participants being significantly faster when width increases (p 's < 0.002) or when Dist decreases ($p < 0.001$). Moreover, we have a significant input × width interaction effect⁴ ($F_{1,3,10,0} = 8.22$, $p = 0.013$, $\eta_G^2 = 0.08$): *Handle* is faster than *Hand* and *Torch* for each width, but *Torch* is significantly faster than *Hand* only when width is small.

Error Rate. We notice an unexpectedly high error rate for *Torch* even for the larger targets (26% for 5.4°-large targets). Looking at the event-level logs, many of these errors with *Torch* occur either at the beginning or in the middle of the pointing movement. We believe that these errors actually reflect tracking issues. Indeed, the switch mechanism of a *Torch* is along the direction of movement and might have been accidentally triggered because of the movement's acceleration. Such errors should not have an impact on pointing performance. In order to focus on errors that can actually impact pointing performance, we filter out errors to consider only those that are close enough to the target ($> \frac{2}{3}$ of the distance) to be actual selection errors. This filtering operation significantly decreases the error rate for *Torch*, and marginally reduces it for *Handle* and *Hand*.

Figure 4.9 shows this corrected error rate. As participants had to continue with the current pointing task in case of an error, it is not surprising to observe effects that are similar to the ones we observed on *Pointing Time*. We observe a significant effect of input ($F_{2,16} = 38$, $p < 0.001$, $\eta_G^2 = 0.28$), with *Handle* having a lower error rate than both *Hand* ($p < 0.001$, $d = 1.95$) and *Torch* ($p = 0.013$, $d = 0.75$), and *Torch* having a lower error rate than *Hand* ($p = 0.001$, $d = 1.05$). Unsurprisingly, the target's width ($F_{2,16} = 28.6$, $p < 0.001$,

⁴Other interaction effects are not significant.

$\eta_G^2 = 0.38$) has a significant effect as well: the error rate decreases as the width increases ($p < 0.001$ and $p = 0.051$). Finally, we observe a significant input \times width interaction effect but with a small effect size ($F_{4,32} = 3.77$, $p = 0.013$, $\eta_G^2 = 0.04$).

These results support $H_{precision}$: *SurfAirs* enable higher selection precision than bare-hand input does. Looking at the movement deviation during a click action, we observe that participants were actually more stable when clicking with a *SurfAir* than when switching between two free-hand postures. The average absolute angle deviation between press and release events of a successful click is $0.45^\circ \pm 0.03^\circ$ for *Handle*, $0.58^\circ \pm 0.03^\circ$ for *Torch*, and $0.73^\circ \pm 0.04^\circ$ for *Hand*, all pairs being significantly different (p 's < 0.025).

Subjective Feedback. There is no significant difference between the input conditions for questions related to the pointing phase (we use paired Wilcoxon signed-rank tests). Overall, participants performed this part of the task with confidence (4.30 ± 0.67), and found it easy to perform (4.22 ± 0.70) with low mental and low physical demand (1.22 ± 0.42 and 2.07 ± 0.92). However, participants found it easier to click with *Handle* and *Torch* than with *Hand* (4.56 ± 1.01 and 4.78 ± 0.44 vs. 2.89 ± 1.17 , $p = 0.031$ and $p = 0.023$). They were also more confident with *Handle* and *Torch* than with *Hand* (4.56 ± 0.73 and 4.44 ± 0.53 vs. 2.33 ± 1 , p 's = 0.012). The differences regarding physical and mental demand were not significant (1.85 ± 1.2 and 1.33 ± 0.73). This resulted in participants feeling that they were performing better with *Handle* and *Torch* than with *Hand* (4.56 ± 0.73 and 4.56 ± 0.53 vs. 2.89 ± 1.17 , p 's = 0.012).

Regarding global rankings, all participants ranked either *Handle* (5 participants) or *Torch* (6 participants, 2 *ex-aequo* ranking) as their preferred technique, and all participants ranked *Hand* last (differences in ranking are significant: p 's = 0.012). Differences in ranking regarding physical demand and cumbersomeness are not significant, but they are significant regarding mental load, with *Handle* and *Torch* ranked better than *Hand* (p 's = 0.012).

Results: Docking

Docking Time. We remove 20 points from our collection of 1296 data points: 6 outliers (3 *Hand*, 2 *Torch*, 1 *Handle*) and 14 trials where participants experienced tracking issues (8 *Torch*, 3 *Hand*, 3 *Handle*). We then run a repeated-measures factorial ANOVA. Figure 4.10 illustrates our results.

The Anova reveals a significant effect of input ($F_{2,16} = 21.4$, $p < 0.001$, $\eta_G^2 = 0.28$) on *Docking Time*. *Handle* is significantly faster than both *Hand* ($p < 0.001$, $d = 1.6$) and *Torch* ($p = 0.037$, $d = 0.77$), and *Torch* is significantly faster than *Hand* ($p = 0.037$, $d = 0.93$). Unsurprisingly, rotation has a significant effect on time as well ($F_{3,24} = 21.8$, $p = 0.025$, $\eta_G^2 = 0.28$), with participants being faster with 0° than with all other rotation angles (p 's < 0.008).

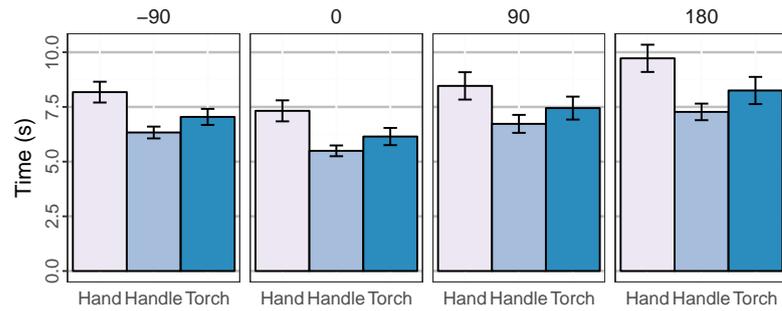


Figure 4.10: Docking time by input \times rotation.

They were also faster with -90° than with 180° ($p = 0.007$). Since the timer starts as soon as the *Modulus* and *Stimulus* appear, *Docking Time* includes both the preparation time (the time taken to grab the *modulus*) and the manipulation time. Analyses on either the preparation time or the handling time in isolation lead to the same conclusions.

Clutching, Integrality & Simultaneity. About 92% of the trials have been performed without clutching, *i.e.*, in “one movement” without releasing the switch since the initial press to grab the *Modulus*. This percentage is high in all three conditions, without any significant differences between them. This suggests that the three techniques are adapted to control two dimensions (translation and rotation) in a single movement.

To better understand how both dimensions are manipulated by the participants, we computed the movement’s *integrality* [101] for trials where rotation is not zero. For this purpose, we divide the drag motion until the instant all docking conditions are met for the first time into a series of 50ms intervals. For each of these intervals, we compute the difference in position and orientation, Δ_{pos} and Δ_{orient} (normalized in $[0, 1]$), these differences being positive if the differences in position or orientation between the *Stimulus* and the *Modulus* decrease. We classify each interval as either (i) *integral* if both Δ_{pos} and $\Delta_{orient} \geq th$; (ii) *stable* if $-th < \Delta_{pos} < th$ and $-th < \Delta_{orient} < th$; (iii) *separate* if neither integral nor stable. Then, we remove the stable intervals, and compute the percentage of the intervals that are integral.

With $th = 0.005$ (0.5% of the movement amplitude), the average integrality score is $44.6\% \pm 1.9$ for *Handle*, $43.5\% \pm 1.9$ for *Torch*, and $36.4\% \pm 1.7$ for *Hand*. The difference between *Handle* and *Torch* is not significant, but both *Handle* and *Torch* have a significantly higher integrality score than *Hand* (p ’s < 0.002 , $d \sim 1$).

Precision. The above results suggest that participants had better control with *Handle* and *Torch* than with *Hand* for both position and rotation in a single movement. Regarding the precision of that movement, we focus on the end of the task and an-

alyze the number of overshoot errors, *i.e.*, the number of times the *Modulus* leaves its docking position after having met the conditions. On average, the number of overshoot errors is 0.47 ± 0.07 for *Torch*, 0.48 ± 0.07 for *Handle*, and 0.70 ± 0.09 for *Hand*, with *Hand* leading to significantly more overshoot errors than both *Torch* and *Handle* (p 's < 0.008 , $d \sim 1$). This suggests that participants have better stabilization and finer control abilities with *Handle* and *Torch* than with *Hand*.

Subjective Feedback. For this Docking phase, participants' subjective feedback is not significantly different across input conditions. Overall, participants performed the task with confidence (4.5 ± 0.59). They found it easy to perform (4.19 ± 0.52), requiring low physical (2.11 ± 0.87) and mental (1.67 ± 0.76) demand. Regarding the global ranking between techniques, six participants ranked *Handle* first, 5 participants ranked *Torch* first, and 1 participant ranked *Hand* first (*ex-aequo* rankings were allowed).

Summary of Results

Overall results of this experiment support our main hypothesis ($H_{general}$): *SurfAirs* perform better and are preferred over bare-hand input. For both *Point&Click* and *Docking*, participants were more accurate with *SurfAirs* than with hand gestures ($H_{precision}$). In particular, participants were more stable when clicking with the mechanical switches of the *SurfAirs* than with hand gestures, which likely contributed positively to the confidence they had when interacting with a physical controller ($H_{confidence}$). Observations in the *Docking* phase also suggest that participants are better at controlling two dimensions concurrently with a physical controller than with their bare hand. Although both *SurfAir* prototypes outperformed bare-hand input, *Handle* seems to be a more promising design than *Torch*. Participants are very stable when clicking in the air with *Handle*, allowing for precise selection. However, while participants' informal feedback suggested more fatigue with bare-hand input than with *SurfAir*-based input, quantitative answers in the questionnaire does not support $H_{fatigue}$: participants did not find bare-hand input more tiring than *SurfAirs*.

4.5 *SurfAirs* vs bare hand for both on surface and mid-air input

In this second experiment, we evaluate *SurfAirs*' performance for tasks that involve both precise manipulations on screen and coarse manipulations in the air. We use a docking task to operationalize manipulations at these two levels of precision, as well as transitions between both. We conduct a comparative evaluation of our *Handle* prototype against bare-hand input (*Hand*). We do not include the *Torch* prototype

in this second study for several reasons. First, we wanted to keep our experiment reasonably short for participants. Second, *Handle* performed slightly better than *Torch* in our first experiment. Finally, as opposed to *Torch*, the grip of a *Handle* is the same whether held on a surface or in the air.

The docking task consists of adjusting the orientation, scale and position of a virtual square (*Modulus*) to make it match the spatial configuration of another square (*Stimulus*). The translation to perform is fairly large (*i.e.*, 160cm), and participants are instructed to interact from afar (*mid-air*) to perform it. On the opposite, they are instructed to interact directly on screen (*surface*) for rotating and scaling the modulus. In order to operationalize the case where users have to do both types of interactions in a single and continuous chunk, participants are also instructed to switch from the surface to the air (or vice versa) during the translation manipulation without releasing control (*hybrid*).

Interaction

Mid-air interaction. As in the first experiment, participants can point with a ray-cast that departs either from the index finger (*Hand*) or from the controller (*Handle*). Participants can grab an object by adopting a *SideTrigger* posture (*Hand*) or by pressing the button (*Handle*) and then drag the object.

Surface interaction. In the *Hand* condition, participants use standard multi-touch gestures: one finger slides the *Modulus* (whether the hand adopts a *SideTrigger* or a released posture), and two fingers both rotate (according to the orientation of the segment defined by the two contact points) and scale (pinch). However, when two fingers are in contact, the translation is disabled. Pilot tests showed that it was very difficult for participants to adjust the rotation and scale without unintentionally moving the object. Participants can still translate on surface by simply lifting one of their two fingers off the surface. Similarly, in the *Handle* condition, the translation is disabled when the button is released. Participants can adjust the *Modulus* orientation by rotating the controller and its scale by sliding the controller up or down. To adjust the *Modulus* position, they must explicitly enter the dragging state by pressing the controller's button.

Mid-air/Surface transition. Users simply put either the controller (*Handle*) or their finger (*Hand*) on the surface to transition from mid-air to surface control. Participants can drag an object mid-air using ray-casting, then come in contact with the surface where they can still drag the object until they release the controller's button (*Handle*) or put a second finger on the surface (*Hand*). They can then continue interacting (rotate and scale).

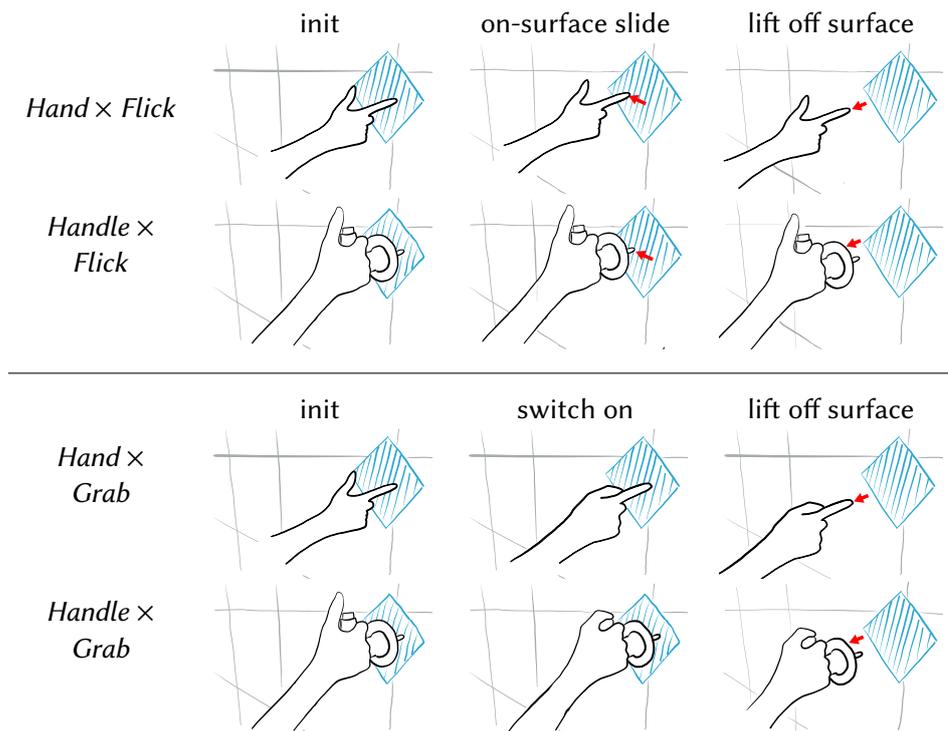


Figure 4.11: Surface to mid-air transition techniques: *Flick* and *Grab*. With the *Flick* technique, users slide on the surface before lifting their hand/controller off the surface. With the *Grab* technique, users activate the switch (SideTrigger posture or button press) before lifting their hand/controller off the surface.

Surface/Mid-air transition. This type of transition is more elaborate as the system must distinguish between releasing control and continuing the ongoing interaction. We consider two techniques, *Grab* and *Flick* (Figure 4.11), to leave the surface without releasing control. With *Grab*, participants have to activate the switch (SideTrigger posture or button press) while being in contact with the surface and maintain the switch on when they leave the surface. They then release control by releasing the switch. With *Flick*, participants must perform a flick gesture (*i.e.*, accelerating the movement as in [183]) when they leave the surface to continue with the ongoing interaction in the air. The flick can be performed with the switch either on or off. They then release control when the switch state changes (on \rightarrow off or off \rightarrow on). This offers more flexibility and avoids relying on clicks, which caused precision issues with *Hand* in our first experiment.

Tasks

Participants had to perform two types of tasks, either transitioning from the air to the surface or the opposite. In both cases, participants are instructed not to release control during the transition. For *AirToSurface*, participants must hold (drag) the *Modulus* when they come in contact with the surface. For example, they cannot drop the *Modulus* over the *Stimulus* using mid-air interaction and then come close to the wall for docking it. For *SurfaceToAir*, once the *Modulus* is docked, participants must use either the *Grab* or *Flick* technique before stepping back.

AirToSurface. The task starts with the *Stimulus* displayed in the center of the wall, and the *Modulus* (a square of 30 *cm* side) located at 160 *cm* on the left or the right of the *Stimulus*. Participants have to face the center of the wall, and be at a distance of at least 150 *cm* from it. Participants have to grab the *Modulus* and drag it over the *Stimulus* using mid-air interaction, while at the same time coming close to the wall to eventually come in contact with it. When in contact, they can precisely adjust the *Modulus* position (tolerance = 7*cm*) and dock the *Modulus* in the *Stimulus*: rotate (tolerance = 10°) and scale (tolerance = 10%). The task is validated once the docking conditions have been maintained for 1s.

SurfaceToAir. The task starts with both the *Modulus* and the *Stimulus* displayed in the center of the wall. Participants have to first dock the *Modulus* into the *Stimulus* on surface (like they do in the second part of an *AirToSurface* task). They then leave the surface without releasing control (using either *Flick* or *Grab* depending on the condition as detailed in the design below). This makes a second *Stimulus* appear on the right or the left at 160 *cm* from the center of the wall. They must drop the *Modulus* over this second *Stimulus* with a tolerance of 7*cm*. The second *Stimulus* appears only when the controller or hand is at 150 *cm* from the wall in order to operationalize the case where users decide on where to position an object only after they have a wider view angle.

Hypotheses

We formulate the following two hypotheses:

$H_{general}$: *SurfAir*-based input outperforms bare-hand input overall. Our first experiment suggests that *SurfAirs* perform better for mid-air input. In addition, studies reported in [53] suggest that tangibles act as better surface controllers than multi-touch gestures. As for the transition, we do not expect to observe strong differences between the two types of input. As a result, we expect a *SurfAir* to perform better than bare-hand input overall.

$H_{transition}$: For transitioning from the surface to the air, we hypothesize that *Flick* is a better technique for *Hand*, while *Grab* is better for *Handle*. This is because 1)

state switching is more costly for *Hand* than it is for *Handle* (as suggested by our first experiment) and 2) a *SurfAir* is more rigid and has a larger contact surface than a finger, which could make the *Flick*'s accelerating gesture more difficult to perform.

Experimental Design and Procedure

Participants & Apparatus. Twelve volunteers (5 men and 7 women), all right-handed, aged 21 to 32 year-old (average 25.33, median 25), participated in the experiment. We use the same apparatus as in the first experiment. The wall display is equipped with a multi-touch PQLabs[®] frame, which we rely upon for on-surface interaction.

Design and Procedure. We follow a within-subject design with primary factor input = { *Hand*, *Handle* }. Trials are blocked by input. With each input, participants had to perform both *AirToSurface* and *SurfaceToAir* types of transition. We also test the two different techniques for transitioning from the surface to the air (tech = { *Grab*, *Flick* }). We thus have three types of tasks: *AirToSurface*, *SurfaceToAir_{flick}* and *SurfaceToAir_{grab}*. Each input-block is a series of three sub-blocks, one per task. The presentation order of blocks and sub-blocks is counterbalanced across participants.

At the beginning of the experiment, participants sign a consent form after having read the general procedure and goal of the experience. Each task sub-block starts with the operator explaining how to perform the task. Participants then perform 16 trials = 2 repetitions (one training, one measured) \times 2 rotations (-90°, 90°) \times 2 scales (-50%, +50%) \times 2 directions (right, left). The presentation order of the 8 trials within a repetition is random.

Between two task conditions, participants must sit and rest. During that break, the operator asks them to rate on a 5-point Likert scale the condition that they have just experienced regarding easiness, confidence, physical demand, mental demand and performance. They rate not only the task in general but also the transition specifically. Moreover, at the end of the experiment, participants rank the two input conditions and the transition techniques.

The whole experiment lasts about 75 minutes. Answering the questionnaire represents about half of this time.

Results

Task Time. Among the 576 measured trials, we remove 9 data points where we experienced logging issues and 7 outliers based on a linear analysis (all from *Hand*). After checking the normality of our data, we run two Anova tests: 1) input \sim Time for *AirToSurface* tasks and 2) input \times tech \sim Time for *SurfaceToAir* tasks. Figure 4.12 illustrates the different effects that we observe, and Figure 4.13 details the test results.

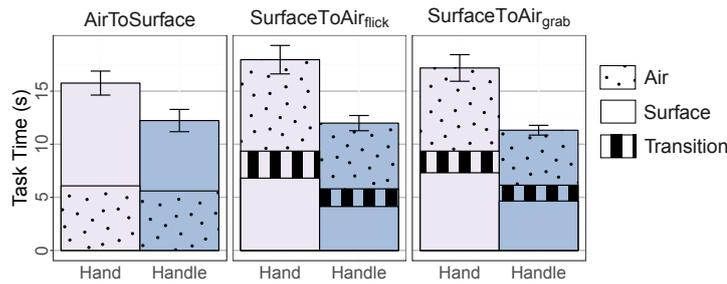


Figure 4.12: Task time by input for each task.

	<i>AirToSurface</i>	<i>SurfaceToAir</i>
input	$F_{1,11} = 22.7, p < 0.001, \eta_G^2 = 0.23$	$F_{1,11} = 70.1, p < 0.001, \eta_G^2 = 0.49$
tech	-	$F_{1,11} = 1.01, p = 0.336, \eta_G^2 = 0.01$
input \times tech	-	$F_{1,11} = 0.01, p = 0.918, \eta_G^2 = 0.00$

Figure 4.13: Results of Anova tests for task time.

First, *Handle* significantly outperforms *Hand* with a large effect size, and this overall difference is not impacted by the transition technique (tech) for *SurfaceToAir* tasks.

Figure 4.12 shows a breakdown of the total Task Time into: the time on the surface, the time in the air, and the transition time for *SurfaceToAir* tasks (*i.e.*, the interval between the moment the docking task is completed and the moment the *Hand/Handle* leaves the surface with a successful use of either *Flick* or *Grab*). We analyze each phase of the task below.

Transition. An Anova input \times tech on the transition time reveals: (i) a significant effect of input ($F_{1,11} = 17.9, p = 0.001, \eta_G^2 = 0.28$), with *Handle* being faster than *Hand* (1578 ± 126 ms vs. 2291 ± 158 ms); (ii) a marginal effect of tech ($F_{1,11} = 4.70, p = 0.054, \eta_G^2 = 0.13$), with *Grab* being faster than *Flick* (1764 ± 118 ms vs. 2109 ± 177 ms); and (iii) no input \times tech interaction effect ($F_{1,11} = 3.60, p = 0.084, \eta_G^2 = 0.02$).

An analysis of errors can partially explain these differences. A transition error happens when participants fail at transitioning from the surface to the air at their first attempt with either *Flick* or *Grab*. The following table reports the percentage of trials where such errors occur:

Flick		Grab	
<i>Hand</i>	<i>Handle</i>	<i>Hand</i>	<i>Handle</i>
$17.0\% \pm 7.6\%$	$7.6\% \pm 5.4\%$	$5.4\% \pm 4.6\%$	$2.2\% \pm 3.0\%$

First, we observe that we have more transition errors with *Flick* than with *Grab* ($p = 0.045, d = 0.9$). This might be specific to the case that we operationalize in our

experiment, with participants needing a wide viewing angle to decide on where to place the *Modulus*. As they do not know the direction of their future movement when they leave the surface, they make an arbitrary choice regarding the direction of their flick gesture. This could have played against the *Flick* technique. Second, we observe more transition errors with *Hand* than with *Handle* ($p = 0.037$, $d = 0.8$) for both transition techniques. Contrary to our hypothesis about *Flick* being better suited for *Hand* and *Grab* for *Handle* ($H_{transition}$), participants consistently performed better transitions with *Handle* than with *Hand* whatever the transition technique considered.

Surface. For the time spent on the surface (*i.e.*, time for docking), the comparison between *Hand* and *Handle* is very similar to what it is for the total task time. This supports results from previous studies [53], where tangibles were better than multi-touch gestures when used as surface controllers. We looked at some specific lower-level events to better interpret those observations. The table below reports the percentage of trials where participants (i) had to perform at least one *Clutch* action during the docking on surface; and (ii) entered and then left the target docking position (*Enter/leave*):

	<i>AirToSurface</i>		<i>SurfaceToAir</i>	
	<i>Hand</i>	<i>Handle</i>	<i>Hand</i>	<i>Handle</i>
Clutch	89.1% ± 6.4%	19.8% ± 8.1%	67.4% ± 6.8%	3.2% ± 2.6%
Enter/leave	55.4% ± 10.3%	39.6% ± 9.9%	59.3% ± 7.1%	24.9% ± 6.3%

There are significantly fewer clutch actions with *Handle* than with *Hand*, suggesting a more continuous control with *Handle* for *AirToSurface* tasks than with *SurfaceToAir* tasks (going down to 3.2% of trials with *Handle*). This is probably because participants came in contact while maintaining a *SideTrigger* posture. Putting the thumb down from this posture led to uncomfortable positions. Participants then tended to lift off their finger and reposition their hand in order to make future on-surface manipulations more comfortable. Finally, in the *Handle* condition, participants performed less enter/leave actions than in the *Hand* condition, suggesting a better precision control and stability with *Handle*.

Mid-air. For the time spent in the air, *Handle* is significantly faster than *Hand* for the *SurfaceToAir* task ($F_{1,11} = 27.4$, $p < 0.001$, $\eta_G^2 = 0.28$). However, for the *AirToSurface* task, the difference between *Hand* and *Handle* is not significant ($F_{1,11} = 0.92$, $p = 0.358$, $\eta_G^2 = 0.01$). These different results can be easily explained by the nature of the tasks and their difficulty:

- In *AirToSurface* tasks, participants had no difficulty with either *Hand* or *Handle* to put the *Modulus* over the *Stimulus* with the required tolerance of 7 cm when coming in

contact with the wall (participants almost never adjusted the position of the *Modulus* once on the surface).

- In *SurfaceToAir* tasks, participants had to release the *Modulus* over the *Stimulus* from afar with the same 7 cm tolerance. Consistently with our first experiment, it was more difficult with *Hand* (error rate 30%) than with *Handle* (error rate 10%). A larger tolerance would certainly lead to different results.

Subjective Feedback. Overall, participants were more confident with *Handle* (4.8 vs. 3.9, $p = 0.002$) and found that *Handle* was easier to use (4.5 vs. 3.5, $p = 0.002$), was less physically demanding (1.3 vs. 2.2, $p = 0.004$), and performed better (4.5 vs. 3.9, $p = 0.012$). The difference regarding mental demand is not significant (1.3 vs. 1.7, $p = 0.203$).

For the transition-specific questions, there are significant differences neither between the input conditions nor between *Flick* and *Grab*. Overall, the participants performed transitions with confidence (4.6), found the transition techniques easy to use (4.4), with low physical and mental demands (1.4 and 1.3), and perceived their performance as good (4.5).

Regarding rankings, *Handle* was better than *Hand* for all participants. For *SurfaceToAir* tasks, *Handle* with *Grab* is always ranked first (7 participants) or second (5 participants). *Handle* with *Flick* has also very good rankings: first for 5 participants, and second for 5 other participants. Looking at *Hand* only, a majority of participants (8) ranked *Grab* before *Flick*. For rankings related to the transition action only, 8 participants ranked first *Handle* with *Grab*, 3 participants ranked *Handle* with *Flick* first, and 3 participants ranked *Hand* with *Flick* first (2 *ex-aequo*).

Overall, *Handle* obtained better subjective scores than *Hand*, and was preferred by the participants. However, we could not observe clear differences between the transition techniques.

Summary of results

Our results support ($H_{general}$): *SurfAirs* performed better and were preferred over bare-hand input for hybrid interactions involving both surface and mid-air control as well as transitions between both. Moreover, participants were able (with a low error rate) to transition from the surface to the air with both the *Grab* and *Flick* techniques. Although *Grab* performed slightly better than *Flick*, our observations do not support ($H_{transition}$): there is statistical evidence neither that *Grab* is better for *Handle* nor that *Flick* is better for *Hand*. Finally, contrary to our first experiment, participants found *SurfAir*-based input significantly less tiring than bare-hand input.

4.6 Limitations

As opposed to off-the-shelf technologies, experimental setups like our wall room are unique. Observations are thus dependent on the specificities of the setup. In particular, tracking accuracy depends on the motion-capture system, the number of cameras and their positioning in the room. In our case, we optimized tracking accuracy for the volume effectively used during the studies. However, we chose to implement all our conditions using the exact same tracking setup so that it cannot be a confounding factor in comparisons across conditions. This means that even if absolute numbers are likely to be different in another setup, the comparison between conditions should be the same.

The tracking accuracy might also be impacted by the specific PQLab[©] frame that we use for capturing multi-touch input. Such a frame is based on optics, meaning that the fingers that are in contact with the surface can be occluded by other parts of the hand. In addition, fingers that are very close to the screen may even be considered to be in contact with it. Bare-hand input might have been impacted more than *SurfAirs* by these issues. However, whatever the technology considered, the rigid structure of a controller reduces variability. Controller-based input is thus usually more resistant to technological imperfections.

We have tested a specific implementation of bare-hand input, using the SideTrigger gesture for clicking and the flick gesture for transitioning. We chose the SideTrigger gesture because both our personal experience and the literature indicate that it reaches good performance in terms of speed, stability and precision [14, 106]. We chose the flick gesture to transition for multiple reasons. Firstly, a flick gesture can be performed independently from the click state for each input technique. Secondly, a flick starts on the surface and finishes in the air, strengthening the metaphor of moving control from the surface to the air. However, alternative gestures within the very large space of multi-touch gestures could be considered and tested. Similarly, alternative handle and switch designs could be considered for a physical controller. But, a high-level property that remains independent from the design choices is that a gesture requires training or per-user calibration while a mechanical action on a physical controller does not.

Finally, participants in the two experiments were researchers, engineers, or graduate students in Computer Science. They were all familiar with multi-touch gestures from their experience with personal devices such as smartphones and tablets. However, except for two participants, they had no experience interacting with a wall display, and none of them had ever used multi-touch gestures on a wall display. Although post-experiment questionnaires suggest that participants were comfortable in all the experimental conditions, replicating the experiment with expert users may decrease the difference between bare-hand gestures and *SurfAirs* as bare-hand input might

more benefit from learning effects. We also observe some anecdotal evidence that bare-hand input conditions may have been impacted by hand anatomy, with participants with large hands being more comfortable with multi-touch gestures than participants with small hands. We did not observe such a tendency in *SurfAir* conditions, which reinforces the hypothesis that *SurfAirs*' performance is less user-dependent.

4.7 Conclusion and Future Work

SurfAirs are physical controllers that can be tracked both on a surface and in the air. They enable controlling multiple degrees of freedom: translation and rotation on 2D surfaces, and full 6-DoF manipulation in the air. They are equipped with a switch that not only enables selection from afar but also enriches surface interaction with an additional state. For example, in our second experiment, we took advantage of this additional state to support translation, rotation and scaling on surfaces with a single controller. *SurfAirs* also feature mechanism based on a spring and suction cup [53] that makes them particularly well suited for interaction with vertical surfaces, as users can attach and detach them at will.

Our empirical studies compare the performance of *SurfAir*-based input against bare-hand input. Across the two studies, participants had to perform pointing and docking tasks in the air, on a surface, and across the air and a surface. *SurfAirs* performed better than, and were preferred to, bare-hand input. They enable steadier selections in the air, and more precise control both in the air and on-surface. In comparison with bare-hand input, they also have strong advantages by design. First, their manipulation is user-independent. Activating a switch with a change of hand posture is subject to both intra-user and inter-user variability as postures may vary over time and between users. In comparison, *SurfAirs* do not require any per-user calibration or pre-training. Second, *SurfAirs* do not require instrumenting the user, which is often cumbersome. For instance, even light instrumentation such as the finger sleeves we used (Figure 4.6) made it difficult for participants to fill out questionnaires during the study.

The fabrication of a *SurfAir* is relatively simple, relying on passive components, an optics-based multi-touch frame, and an optical tracking system that can accurately capture motion in 3D. Relying on modular components without any electronic connection was particularly convenient for testing alternatives during the design phase. For example, it allowed us to design the *Torch SurfAir* that features a switch that is orientation-independent. Such a switch would have been challenging to design with electronic components.

In our studies, we consider basic tasks with a single *SurfAir* in a specific technological setup. As future work, we would like to replicate our second experiment with other technologies such as a capacitive display or an electronic button for the *Handle*

SurfAir. Future work should also evaluate *SurfAirs* with more ecological, high-level tasks that involve *e.g.*, a higher cognitive demand, multiple controllers and multiple users. Finally, it would be worth studying *SurfAirs* in the context of multi-display environments that can feature both vertical and horizontal displays such as tabletops.

TANGIBLE INTERACTION FOR MULTI-DISPLAY ENVIRONMENTS

5.1 Introduction

Multi-display environments are now common and often ask users to interact with different devices. In addition to desktops and laptops, those spaces can include large screens such as wall displays [31, 229] or tabletops [16, 86] to enable collaborative work as well as personal devices such as smartphones [123, 148, 191] and tablets [21, 110, 169] that allow users to keep interacting while being mobile. From the smallest to the biggest devices reviewed above, HCI literature investigated the use of tangible controllers. Such controllers have proven efficient to interact with digital displays both on surface or in the air. On surface, they take advantage of eyes-free and precise manipulations [53, 69, 85, 213, 219], and increase other users' awareness [45] and improve collaboration between them [5, 66]. In the air, they can propose a rich set of interactions [21, 178], are faster and more precise than traditional devices [20] or bare hand gestures [49], and provide a wide variety of haptic feedback [164, 236].

However, tangible controllers are mainly used in a single device context [8, 22, 36, 47, 53, 55, 84, 90, 142, 234]. A couple of projects take interest in creating frameworks to provide ways of enabling tangible interaction in a multi-display environment [15, 169, 229], but they mainly contribute software architectures to help developers with multi-device applications without discussing the design aspects of a physical controller that would be used to interact with multiple displays.

In fact, creating versatile tangible controllers for different devices can quickly become challenging. Tangibles are often designed for contexts relying on specific hardware setups as they need to be recognized and tracked by the system. Recognition and tracking approaches are thus often coupled with the technology of the display itself. Such approaches are often based on vision [65, 79, 114, 238] or capacitive technology [26, 41, 46] but they can also rely on magnetic fields [126, 130], wireless communication [137, 201] or RFID tags [78, 96]. Moreover, the targeted display's physical characteristics (*i.e.*, size and orientation) often drive the controllers' form factor which

have to be adapted to that display. For instance, a tangible controller that has been built for a tabletop will probably not be usable on a smartphone if size constraints are not considered in the first place. This example illustrates how the form factor of a multi-display tangible has to be cautiously thought early in the design process to ensure its compatibility with multiple devices.

The third research question I address in this chapter is about extending tangible interaction to the other devices that often surround a wall display. A wall display is rarely used in isolation. It is most often used as an additional, shared display space that is used in conjunction with individual workstations, handheld devices or tabletops. Designing physical controllers that can work with multiple displays would offer advantages. In particular, it could *unify some interactions* [15] (for instance, the same physical controller could be used to control a magnifying lens over a map independently from where this map is displayed) and *facilitate the transfer of data between different devices* [208]. As a first step toward designing multi-display controllers, we survey the literature of tangibles that are used to control a physical display and analyze the different dimensions that impact their design. This chapter presents an overview of the literature about tangible controllers through a systematic survey reviewing 284 papers along with a tool to explore this corpus. We then present an analysis categorizing tangible controllers regarding 12 dimensions that impact tangibles' design and form factor. We highlight the projects proposing multi-display interactions and extract the physical design choices that make it possible. Based on those projects and our personal experience, we propose directions for interaction designers to create multi-display tangible controllers.

5.2 Survey Scope and Methodology

As TUI literature is wide and spans across multiple sources, we had to refine our exploration to cover as many projects as possible without going out of the scope of our research question. The data collection method we used is a selective literature survey of 4 different ACM conferences using the advanced search tool of the ACM digital library. Those conferences, CHI, UIST, TEI and ISS (formerly ITS), were chosen either because of their flagship status as HCI conferences or because tangible user interface projects usually represent a significant part of their proceedings. We thus start with the following query:

Q_{conf} : articles in CHI, UIST, TEI or ISS (ITS); and

$Q_{tangible}$: the *title* or *abstract* contains the word “tangible”.

This query outputs 2386 items. To reduce this corpus, we decided to only consider contributions of type Research Article to focus on “complete” projects, reducing the cor-

pus to about 1185 articles (note that the specific Research Article tag was introduced in the ACM DL in 2008 only, so we had to manually classify articles published before 2008 that would be not considered as Research Article – mostly CHI extended abstracts). Moreover, we defined a query that captures contributions that are about physical controllers used for interacting with one or several displays. To define such a query, (Q_{words}), we took inspiration from an initial corpus (C_{init}) of 43 papers that we were familiar with (mostly because we studied them in the context of my first two projects: *WallTokens* and *Surfairs*). Thus we added the following criteria to (Q_{conf}) and ($Q_{tangible}$):

Q_{type} : contributions of type Research Article; and

Q_{words} : the *title* or *abstract* contains at least one of the following words: “physical controller”, “display”, “surface”, “screen”, “table”, “tabletop” or “wall”.

And we obtained a corpus of 263 papers that we named C_{ACM} .

We first evaluated the coverage of that corpus using our C_{init} corpus, which we know to be relevant to our topic of interest. C_{init} contains 8 papers that could not be in C_{ACM} because they came from other venues than CHI, UIST, TEI, and ISS: two at AVI, and one in each of the following venue: OZCHI, GROUP, IEEE TVCG, IEEE TLT, IEEE Computer, and INTERACT. Among the 35 remaining papers of C_{init} , 22 are in C_{ACM} and 13 are not. Among these 13 papers not in C_{ACM} , 11 papers are not in C_{init} because the word “tangible” is not in the title neither in the abstract (typically because the article uses an other word to describe its central concept, such as “graspable UI” [70], or focuses on an input device, e.g., VisionWand [39], or a system, e.g., Miners [217], rather than on tangible input). The 2 remaining papers are not in C_{ACM} because they do not satisfy the Q_{words} query.

Then, we manually went through the C_{ACM} corpus of 263 items and found out that some papers were out of the scope of this survey which focuses on the use of a tangible controller for manipulating or adjusting parameters of an object of interest that is displayed on a physical screen spatially distinct from the tangible itself. We defined the following list of Exclusion Criteria (EC) to filter out papers that do not fall within that scope:

- EC_1 : **No input** (Figure 5.1, Left): Projects presenting tangible user interface that only provide output capabilities are excluded. For instance, examples of this exclusion case can be found in Yannier *et al.*’s project [232] where children observe how towers made from construction blocks fall in an earthquake simulation. In a similar way, Marshall *et al.* [144] propose to move a tangible sphere using ultrasound waves but the sphere is never used as an input controller. This “No Input” criterion also excluded projects where the tangible is the main focus of the interaction and the display is only used as a way to augment it. In such cases, the



Figure 5.1: Examples of project illustrating our exclusion criteria. On the left, a human-shaped avatar giving vitals output only. In the center, a controller used on the street without additional display. On the right, unmodified smartphones used to display additional information on tabletop. Images source: [75], [29] and [125].

tangible is the object of interest while we rather focus on tangibles that act as controllers to interact with an object of interest displayed on a distinct screen. For example, this criterion excluded Suzuki *et al.*'s project [198] where physical objects are augmented using a tablet camera to explain physics phenomena, or Bonnard *et al.*'s [27] project where pieces of paper are augmented to help pupils in primary school classify quadrilaterals. This criterion also excluded projects where the tangible is the display itself without providing any means of decoupling the controller from the display [186].

- **EC_2 : No distinct display** (Figure 5.1, Center): Projects where interaction does not rely on a physical display are excluded. These mainly represent projects where the tangible controllers are standalone embedding a display themselves [28, 29] or projects that do not involve a display at all [145, 153]. We also relied on this criterion to exclude projects that target immersive contexts of use where users interact with a fully virtual world that does not feature any physical display [194, 200].
- **EC_3 : No tangibles** (Figure 5.1, Right): Projects not relying on a custom tangible controller are excluded. These includes projects where either no tangible controllers are used for interaction [4, 23, 124] or projects where the controller is an already existing device such as a smartphone, a tablet or a smartwatch [125, 148, 150], leaving no space for any design considerations.

Note that, in some cases, a project could meet one of the exclusion criteria for some of its parts but not for all of its parts. This is the case, for instance, of Jansen *et al.*'s project [110] where a tangible slider is fixed on a tablet and the user manipulate that slider to adjust parameters of a visualization displayed on a distant wall. In that specific case, the tablet is not a distinct display but is rather part of the tangible itself (EC_2) but the wall is actually a distinct display that the user interacts with using the tangible slider. We thus do not exclude that project but we focus on the relationship

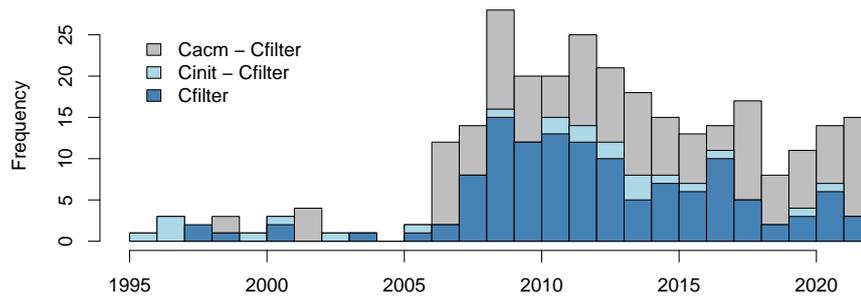


Figure 5.2: Years distribution of our corpora. In grey the papers from our query (C_{ACM}) that we excluded; in light blue the papers of (C_{init}) that are not in the query; and in steel blue (C_{filter}).

between the tangible slider (which includes the tablet it is mounted on) and the wall display.

Filtering out our (C_{ACM}) corpus of papers with those criteria left us with a corpus, (C_{filter}), of 126 papers (excluding 137 papers). To obtain our final corpus (C_{final}), we added the 21 missing papers from (C_{init}) resulting in 147 papers in C_{final} . Years distribution of our corpora is shown in [Figure 5.2](#).

5.3 Survey Dimensions

In this section, we describe and analyze each of the 12 dimensions that we used to classify the 147 papers in (C_{final}). Those dimensions were chosen based on the importance they have on the design aspects of tangible controllers. Among those dimensions, four of them are linked to the display properties, six of them are linked to the controller properties and two of them are linked to the posture of the user during the interaction. Each of those dimensions can take different values which we describe below. It is important to note that values on a dimension are not exclusive and that a project can belong to multiple categories of a given dimension. In such a case, we count the project in each of the categories it belongs to. Note that it entails that the sum of percentages in a category for a dimension, which we report below, can be higher than a 100%.

To set the values of the different dimensions for each project, we relied on:¹

- the user study or studies when the project reported on some,
- the text descriptions, illustrations or videos when the user study did not provide those information or when there was no user study reported.

¹We used the same resources for applying the exclusion criteria when refining our initial result set.

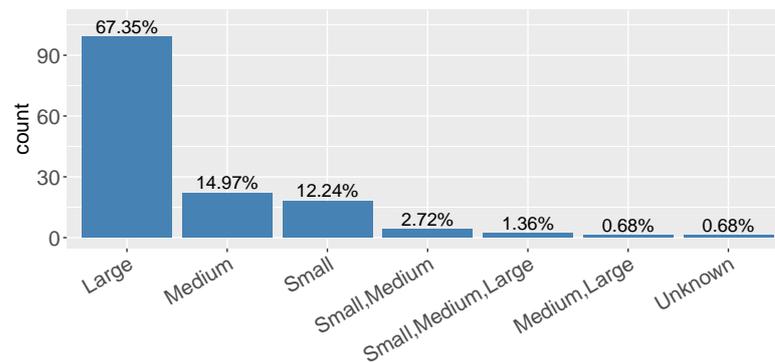


Figure 5.3: Display Size

However, some of the dimension values are sometimes hard to identify and/or not described at all. In such cases, we chose not to make any assumption and used the *Unknown* value for the dimension in question.

Display properties

- *Display Size* (Small, Medium, Big): This dimension describes the physical size of a display. The possible values are *Small* for tablet- and smartphone-sized devices, *Medium* for displays that typically fit a desktop workstation, and *Large* for tabletop-, smartboard- or wall-sized displays. The size of a display plays an important role on the form factor of a tangible and even more when the tangible is used as a surface controller. It typically constrains the size of the tangible itself which then impacts its weight, its portability, how the user can grab it, how much occlusion it creates on the display, and the number and type of additional features (e.g., buttons) that can be mounted on or included in it.

As illustrated by Figure 5.3, we observe that most of the projects from our corpus use tangible controllers with a Large display (69.39%). Tangible controllers for small- or medium-sized displays are rarer and represent respectively 15.32% and 19.73% of our corpus. This is not very surprising as the smaller the display is, the smaller the controller must be, and designing small but meaningful controllers is challenging. For instance, a WallToken (chapter 3) is roughly 10cm-large. This is good for a wall display but it can hardly be used on a tablet or a smartphone without causing major occlusion. In a multi-display environment, the controller's form factor should fit all displays, and, in particular, its size is constrained by the smallest one. However, there might be some trade-off to consider as, at the same time, a too small controller might have drawbacks. For

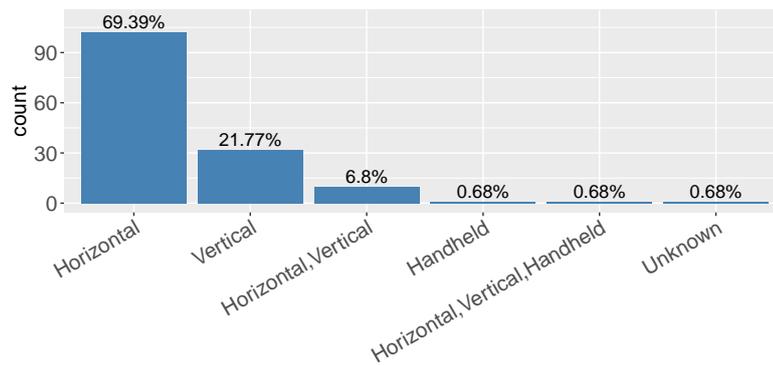


Figure 5.4: Display Orientation

instance, it might not offer a comfortable enough grip or it might be difficult to visually perceive on a very large screen.

- *Display Orientation* (Horizontal, Vertical, Handheld): This dimension describes the orientation of the display. The possible values are: *Horizontal* for displays such as tablesps or setups where a handheld device must be put flat on a table, *Vertical* for displays such as desktop displays or walls, and *Handheld* for mobile devices used normally. The display orientation mainly impacts the ability of letting the controller in place. As we discussed in previous chapters, this is useful for users to free their hands or give physical cues about the current state of the system. For horizontal surfaces, the controller form factor is not really impacted but for vertical ones, leaving the controller on the surface requires to design a feature to attach it. This is also true for tilted surfaces or handheld devices when they are actually held in hand and not put flat on a table.

In [Figure 5.4](#), we can see that only a third of our corpus allows tangible controllers to be used with vertical displays (29.25%) or handheld devices (1.36%) compared to 76.87% for horizontal displays. This can be explained by the fact that tangible controllers used on vertical displays, or handheld displays, requires to be held by the user while interacting, making them more cumbersome compared to horizontal surfaces where users can free their hands. However, this does not impact controllers that are used in a desktop context, where the display is vertical but the controller can stay on the desk surface if needed (*e.g.*, [71, 76, 84, 88, 94]). In a similar spirit to the tension about the right size for a controller mentioned above, designing for vertical displays can make controllers less meaningful as the attach feature can become a hindrance for other horizontal displays such as tablesps [90]. Also, the orientation of the display has an influence on the posture of users' hands when they hold the tangible on the sur-

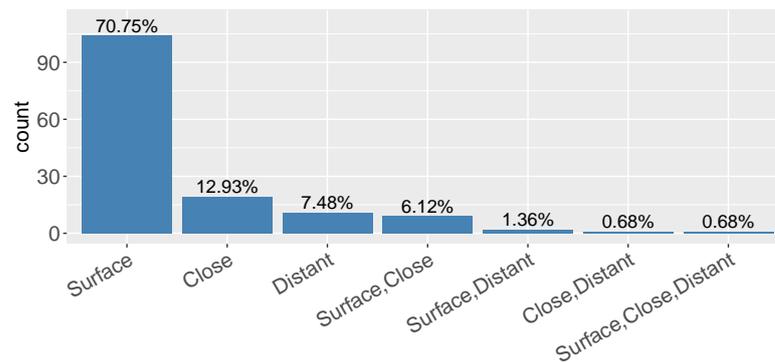


Figure 5.5: Display - Controller Distance

face. Designing for a variety of orientations entails considering more variability in the way users might have to grasp and manipulate the controller.

- *Display - Controller Distance* (Surface, Close, Distant): This dimension describes the distance at which the controller must be from the display. The possible values are: *Surface*, when the tangible has to be in contact with the display, *Close* when the controller has to be used in a restricted space nearby the display (hovering or surrounding it) and *Distant* when there is conceptually no limit regarding the distance the controller can be from the screen. In this latter case, the limit will rather be imposed by the physical environment (e.g., the size of the room) or the user's visual perception. This dimension is tightly coupled with the following one as it is practically constrained by the technology used to track the controller. However, the controller form factor is also impacted by this dimension as, for instance, leaving the surface of a display offer additional degrees of freedom regarding how users can manipulate the controller, which means that the controller's form factor must account for these richer manipulations.

As pointed out by Cherek et. al in [44] and confirmed in our survey, tangible controllers that are not used in contact with the display surface are rarer. If we combine both projects used close and at a distance from the display, they represent less than a third 29.25% of our final corpus (respectively 20.41% and 10.2%) [Figure 5.5](#). This might be partly due to the technologies that often make it easier to identify and track a controller when it actually is on the display. However, in a multi-display environment, finding ways to enable distant interactions is highly desirable as it not only expands individual displays' interaction space, but it can also enable continuous interaction between different displays.

- *Display - Controller Communication* (Cable, Capacitive, InfraRed, Magnetic, Vision, RFID, Wireless, Pressure): This dimension describes the communication

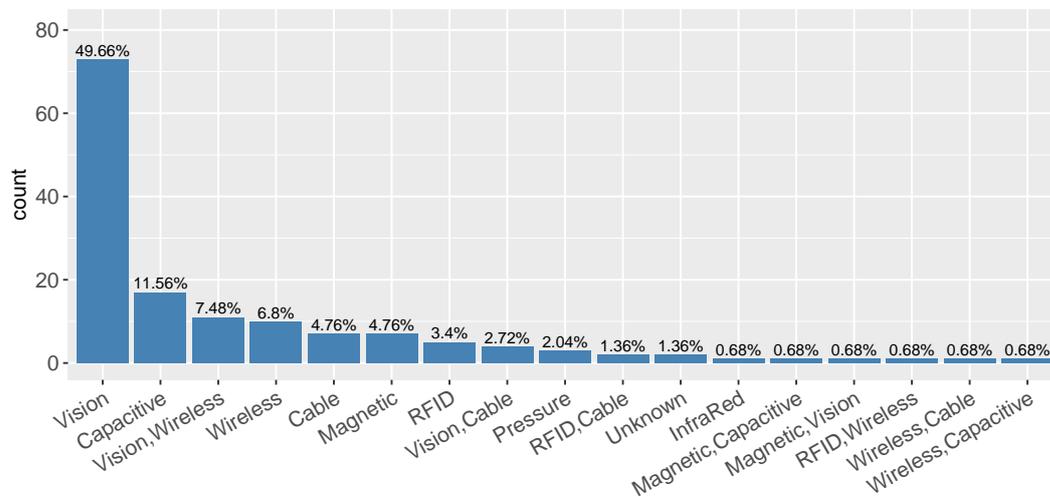


Figure 5.6: Display - Controller Communication

technology allowing the use of a controller with a display. The possible values are *Cable*, for controllers relying on wire-based communication, *Capacitive* for controllers equipped with capacitive footprint patterns, *InfraRed* for controllers equipped with physical footprints that can be detected using an infrared touch frame, *Magnetic* for controllers endowed with magnetic footprint, *Vision* for controllers endowed with markers and tracked using cameras, *RFID* for controllers with embedded RFID tags, *Wireless* for controllers embedded with wifi or blue-tooth modules and finally *Pressure* for controllers using physical footprint to be used with pressure sensitive surfaces. This dimension is, if not the most important, critical regarding the controller's form factor. For the controller to be usable, the communication between it and the device must go as flawless as possible. However, putting forward the quality of the tracking in a tangible controller design might impact multiple things according to the technology used. For instance, while it is easy to attach a passive marker under a tangible to be used on a tabletop that feature a camera under it, embedding some electronics in or mounting some infrared markers on a tangible will have a more important impact on its form factor.

Regarding the tracking/recognition technologies used, we see a clear unbalance in favor of technologies that can work with passive controllers (passive in the sense that they do not embed any electronics). In such cases, the recognition and tracking aspects heavily rely on the display's built-in technology. Figure 5.6 shows that, vision-based, capacitive, magnetic, RFID, pressure-based and infrared frame systems constitute 75.87% of the survey results while cable-based and wireless controllers sharing the remaining 22.48% (respectively 8.27% and

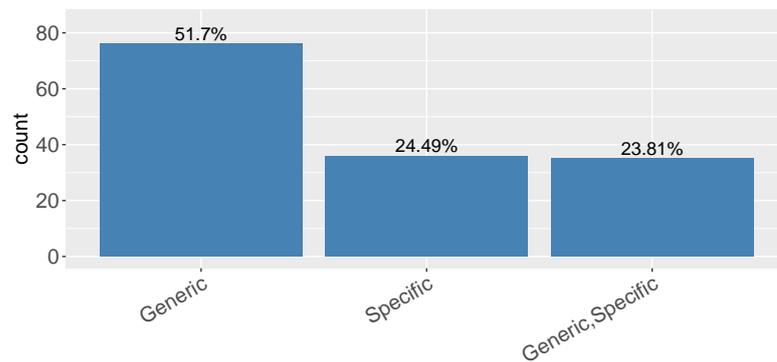


Figure 5.7: Controller Form Factor

14.2%). This unbalance could be explained by the constraint active tracking puts on a controller. Even though active parts like buttons or actuators can enable new interactions, they make the tangible design more difficult as they require knowledge in assembling and programming with electronics. In all cases, it is difficult to completely avoid technologies that rely on passive tracking, because it is very difficult to achieve accurate tracking with only electronics embedded in a tangible object.

Controller properties

- *Controller Form Factor* (Generic, Specific): This dimension describes if the controller shape is designed for generic or specific tasks. The possible values are *Generic*, when the controller shape relies on basic geometries and can be used in multiple scenarios or *Specific*, if the controller is designed for a very specific task or to depict a specific virtual object in the display's scene. This dimension is also crucial as the more complex a controller shape has to be, the more it will constrain other design dimensions.

Figure 5.7 illustrates that most of the projects in our corpus propose generic form factors (75.51%). Specialized controllers are often used in combination with generic ones when the project proposes a set of tangibles (48.3%). Only 23.81% of our corpus is about tangibles that have a specific form factor only. This makes sense as simple geometrical shapes like pucks, squares or other basic geometrical shapes are less restrictive in terms of possible applications than tangibles whose shape is specific to a role or application domain. Generic shapes are likely desirable in a multi-display environment as different displays may show different scenes and types of data. However, some manipulations may require specific grips that a generic shape might fail to support. For instance, writing or drawing need both a specific grip and a contact surface which minimize occlusion on

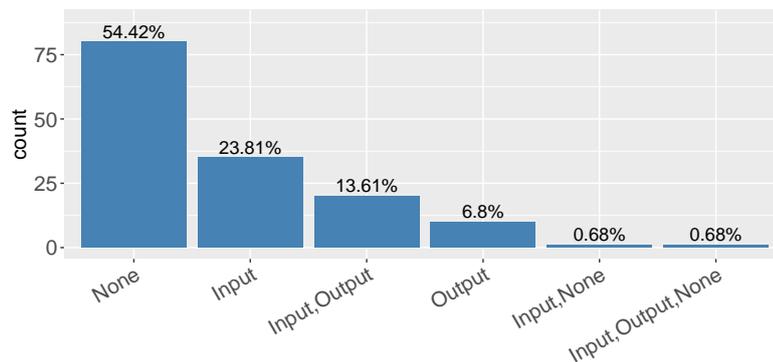


Figure 5.8: Controller Input/Output

the display as much as possible. In a multi-display environment, the number of different interactions can increase with the number of displays, likely calling for both generic and specific form factors. Such a design solution can be addressed using the *ControllerSet* and *ControllerModularity* dimensions discussed below.

- *Controller Input/Output* (Input, Output, None): This dimension describes the capabilities of a controller in terms of input or output. It captures I/O possibilities that are added to the controller itself such as a button (input) or active feedback in the form of vibrations or LEDs/screens mounted on the controller (output). The possible values are *Input*, if the controller has any input capabilities or *Output*, if the controller features any output capabilities. Adding I/O capabilities also impacts the form factor, often making it more complex. In addition to the electronic embedding it might require, the physical placement of additional components quickly becomes a design issue. For instance, a button or a tactile surface must be placed thoroughly on the controller in order for it to be used comfortably. The same is true for output to be easily perceived and understood by the user.

In 55.78% of our survey corpus, controllers do not embed any additional input or output capabilities (Figure 5.8). The rest of the corpus is shared between input (38.78%) and output (21.77%). This gives a hint that, in most cases, manipulations of the controller itself is a sufficient enough input channel. Adding more features, in addition to making the design more complex, also puts constraints on the form factor as the way the user grasps the controller needs to account for both interactions with the distinct display and interactions on the controller itself. This is a challenge we faced when designing the switch in the SurfAirs project (chapter 4). However, a multi-display environment may require controlling a large variety of functions and commands, which can put some weight in favor of more I/O capabilities on the tangible itself.

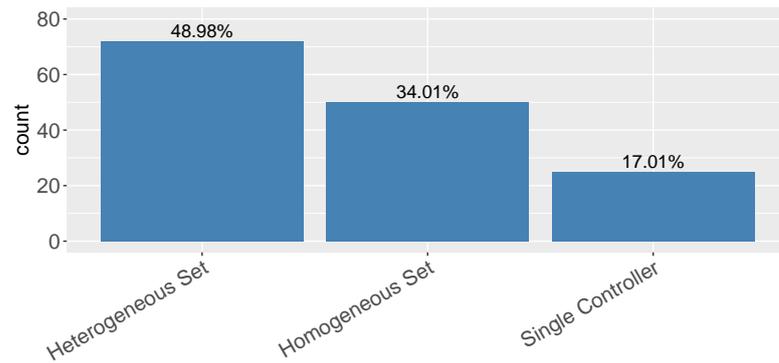


Figure 5.9: Controller Set

- *Controller Set* (Unique, Homogeneous Set, Heterogeneous Set): This dimension captures whether a controller is designed to be used in isolation or whether it can be used with other controllers to create a set of controllers. The possible values are *Unique*, if a project relies on a single controller ; *Homogeneous Set*, if a project relies on multiple controllers having a similar form factor ; and *Heterogeneous Set*, if a project relies on multiple tangible controllers with different form factors. Using a set of controllers as opposed to a single controller has consequences as the system must be able to identify each of them independently. This typically can have an impact on the form factor to make those tangibles feature identifiable traits. Moreover, this is also tightly linked to the other Display Dimensions, such as the Display Size or the Display Orientation, as multiple controllers can mean having enough display real estate to host them concurrently but it can also mean for the user to be able to put them down.

We observe that, in most of the articles, controllers are used in a set (83.99%) versus 17.01% cases where control is achieved with a unique controller (Figure 5.9). Providing a set of controllers diversifies the interactions available to the user. Multiple controllers open space for both generic and specific form factors as well as distributing the different controls across different tangible tools [65, 92, 159]. In a multi-display environment, profiting from multiple controllers might be an interesting design option to consider. However, it requires to take into account that, in a multi-display environment, the mobility of users could be impaired if they have to carry multiple controllers from one place to another.

- *Controller Modularity* (Modular, Not Modular): This dimension describes the degree of modularity of a tangible controller. A controller is *Modular* when it proposes ways of physically adapting its shape or function for different tasks. Such modularity can be achieved either automatically [194, 199] or manually with reconfigurable modules [165]. We also consider as *Modular* when several

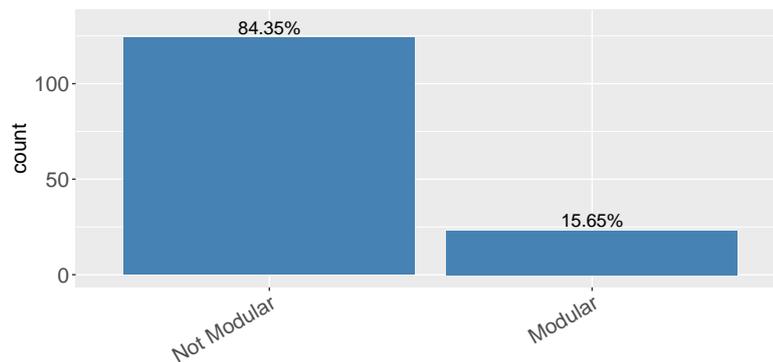


Figure 5.10: Controller Modularity

tangibles within a set can be combined to propose new form factors and/or interactions [41, 120, 127].

As illustrated in Figure 5.10, our survey corpus shows that tangible controllers are modular in only 15.65% of the projects. The vast majority of projects (84.35%) uses a single controller whose shape cannot be altered or a set of predefined controllers that cannot be combined. This dimension could however be key for multi-display interaction as it can impact all of the other dimensions covered in this section. In a multi-display environment, both for a controller set or a single controller, the lack of modularity means that they do not profit from a flexible form factor to adapt to different displays. The low number of projects that propose modularity can likely be explained by the fact that modular controllers ask more thinking regarding their design than a monolithic tangible. Furthermore, modularity can also mean more engineering effort as recognizing a controller that can take multiple shapes can be tricky.

- *Controller Transparency* (Transparent, Opaque): This dimension defines the transparency of a controller. The possible values are *Transparent*, if the display can be seen through the controller or *Opaque* otherwise. Making a controller transparent has great advantages as it allows to avoid occlusion but it also requires specific material which can impact the form factor of the controller as well as making the embedding of electronic components complicated.

As illustrated in Figure 5.11, only 21.77% of the controllers are transparent. Some projects propose both values if they propose a set of tangibles that feature both *Transparent* and *Opaque* tangibles. Transparency is an interesting feature for controllers used with a display, but loses some significance as soon as the controller is not used on the surface. Moreover, full transparency can be difficult to achieve when the controller has a non-flat shape as light gets deflected as it

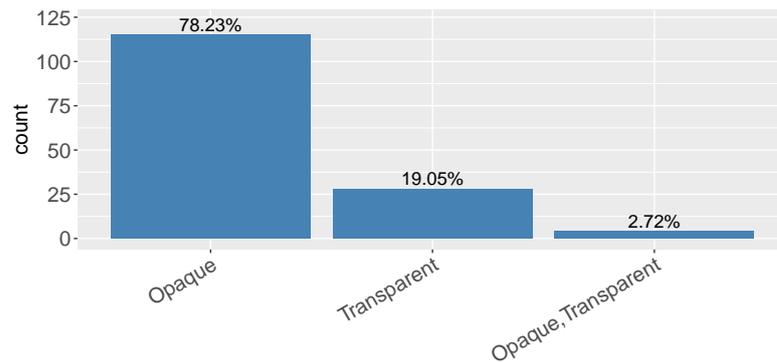


Figure 5.11: Controller Transparency

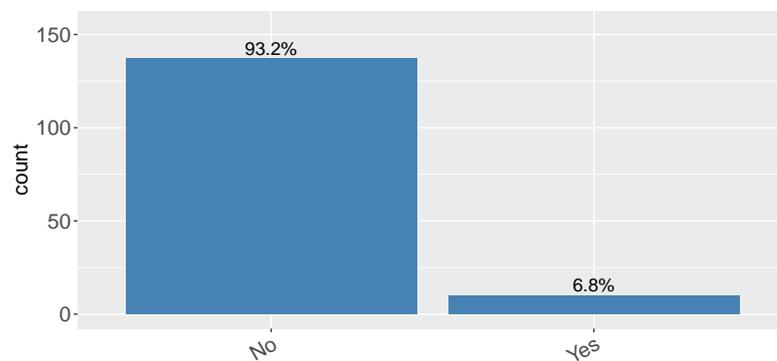


Figure 5.12: Based on Existing Controller

passes through the object. However, when it comes to interaction in a multi-display environment, transparency could be an interesting way to adapt large controllers to small displays, removing efficiently the occlusion problem without changing the controller's form factor.

- *Based on existing controller* (Yes, No): This dimension defines if the tangible controller is made from scratch or built upon an existing device. The possible values are *No*, if the controller is specifically built for the system or *Yes* if the controller relies on an existing device such as a smartphone or a smartwatch that is augmented for the project's needs. As mentioned above, our corpus does not include projects that use off-the-shelf devices only as they set aside all design-related questions. We consider solutions as soon as they have an impact on the form factor of the existing devices even if the impact is small (*e.g.*, equipping it with a specific casing [6, 140] or attaching additional grips [201, 226]). Even though adaptation of existing devices give some flexibility for the designer, it usually is quite constraining regarding the final form factor of the controller.

It is interesting to see that our survey corpus is mainly composed of fully cus-

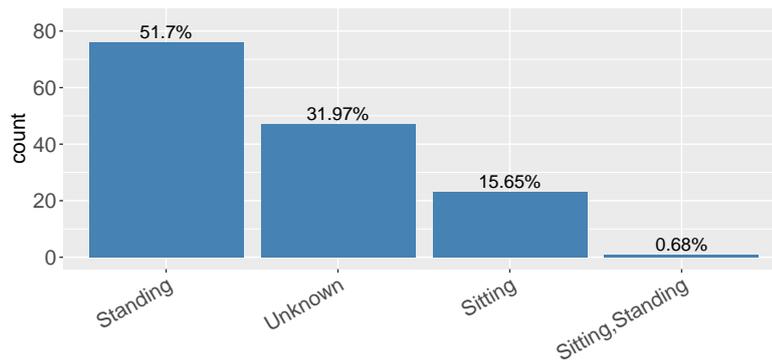


Figure 5.13: User Posture

tom controllers (93.2%). On the opposite, as illustrated in Figure 5.12, 6.8% of projects propose controllers that are built by enhancing pre-defined ones such as Nintendo Wii controllers [201], mice [226], smartwatches [140] or smartphones [123]. This suggests that designers of tangible controller tend to opt for homemade solution to adapt to their setups. This effect can be even more pronounced in a multi-screen environment, as the more different screens there are, the more constraints the controller design piles up. In this case, and even though relying on existing controller allows interaction designer to profit from already efficient designs, going from scratch might be the way-to-go to get a controller that is efficient with the various screens.

User Posture

- *User Posture* (Standing, Sitting): This dimension describes the posture of the user when interacting with the controller. The possible values are *Standing* if the user stands when interacting or *Sitting* if the user is seated when interacting. This dimension might impact the form factor. In particular, designers should take a particular care to the weight of a controller if the user will have to stand and keep holding it in the air. See Figure 5.13.
- *User Mobility* (Mobile, Static): This dimension describes if the user is mobile or not while using a controller. The possible values are *Mobile* if the user has to move in space while interacting with the controller or *Static*, if the user stays in place during the interaction. In addition to increased tracking requirements, this dimension also impacts the form factor as a mobile user aims for a more portable and lightweight controller. See Figure 5.14.

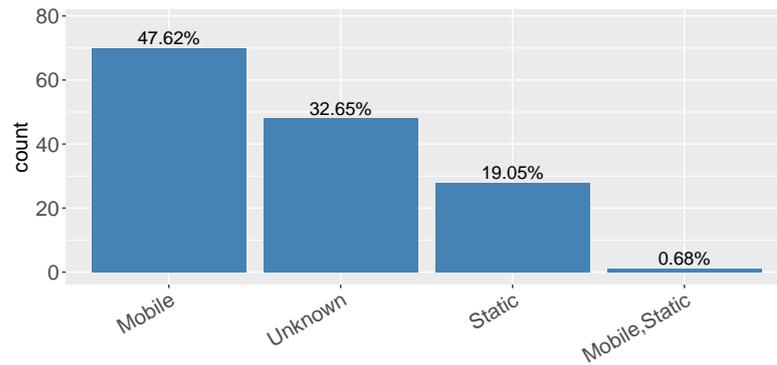


Figure 5.14: User Mobility

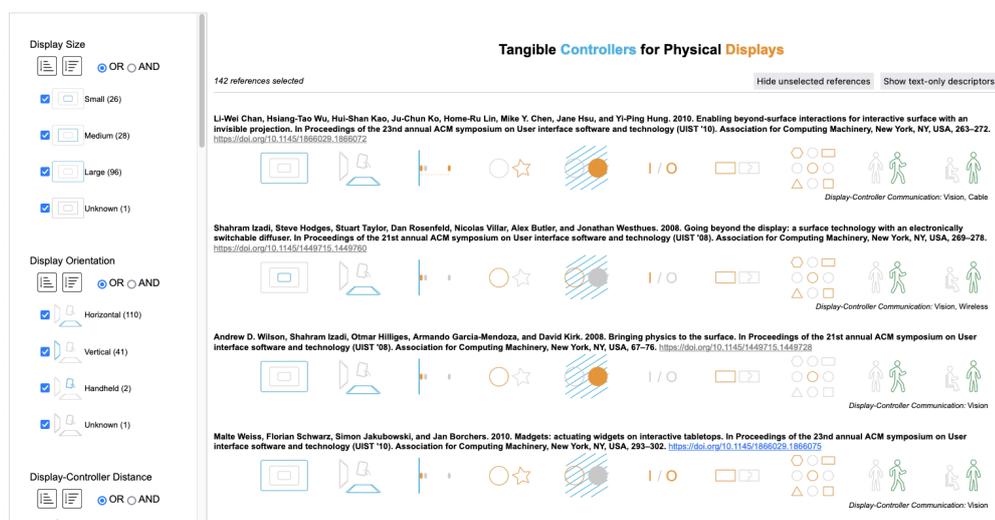


Figure 5.15: The Web tool to explore our corpus of papers. The left panel features one facet per dimension, and the main panel on the right lists the references in our corpus. A reference is represented by its textual description and a series of icons that represent the values it has along each dimension.

5.4 Survey Exploration Tool

To explore the corpus resulting from our survey, we created an exploration tool [Figure 5.15](#). This Web page, available at

<https://tangibles4displays.lisn.upsaclay.fr/>

lists the references and proposes a list of facets to filter out those references according to the dimensions described above. We used it to help us interpret the analysis that we present above, and we make it available to the community in order to help TUI designers explore the literature and take design inspiration for their specific setups.

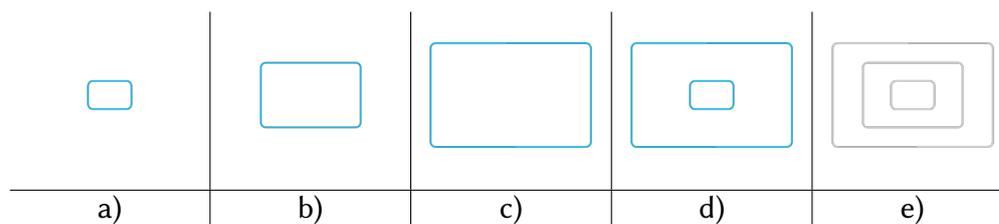


Figure 5.16: Pictograms used to represent values along the *DisplaySize* dimension. (a-c) Pictograms for projects that work with Small, Medium or Large displays. (d) Pictogram for a project that considers both Small and Large displays. (e) Pictogram for a project whose the value of *DisplaySize* is Unknown.

The left panel features one facet per dimension. Each facet can be set to either select the union (OR model) of the selected values or the intersection (AND model) of the selected values for that facet. By default, all values are selected and the facet filters according to a OR model. Each facet also features two small buttons that allow users to sort the references according to the values of that facet. Sorting along a facet will group references according to the number of values they take for that dimension. For instance, when sorting along *Display Size*, a reference that considers only *Small* displays will be listed before a reference that considers both *Small* and *Large* displays. Within a group of references that is tagged with the same number of values, the order matches that of the presentation order of the values under that facet title in the left side panel. For instance, a reference that considers only *Small* displays will be listed before a reference that considers only *Medium* displays. Each facet proposes two buttons for sorting according to an ascending or descending order.

To support visual exploration of the corpus, we designed icons for each value of a dimension. Those icons are designed so that they can be superimposed when a project combines multiple values for a single dimension. For instance, a project that proposes controllers for *Small* displays only will be depicted by Figure 5.16-a), a project that proposes controllers for *Large* displays only will be depicted by Figure 5.16-c), and a project that consider both *Small* and *Large* displays only will be depicted by Figure 5.16-d). We designed icons for most of our dimensions. However, when a dimension has too many values or is simply difficult to match with a simple visual representation, we kept a textual description. Users can also turn off visual representations to get text-only representations by clicking the Show text-only descriptors at the top left.

In addition to the 12 dimensions of interest, the tool also proposes a couple of additional filters:

- In/Out ACM search: To indicate whether the project was a result of our search in the ACM DL or whether we added it (*i.e.*, projects from C_{init} that were not in C_{ACM} are marked “Out ACM Search”).

- Application domain: The application domain for the project. Possible values include Art, Education, Visualization, Health, etc.²

At the very bottom of the page, users can click a link to see the list of projects that we put aside because of the exclusion criteria that we list above.

We made this information available to make the way we built our corpus transparent and because we believe that they might be useful to practitioners who would be, for example, interested in a specific application domain.

5.5 Designing for Multi-Display Environments

Designing tangibles for multi-display environments requires to take into account the variety of each display's properties. Thanks to the taxonomy and the tool that we present above, we can take a closer look at projects that cover multiple values regarding the different display dimensions. These projects are likely to provide clues about how to make tangible input that is versatile enough to adapt to different display type. For instance, even if they do not propose a straightforward multi-display interaction, projects that present controllers that work with displays of multiple sizes, or that rely on different communication technologies, or that can be used with either horizontal and vertical displays are worth to be investigated. We thus carefully look at the controller properties for these projects in order to understand the design solutions that they adopted to support multiple types of displays.

Among our final corpus of 147 projects, 53 of them take multiple values along at list one of the display dimensions described in [section 5.3](#): 13 of them allows multiple *Display Distances*, 7 of them can support at least 2 *Display Sizes*, 11 of them are usable on multiple *Display Orientations* and 22 of them relies on multiple *Display - Controller Communication*. This section presents our analysis of these specific projects.

Display Size

As illustrated in [Figure 5.3](#), only 5.44% of the projects propose interaction on more than one display size. Some of them simply propose controllers that are small enough to be usable with small display surfaces without causing too much occlusion [57, 233]. Another solution consists in not supporting on surface interactions (Display-Controller Distance \neq Surface), which somehow decouples the controller's form factor from the display size as they do not have to be in contact with each other [78, 178]. Other projects such as PUCs [212] or CapStones [41] propose a set of tangible objects of

²Identifying relevant values for the Application Domain dimension requires delimiting "domains", which is a notion that can be subject to debate. We plan to iterate again on the definition of this dimension as our current values may not be completely consistent.

different sizes so that users can choose the one that is adapted to the considered display. Finally, GaussBits [129] propose another approach where the magnetic tangible is not tracked by the technology of the display but rather by an external magnetic grid which can be attached to a smartphone, a tablet or a laptop.

However, some of these approaches have drawbacks that we already mentioned above. For instance, smaller controllers can adapt to any display size but they can suffer from uncomfortable grips and may be difficult to visually perceive on large screens. Avoiding on surface interaction is an interesting solution but it comes at the cost of not offering the precise and comfortable manipulations that on surface interaction enables. Having a set of controllers seems to benefit from all the advantages described above as users can switch at will from one controller to another according to the task and display at hand. However, the more different displays, the more controller might be needed to interact with those various displays, eventually impairing the user mobility. Decoupling the tracking technology from the surface is a good solution from a technological standpoint but does not address the design aspects related to the form factor a controller should take to support efficient and comfortable interactions. One solution that we personally think would be worth to investigate would consist of making modular tangible controllers, which would be able to adapt their shape according to the display size.

Display Orientation

Only 8.16% of the projects propose to interact with displays having multiple orientation (Figure 5.4). However, in most cases, this combination relies on a vertical and a horizontal display alternatively taking the role of display of interest and, when the display of interest is the vertical one, the user relies on the horizontal one for manipulating the tangible. Indeed, in projects such as [48, 51, 65, 92, 165, 180, 230], users manipulate the tangible on the horizontal surface to act on the representation displayed on the vertical surface. Such a solution not only makes the position of one display dependent on that of the other display but also reduces users' mobility as they must stay close to the horizontal display to interact, thus losing the advantages of movements in space. In Jansen *et al.*'s project [110], a tablet is used to interact with a wall display. This time, as the tangible is manipulated on a mobile device, the user can freely move in space. However, while attaching tangibles to a handheld device makes users free to move in space without worrying about gravity, it also entails that users cannot manipulate the tangibles directly on the display of interest.

Interestingly, Gomes de Siqueira *et al.*'s project about tangibles for interacting with scientific posters [57] is the only one showing a capacitive controller used on display surfaces that have different orientations, making the assumption that, in a near future, scientific posters will likely be presented in environments that feature multiple capacitive displays. However, as those tangible knobs cannot be attached to the display, the

possibilities in terms of interaction quickly become limited on vertical surfaces. Some projects actually address this issue and, even if they do not propose multi-display interactions, give hints on how the possibility for a token to be attached to a surface can enable interactions for both vertical and horizontal displays. For instance, WallTokens (chapter 3), Vertibles [90] and Back to Tangibility [159] use suction cups to achieve this while Geckos [126] and Mechanix [204] use magnets. However, when the design includes a solution to make the controller attachable on vertical surfaces, this solution must not impair the use of the controller on horizontal ones. For instance, Vertibles' design makes the controllers attachable on vertical displays, but they are then static once put on the display surface. WallTokens, on the opposite, rely on a physical mechanism that supports two states, attached or detached, and transitions between these states in order to make users able to slide tangibles on the surface. Geckos and Mechanix relies on magnetic surfaces offering both attachment and movement at the same time but such magnetic-based solutions require some trade-offs. In particular, they rely on projection-based displays as this makes it easy to make a display magnetic, but this is at the cost of losing the high resolution that an active display can offer.

Designing controllers for multiple display orientations requires trade-offs that have repercussions either on the display technology or on the controller's form factor as it is difficult to design physical objects that are both convenient and comfortable to use on both vertical and horizontal displays. One solution that we personally think would be worth to investigate and that has not been considered in the literature is either to build sets of tangibles that propose different types of controllers suitable to different display orientations or to make modular controllers onto which different handles and attachment mechanisms could be placed as needed.

Display-Controller Distance

As pointed out by Cherek *et al.* [44], smoothly transitioning between on-surface control and distant control allows interaction designers to empower their tangible controllers. However, projects proposing multiple value for the Display-Controller Distance dimension represent only 8.84% of our corpus (Figure 5.5). Most of these projects rely on multiple technologies to track the controllers both on the display surface, close, or afar from it. This can be achieved by combining the display tracking technology with cameras [15, 169] or by using active controllers with embedded wireless communication technologies [63, 81, 210, 236]. Some projects also use a single technology that is able to cover not only the surface, but also the space surrounding it. For instance, the Portico project [8] mounts cameras on a tablet to track the space on the surface but also surrounding it. In a similar spirit, Wu *et al.*'s project [230] uses a camera mounted above a tabletop to allow the tangible to be lifted and lowered to zoom in/out while allowing panning at a specific zoom level. Magnetic-based track-

ing systems also allows interacting while hovering [129] or being close to the display surface [22]. Another solution to interact from different distances is to rely on a set of controllers [42, 65, 236], with some usable on the display surface and other usable from a distance.

In a multi-display environment, interacting from various distances is important not only because users will typically move in space, but also because it might be desirable to empower the whole environment with the possibility of making continuous interactions between displays. This dimension is tightly linked to the technology used for the controller-display communication as not all these technologies are on an equal footing regarding the interaction space covered. While it is possible to rely on a set of controllers to switch to different distances, using a global technology that covers the whole environment seems the most promising solution. Vision-based tracking seems the most obvious solution. However, such an approach often entails heavy and complex hardware setups and its tracking accuracy may quickly degrade in a large space involving multiple users and objects that will cause a lot of occlusion.

Display-Controller Communication

In our corpus, 15.64% of the controllers communicate with displays using more than one technology [Figure 5.6](#). Most of them use a combination of vision-based system to track the controller position on the surface and a wireless communication system embedded in the controller. This combination is mostly used to provide input and output capabilities to active controllers [6, 85, 100, 130, 151, 195, 196, 199]. As mentioned above, combining technologies can also enable tracking at different display-controller distances [15, 63, 81, 140]. Cables are also used in combination with vision-based technologies for active controllers that have some I/O capabilities [42, 111]. Combining technologies can also help increase the accuracy when tracking controllers [77].

A multi-display environment often features displays that rely on different technologies. A tangible for such an environment should thus support various communication technologies to be versatile. This is a challenging design concept. A solution to address this is to use a set of controllers with each one of them being adapted to different communication system. While this may seem like a purely technical question, it is actually not completely technical, as it often requires the integration of certain components in the controller so that it can be tracked (*e.g.*, a specific footprint/marker or some electronics). This can have an impact on the form factor. In all cases, multi-display environments likely call for combining technologies in order to get a good tracking accuracy anywhere in the environment. Controllers should thus be able to get tracked by each display surface using the surface's native communication system, and by a global vision system when interacting in between displays. However, it certainly causes non-trivial challenges for integrating heterogeneous tracking systems.

5.6 Limitations

Although we tried to give an overview of the literature focused on the use of tangibles with physical displays, our survey may have some limitations. First, our research was focused on ACM conferences. Although we think that it certainly is the most relevant source, other sources from the IEEE digital library might be interesting to look at to cover a wider range of the literature. Second, the exclusion criteria (section 5.2) that we used to refine our results helped reduce the scope of our set of references, they may have also excluded interesting design solutions that were applied in another context and that could prove relevant for multi-display environments. Finally, the dimensions we chose to classify the different projects were the ones we thought would be the most useful regarding the design of tangible controllers but they do not cover all aspects of the fabrication and development process. Other considerations such as material cost and availability, production and assembly time or software implementation complexity were not considered in this work. When producing a physical object, those considerations are nevertheless important but they are rarely central to research activities and are therefore rarely discussed in articles. Such a lack of detail is even true for some of the dimensions that we considered. For example, user posture and mobility are often not explicitly described (Figure 5.9, Figure 5.10). In some cases, even the tracking technology used is not reported [217]. Despite those limitations, we still believe that the dataset resulting from our survey, as well as the exploration tool we provide (section 5.4) should be useful to designers for creating tangible controllers for multi-display environment or for a single display.

5.7 Conclusion and Future Work

In this chapter, we present a survey on the use of tangible controllers with physical displays that we built with a systematic search among four ACM conferences and an initial corpus of relevant papers that we were familiar with. Our motivation was to identify interesting design approaches to enable tangible interaction in a multi-display environment. Our survey covers 27 years of projects using tangible controllers with displays. However, apart from a few projects that took interest in using such controllers in a multi-display environment [15, 169, 229], most contributions are about single display setups. As we discuss in this chapter, this can be explained by the numerous constraints that underly the design of a versatile controllers.

We organize our corpus of references along 12 dimensions that we group into two main categories: display properties and controller properties. Those dimensions are chosen to represent the constraints that the designer of a tangible controller has to consider when creating a controller for a specific setup. Our work contributes a classification of 147 papers along those dimensions and a Web-based tool to present and

explore them. We also contribute an analysis of the corpus along those dimensions showing that few controllers actually support some degree of versatility. We discuss those specific projects, discuss the trade-offs they require and suggest approaches that are worth exploring in the context of a multi-display environment. In particular, we believe that modular controllers are promising.

As future work, we want to widen our corpus by considering more sources and include dynamic interactive charts in the Web-based tool to facilitate the exploration of the dataset and identify trends in physical controller design. We would also like to open it for other researchers to contribute to the database. Finally, we plan to build some prototypes of modular tangibles to investigate how this design dimension could impact tangible multi-display interaction.

CONCLUSION AND PERSPECTIVES

This thesis investigated the use of Tangible User Interfaces (TUIs) to empower interactions with wall displays. TUIs have been investigated in many contexts, but rarely for interacting with large vertical displays. As discussed in this manuscript, using tangibles with vertical surfaces raises many challenges regarding interaction design that are mostly related to the vertical orientation of the display and the large space users evolve in. My thesis work addresses these challenges to answer the broad research question: “How to enable tangible interaction with wall displays?”.

This final chapter summarizes my contributions towards answering that question and discusses possible future research directions to push further tangible interaction with wall displays.

6.1 Summary of Contributions

My thesis contribution is three-fold, addressing the three specific Research Questions (RQ) introduced in [section 1.1](#).

***RQ₁*: How to enable *on surface* tangible interaction on a wall display?**

Interacting with wall displays can happen from a distance or close to it. In this second case, users usually rely on the tactile technology the wall display features. Using tangibles for *on surface* interaction has been widely explored on horizontal surfaces. However, with vertical surfaces, users cannot leave tangibles on the display surface without holding them, which quickly impairs user experience. My first contribution, *WallTokens* ([chapter 3](#)), proposes a solution to this problem with a set of tangible tokens designed for *on surface* interaction with wall displays. In particular, each *WallTokens* features a simple mechanism with a push-handle that controls a suction cup. This makes it easy for users to switch between sliding the token or attaching it to the wall. Those tangibles are easy to fabricate and are recognized on the display

surface passively thanks to their physical footprints. In addition to the low-cost and easy-to-replicate design that we propose, we report on a study showing the benefits of *WallTokens* for manipulating virtual objects over multi-touch gestures. *WallTokens* proved more precise and comfortable than bare-hand gestures to perform low-level manipulation tasks on a wall display surface. This project is a first step towards enabling tangible interaction in a wall display context.

Major parts of this contribution were published as a full paper at CHI'21 [53]

***RQ₂*: How to enable continuous tangible interaction from the wall surface to the air?**

Enabling tangible interaction on wall displays requires to not limit the interaction space only to its surface. Even if interacting on the display itself enables precise manipulations with a detailed view on the scene, users often move in space along the wall or away from it. Taking some distance actually allows users to have a wide viewing angle and makes large-scale manipulations easier. My second project, *SurfAirs* (chapter 4), aims at making users able to interact both up-close and afar from the wall while offering some continuity across these two types of interaction. *SurfAirs* are physical controllers that users can manipulate on screen (*on surface* input), in the air (*mid-air* input), and transition from the surface to the air during a single manipulation (*hybrid* input). We propose two different *SurfAir* prototypes usable in a wall display environment and report on two user studies that compare *SurfAirs*' performance with bare-hand input for both mid-air and hybrid input. *SurfAirs* performed better on low-level manipulation tasks than bare-hand gestures regarding accuracy, speed and user preferences.

Major parts of this contribution have to appear as a full paper at CHI'23 [54]

***RQ₃*: How to design a multi-display/multi-purpose tangible controller for interactive rooms that feature more than just a wall?**

A wall display is often located in an environment that features multiple devices such as tabletops, laptops, tablets or even Augmented Reality headsets. While all these devices can work independently, a richer user experience could be achieved if those devices could actually work in concert, efficiently complementing one another. In my third contribution, I thus consider interaction with a wall display in a larger context, tackling the challenges that entail the design of tangibles for interacting with multiple displays. We contribute a survey (chapter 5) that reviews projects that

have investigated the use of tangible controllers with various types of displays over the last 27 years. We systematically explore a corpus of references built over the four main ACM conferences that present contribution in the domain of TUIs. We filter out projects that made use of tangible controllers in combination with a physical display, resulting in a dataset of 147 research articles. We present 12 dimensions to describe and organize this dataset, capturing properties of the controller, the display and the communication between the two. We present a Web tool to explore our dataset along those dimensions, and we use it to identify the projects that feature interesting properties for designing tangibles that are versatile enough to work with diverse displays. We discuss these projects, and propose some directions for designing multi-display tangibles. Although we focused on the specific use case of multi-display tangibles, our survey and tool can help TUI designers in general to identify projects that rely on a specific setup.

This project is presented for the first time in this thesis manuscript, it has not been submitted for publication yet.

6.2 Future Work

My thesis studies tangible interaction with wall displays, considering the rich interaction space they can propose. This interaction space is actually not limited to the surface of the wall itself but rather encompasses the potentially large volume in front of it that can accommodate multiple users who move in space and multiple devices. From the surface to the air, and finally to the surrounding devices, we empirically or theoretically investigated the use of tangibles for interacting with wall displays. However, while this thesis is a step towards enabling tangible interaction with wall displays, it also opens up several research questions that we would like to explore in the future.

Tangibles for Vertical Displays

Both *WallTokens* and *SurfAirs* are tangible objects that users can attach on the wall display. While it has the obvious advantage for users to free their hands when needed, we believe that it offers other interesting properties in terms of manipulation and perception. In terms of manipulation, the user can set the token's position in space while still being able to rotate it to use it as a rotary knob. In terms of perception, leaving a physical object on a digital scene can serve as a visual marker that is easy for users to spot. Our personal experience is that, on a large display, a physical object is much more salient than a virtual object is, especially when the digital information space is complex. We would like to study this *pop out* effect, taking inspiration from

studies in experimental psychology on preattentive processing [202]. Factors such as the size of the visual marker, the complexity of the digital scene and the user's distance to the wall would be interesting to study when comparing physical and virtual markers regarding their visual saliency.

Related to the question above, a design challenge that would be interesting to tackle is the construction of self-actuated controllers that would be able to move along vertical surfaces. It would require to find a means to make tangibles stick on the display surface without preventing their mobility. On horizontal surfaces, Weiss *et al.* [222] propose a solution to actuate tangibles with magnets. As magnets can also be used as an attach solution on vertical surface, they could be used to combine both actuation and an attaching mechanism. However, as we discussed earlier in this manuscript, using magnets is not possible with many active displays so this would probably require to rely on passive, projection-based displays or on displays that are equipped with specific technology (as in *e.g.*, [128]). An alternative could consist of embedding motors in passive tangibles to eventually reproduce caterpillar-like movements while relying on suction cups.

Tangibles for Multi-display Environments

In [chapter 5](#), our analysis of projects on tangible interaction reveals that tangible controllers are often declined into sets of controllers when used with displays. While this enables a certain versatility as users can switch from one tangible to another depending on their needs or on the display technology, it can not only prevent continuous interactions but also be quickly cumbersome when users move around as they have to carry multiple tokens. Investigating how to address this issue would be a promising direction to enable tangible interaction in multi-display environments. One solution could consist of making users able to leave their tangibles in place even in the air. While the tangibles presented in this thesis work can be easily attached to a vertical surface, it does not provide a solution when interacting from a distance. Taking inspiration from projects like BitDrones [79] could help design controllers that can stay in place mid-air making users able to pick them on the go or leave them floating in the interaction space. Combined with the idea presented in the section above, such tangibles could be self-actuated to go back and forth between the surface and the air.

An alternative or complementary design solution that we also discuss to enable multi-display tangible interaction is modularity. Users could reconfigure tangibles to *e.g.*, adapt their form factor depending on the manipulations they want to do and the displays they are interacting with. Studying how to design such modular tangibles is a question that we would like to explore. For instance, the grip and the contact surface of a tangible could be made modular so that users could change them according to the display it is used on, or the task it is used for. Such modularity could also make the tangible customizable, changing its form factor according to the user's hand size

or handedness. Modularity could also be used to plug/unplug some input or output capabilities on the tangible when needed. The challenge lies mainly in the design of such controllers because changes to the tangible must be easy to make and the modular parts must not become cumbersome when not needed. We could think of modules designed as drawers or modules that can be folded so that users do not end up in a situation that is close to handling a set of distinct controllers.

Tangibles for Collaboration

Although we demonstrate in WallTokens's companion video how multiple users can interact with tangibles on a wall display, this thesis mostly focused on the situation where only one user is interacting with the wall display. Studying tangibles' performance and the role they can play in multi-user scenarios is an interesting research topic. Tangibles have been shown to improve user collaboration on horizontal surfaces [207] but does it transfer to the specific interaction space a wall display offers? We actually believe that tangibles could help collaboration not only in contexts that involve single wall display but also in multi-display contexts. For instance, the physicality of the tangibles might change the way users exchange roles and information when accomplishing collaborative tasks in a multi-display environment, which could better support coordination. Studying such phenomena requires to design scenarios that operationalize high-level collaboration tasks that involve tightly-coupled collaboration and/or loose collaboration such as *e.g.*, sensemaking tasks [2] or classification tasks [133].

We also believe that considering collaboration during the design process of tangibles could generate interesting alternative design solutions with *e.g.*, specific form factors or output capabilities that promote awareness of other users' actions. In multi-user contexts, it is important for users not only to be aware of other users' actions but also to be aware of the workspace state to ensure an efficient coordination. While we focused on tangible controllers, *ambient* tangibles could complement our approach to convey information about the workspace and about other users. Such tangibles could be hung to the ceiling [30, 174], attached to the walls [189], or put on the ground [50, 116, 192] to take into account the large space users evolve in.

BIBLIOGRAPHY

- [1] Jérémy Albuys-Perrois, Jérémy Laviolle, Carine Briant, and Anke M. Brock. 2018. Towards a Multisensory Augmented Reality Map for Blind and Low Vision People: A Participatory Design Approach. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, 1–14. <https://doi.org/10.1145/3173574.3174203> 5
- [2] Christopher Andrews, Alex Endert, and Chris North. 2010. Space to Think: Large High-resolution Displays for Sensemaking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 55–64. <https://doi.org/10.1145/1753326.1753336> 101
- [3] Christopher Andrews, Alex Endert, Beth Yost, and Chris North. 2011. Information Visualization on Large, High-Resolution Displays: Issues, Challenges, and Opportunities. *Information Visualization* 10, 4 (Oct. 2011), 341–355. <https://doi.org/10.1177/1473871611415997> 15
- [4] Alissa N. Antle, Allen Bevans, Theresa Jean Tanenbaum, Katie Seaborn, and Sijie Wang. 2010. Futura: Design for Collaborative Learning and Game Play on a Multi-Touch Digital Tabletop. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, 93–100. <https://doi.org/10.1145/1935701.1935721> 76
- [5] Alissa N. Antle, Alyssa F. Wise, Amanda Hall, Saba Nowroozi, Perry Tan, Jillian Warren, Rachael Eckersley, and Michelle Fan. 2013. Youtopia: A Collaborative, Tangible, Multi-Touch, Sustainability Learning Activity. In *Proceedings of the 12th International Conference on Interaction Design and Children (IDC '13)*. ACM, 565–568. <https://doi.org/10.1145/2485760.2485866> 5, 11, 24, 73
- [6] Ahmed Sabbir Arif, Roozbeh Manshaei, Sean DeLong, Brien East, Matthew Kyan, and Ali Mazalek. 2016. Sparse Tangibles: Collaborative Exploration of Gene Networks Using Active Tangibles and Interactive Tabletops. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, 287–295. <https://doi.org/10.1145/2839462.2839500> 86, 93
- [7] Mirjam Augstein, Thomas Neumayr, Isabel Karlhuber, Sabine Dielacher, Sylvia Öhlinger, and Josef Altmann. 2015. Training of Cognitive Performance in Complex Tasks with a Tabletop-Based Rehabilitation System. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 15–24. <https://doi.org/10.1145/2817721.2817738> 5

- [8] Daniel Avrahami, Jacob O. Wobbrock, and Shahram Izadi. 2011. Portico: Tangible Interaction on and around a Tablet. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 347–356. <https://doi.org/10.1145/2047196.2047241> 73, 92
- [9] Alec Azad, Jaime Ruiz, Daniel Vogel, Mark Hancock, and Edward Lank. 2012. Territoriality and Behaviour on and around Large Vertical Publicly-Shared Displays. In *Proceedings of the Designing Interactive Systems Conference (DIS '12)*. ACM, 468–477. <https://doi.org/10.1145/2317956.2318025> 14
- [10] Benjamin Bach, Ronell Sicat, Johanna Beyer, Maxime Cordeil, and Hanspeter Pfister. 2018. The Hologram in My Hand: How Effective is Interactive Exploration of 3D Visualizations in Immersive Tangible Augmented Reality? *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 457–467. <https://doi.org/10.1109/TVCG.2017.2745941> 20
- [11] Robert Ball and Chris North. 2008. The Effects of Peripheral Vision and Physical Navigation on Large Scale Visualization. In *Proceedings of Graphics Interface 2008 (GI '08)*. CIPS, 9–16. <https://dl.acm.org/doi/abs/10.5555/1375714.1375717> 17
- [12] Robert Ball, Chris North, and Doug A. Bowman. 2007. Move to Improve: Promoting Physical Navigation to Increase User Performance with Large Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, 191–200. <https://doi.org/10.1145/1240624.1240656> 17, 45, 46
- [13] Till Ballendat, Nicolai Marquardt, and Saul Greenberg. 2010. Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, 121–130. <https://doi.org/10.1145/1936652.1936676> 13, 19
- [14] Amartya Banerjee, Jesse Burstyn, Audrey Girouard, and Roel Vertegaal. 2011. Pointable: An in-Air Pointing Technique to Manipulate out-of-Reach Targets on Tabletops. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. ACM, 11–20. <https://doi.org/10.1145/2076354.2076357> ix, 17, 18, 19, 45, 47, 54, 69
- [15] Stefano Baraldi, Luca Benini, Omar Cafini, Alberto Del Bimbo, Elisabetta Farella, Giulia Gelmini, Lea Landucci, Augusto Pieracci, and Nicola Torpei. 2008. Evolving Tuis with Smart Objects for Multi-Context Interaction. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems (CHI EA '08)*. ACM, 2955–2960. <https://doi.org/10.1145/1358628.1358790> 73, 74, 92, 93, 94

- [16] Tom Bartindale, Jonathan Hook, and Patrick Olivier. 2009. Media Crate: Tangible Live Media Production Interface. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 255–262. <https://doi.org/10.1145/1517664.1517718> 5, 73
- [17] Michel Beaudouin-Lafon, Olivier Chapuis, James Eagan, Tony Gjerlufsen, Stéphane Huot, Clemens Klokrose, Wendy Mackay, Mathieu Nancel, Emmanuel Pietriga, Clément Pillias, Romain Primet, and Julie Wagner. 2012. Multi-surface Interaction in the WILD Room. *IEEE Computer* 45, 4 (April 2012), 48–56. <https://doi.org/10.1109/MC.2012.110> ix, 4, 5, 16
- [18] Ross Bencina, Martin Kaltenbrunner, and Sergi Jorda. 2005. Improved topological fiducial tracking in the reactivation system. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*. IEEE, 99–99. <https://doi.org/10.1109/CVPR.2005.475> 15
- [19] Corentin Bernard, Jocelyn Monnoyer, Sølvi Ystad, and Michael Wiertlewski. 2022. Eyes-Off Your Fingers: Gradual Surface Haptic Feedback Improves Eyes-Free Touchscreen Interaction. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. ACM, Article 427, 10 pages. <https://doi.org/10.1145/3491102.3501872> 5
- [20] Lonni Besançon, Paul Issartel, Mehdi Ammi, and Tobias Isenberg. 2017. Mouse, Tactile, and Tangible Input for 3D Manipulation. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, 4727–4740. <https://doi.org/10.1145/3025453.3025863> 73
- [21] Lonni Besançon, Mickael Sereno, Lingyun Yu, Mehdi Ammi, and Tobias Isenberg. 2019. Hybrid Touch/Tangible Spatial 3D Data Selection. *Computer Graphics Forum* 38, 3 (June 2019), 553–567. <https://doi.org/10.1111/cgf.13710> 73
- [22] Andrea Bianchi and Ian Oakley. 2015. MagnID: Tracking Multiple Magnetic Tokens. In *Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 61–68. <https://doi.org/10.1145/2677199.2680582> 4, 73, 93
- [23] Andrea Bianchi, Ian Oakley, Vassilis Kostakos, and Dong Soo Kwon. 2010. The Phone Lock: Audio and Haptic Shoulder-Surfing Resistant PIN Entry Methods for Mobile Devices. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, 197–200. <https://doi.org/10.1145/1935701.1935740> 76

- [24] Mark Billingham, Ivan Poupyrev, Hirokazu Kato, and Richard May. 2000. Mixing realities in Shared Space: an augmented reality interface for collaborative computing. In *International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, Vol. 3. IEEE, 1641–1644 vol.3. <https://doi.org/10.1109/ICME.2000.871085> 20
- [25] Durrell Bishop. 2009. Visualising and Physicalising the Intangible Product: "What Happened to That Bloke Who Designed the Marble Answer Machine?". In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, Article 2, 1 pages. <https://doi.org/10.1145/1517664.1517667> video: <https://vimeo.com/19930744>. 4
- [26] Rachel Blagojevic and Beryl Plimmer. 2013. CapTUI: geometric drawing with tangibles on a capacitive multi-touch display. In *IFIP Conference on Human-Computer Interaction (INTERACT '13)*. Springer, 511–528. https://doi.org/10.1007/978-3-642-40483-2_37 73
- [27] Quentin Bonnard, Patrick Jermann, Amanda Legge, Frédéric Kaplan, and Pierre Dillenbourg. 2012. Tangible Paper Interfaces: Interpreting Pupils' Manipulations. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. ACM, 133–142. <https://doi.org/10.1145/2396636.2396658> 76
- [28] Ahmet Börütecene, Idil Bostan, Ekin Akyürek, Alpay Sabuncuoğlu, Ilker Temuzkusu, Çağlar Genç, Tilbe Göksun, and Oguzhan Özcan. 2018. Through the Glance Mug: A Familiar Artefact to Support Opportunistic Search in Meetings. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '18)*. ACM, 674–683. <https://doi.org/10.1145/3173225.3173236> 5, 76
- [29] Andy Boucher and William Gaver. 2017. Designing and Making the Data-catchers: Batch Producing Location-Aware Mobile Devices. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)*. ACM, 243–251. <https://doi.org/10.1145/3024969.3024971> 1, 76
- [30] Elodie Bouzbib, Gilles Bailly, Sinan Haliyo, and Pascal Frey. 2020. CoVR: A Large-Scale Force-Feedback Robotic Interface for Non-Deterministic Scenarios in VR. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, 209–222. <https://doi.org/10.1145/3379337.3415891> 101

- [31] Andrew Bragdon, Rob DeLine, Ken Hinckley, and Meredith Ringel Morris. 2011. Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. ACM, 212–221. <https://doi.org/10.1145/2076354.2076393> 73
- [32] Eugenie Brasier, Emmanuel Pietriga, and Caroline Appert. 2021. AR-Enhanced Widgets for Smartphone-Centric Interaction. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction (MobileHCI '21)*. ACM, Article 32, 12 pages. <https://doi.org/10.1145/3447526.3472019> 16, 49
- [33] Emeline Brule, Gilles Bailly, Anke Brock, Frederic Valentin, Grégoire Denis, and Christophe Jouffrais. 2016. MapSense: Multi-Sensory Interactive Maps for Children Living with Visual Impairments. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 445–457. <https://doi.org/10.1145/2858036.2858375> 5
- [34] Leah Buechley, Sue Hendrix, and Mike Eisenberg. 2009. Paints, Paper, and Programs: First Steps toward the Computational Sketchbook. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 9–12. <https://doi.org/10.1145/1517664.1517670> 5
- [35] Oğuz Turan Buruk and Oğuzhan Özcan. 2018. Extracting Design Guidelines for Wearables and Movement in Tabletop Role-Playing Games via a Research Through Design Process. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–13. <https://doi.org/10.1145/3173574.3174087> 4
- [36] Wolfgang Büschel, Ulrike Kister, Mathias Frisch, and Raimund Dachsel. 2014. T4 - Transparent and Translucent Tangibles on Tabletops. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, 81–88. <https://doi.org/10.1145/2598153.2598179> 73
- [37] Alex Butler, Shahram Izadi, and Steve Hodges. 2008. SideSight: Multi-"touch" Interaction around Small Devices. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. ACM, 201–204. <https://doi.org/10.1145/1449715.1449746> 16
- [38] William Buxton. 1990. A Three-State Model of Graphical Input. In *Proceedings of the IFIP TC13 Third International Conference on Human-Computer Interaction (INTERACT '90)*. North-Holland, 449–456. <https://www.dgp.toronto.edu/OTP/papers/bill.buxton/3state.html> 48

- [39] Xiang Cao and Ravin Balakrishnan. 2003. VisionWand: Interaction Techniques for Large Displays Using a Passive Wand Tracked in 3D. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST '03)*. Association for Computing Machinery, 173–182. <https://doi.org/10.1145/964696.964716> 75
- [40] Arpan Chakraborty, Ryan Gross, Shea McIntee, Kyung Wha Hong, Jae Yeol Lee, and Robert St. Amant. 2014. CAPTIVE: A Cube with Augmented Physical Tools. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, 1315–1320. <https://doi.org/10.1145/2559206.2581340> 4
- [41] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. Cap-Stones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2189–2192. <https://doi.org/10.1145/2207676.2208371> ix, 4, 6, 15, 25, 73, 85, 90
- [42] Li-Wei Chan, Hsiang-Tao Wu, Hui-Shan Kao, Ju-Chun Ko, Home-Ru Lin, Mike Y. Chen, Jane Hsu, and Yi-Ping Hung. 2010. Enabling Beyond-Surface Interactions for Interactive Surface with an Invisible Projection. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, 263–272. <https://doi.org/10.1145/1866029.1866072> 93
- [43] Olivier Chapuis, Anastasia Bezerianos, and Stelios Frantzeskakis. 2014. Smarties: An Input System for Wall Display Development. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2763–2772. <https://doi.org/10.1145/2556288.2556956> ix, 9, 16, 19, 24, 47
- [44] Christian Cherek, David Asselborn, Simon Voelker, and Jan Borchers. 2019. Off-Surface Tangibles: Exploring the Design Space of Midair Tangible Interaction. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. ACM, Article LBW2115, 6 pages. <https://doi.org/10.1145/3290607.3312966> 18, 44, 80, 92
- [45] Christian Cherek, Anke Broucker, Simon Voelker, and Jan Borchers. 2018. Tangible Awareness: How Tangibles on Tabletops Influence Awareness of Each Other's Actions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, Article 298, 7 pages. <https://doi.org/10.1145/3173574.3173872> 5, 12, 24, 73
- [46] Christian Cherek, Anke Broucker, Simon Voelker, and Jan Borchers. 2018. Tangible Awareness: How Tangibles on Tabletops Influence Awareness of Each

- Other's Actions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–7. <https://doi.org/10.1145/3173574.3173872> 73
- [47] Christian Cherek, Simon Voelker, Jan Thar, Rene Linden, Florian Busch, and Jan Borchers. 2015. PERCs Demo: Persistently Trackable Tangibles on Capacitive Multi-Touch Displays. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 389–392. <https://doi.org/10.1145/2817721.2823474> 73
- [48] Jean Ho Chu, Paul Clifton, Daniel Harley, Jordanne Pavao, and Ali Mazalek. 2015. Mapping Place: Supporting Cultural Learning through a Lukasa-Inspired Tangible Tabletop Museum Exhibit. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 261–268. <https://doi.org/10.1145/2677199.2680559> 91
- [49] Adrian Clark, Andreas Dünser, Mark Billingham, Thammathip Piumsomboon, and David Altimira. 2011. Seamless Interaction in Space. In *Proceedings of the 23rd Australian Computer-Human Interaction Conference (OzCHI '11)*. ACM, 88–97. <https://doi.org/10.1145/2071536.2071549> 73
- [50] Jorgos Coenen, Sandy Claes, and Andrew Vande Moere. 2017. The Concurrent Use of Touch and Mid-Air Gestures or Floor Mat Interaction on a Public Display. In *Proceedings of the 6th ACM International Symposium on Pervasive Displays (PerDis '17)*. ACM, Article 9, 9 pages. <https://doi.org/10.1145/3078810.3078819> ix, 13, 101
- [51] Haley Coppins, Tudor Tibu, Jack Shen-Kuen Chang, Ali Mazalek, and Frauke Zeller. 2016. Combining Mobile, Tangible and Virtual World Platforms to Support Participatory Campus Planning. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, 325–330. <https://doi.org/10.1145/2992154.2996775> ix, 9, 91
- [52] Maxime Cordeil, Benjamin Bach, Andrew Cunningham, Bastian Montoya, Ross T. Smith, Bruce H. Thomas, and Tim Dwyer. 2020. Embodied Axes: Tangible, Actuated Interaction for 3D Augmented Reality Data Spaces. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–12. <https://doi.org/10.1145/3313831.3376613> 20
- [53] Emmanuel Courtoux, Caroline Appert, and Olivier Chapuis. 2021. WallTokens: Surface Tangibles for Vertical Displays. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. ACM, Article 421, 13 pages. <https://doi.org/10.1145/3411764.3445404> 5, 20, 21, 23, 45, 47, 51, 53, 64, 67, 70, 73, 98

- [54] Emmanuel Courtoux, Caroline Appert, and Olivier Chapuis. 2023. SurfAirs: Surface + Mid-air Input for Large Vertical Displays. In *Proceedings of the international conference on Human factors in computing systems (CHI '23)*. ACM, 15 pages. to appear. 45, 98
- [55] Céline Coutrix and Yann Laurillau. 2019. Which Tangible Control for Which Visual Task?. In *Proceedings of the 31st Conference on l'Interaction Homme-Machine (IHM '19)*. ACM, Article 3, 8 pages. <https://doi.org/10.1145/3366550.3372249> 5, 73
- [56] Josh Urban Davis, Te-Yen Wu, Bo Shi, Hanyi Lu, Athina Panotopoulou, Emily Whiting, and Xing-Dong Yang. 2020. TangibleCircuits: An Interactive 3D Printed Circuit Education Tool for People with Visual Impairments. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–13. <https://doi.org/10.1145/3313831.3376513> 5
- [57] Alexandre G. de Siqueira, Brygg Ullmer, Mark Delarosa, Chris Branton, and Miriam K. Konkel. 2018. Hard and Soft Tangibles: Mixing Multi-Touch and Tangible Interaction in Scientific Poster Scenarios. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '18)*. ACM, 476–486. <https://doi.org/10.1145/3173225.3173252> 90, 91
- [58] Jialin Deng, Patrick Olivier, Josh Andres, Kirsten Ellis, Ryan Wee, and Florian Floyd Mueller. 2022. Logic Bonbon: Exploring Food as Computational Artifact. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. ACM, Article 47, 21 pages. <https://doi.org/10.1145/3491102.3501926> 4
- [59] Karen Detken, Carlos Martinez, and Andreas Schrader. 2009. The Search Wall: Tangible Information Searching for Children in Public Libraries. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 289–296. <https://doi.org/10.1145/1517664.1517724> 5
- [60] Paul Dourish and Victoria Bellotti. 1992. Awareness and Coordination in Shared Workspaces. In *Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work (CSCW '92)*. ACM, 107–114. <https://doi.org/10.1145/143457.143468> 12
- [61] Julie Ducasse, Marc J-M Macé, Marcos Serrano, and Christophe Jouffrais. 2016. Tangible Reels: Construction and Exploration of Tangible Maps by Visually Impaired Users. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 2186–2197. <https://doi.org/10.1145/2858036.2858058> 5

- [62] John Dudley, Hrovje Benko, Daniel Wigdor, and Per Ola Kristensson. 2019. Performance Envelopes of Virtual Keyboard Text Input Strategies in Virtual Reality. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR '19)*. IEEE, 289–300. <https://doi.org/10.1109/ISMAR.2019.0002732>
- [63] Lisa A. Elkin, Jean-Baptiste Beau, Géry Casiez, and Daniel Vogel. 2020. Manipulation, Learning, and Recall with Tangible Pen-Like Input. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–12. <https://doi.org/10.1145/3313831.3376772> 92, 93
- [64] D. Englmeier, J. Dörner, A. Butz, and T. Höllerer. 2020. A Tangible Spherical Proxy for Object Manipulation in Augmented Reality. In *Conference on Virtual Reality and 3D User Interfaces (VR '20)*. IEEE, 221–229. <https://doi.org/10.1109/VR46266.2020.00041> ix, 4, 5, 20, 49
- [65] Barrett Ens, Sarah Goodwin, Arnaud Prouzeau, Fraser Anderson, Florence Y. Wang, Samuel Gratzl, Zac Lucarelli, Brendan Moyle, Jim Smiley, and Tim Dwyer. 2021. Uplift: A Tangible and Immersive Tabletop System for Casual Collaborative Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2021), 1193–1203. <https://doi.org/10.1109/TVCG.2020.3030334> 6, 8, 9, 20, 73, 84, 91, 93
- [66] Min Fan, Alissa N. Antle, Carman Neustaedter, and Alyssa F. Wise. 2014. Exploring How a Co-Dependent Tangible Tool Design Supports Collaboration in a Tabletop Activity. In *Proceedings of the 18th International Conference on Supporting Group Work (GROUP '14)*. ACM, 81–90. <https://doi.org/10.1145/2660398.2660402> 5, 11, 24, 73
- [67] Cathy Fang, Yang Zhang, Matthew Dworman, and Chris Harrison. 2020. Wireality: Enabling Complex Tangible Geometries in Virtual Reality with Worn Multi-String Haptics. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–10. <https://doi.org/10.1145/3313831.3376470> 5
- [68] Cathy Mengying Fang and Chris Harrison. 2021. Retargeted Self-Haptics for Increased Immersion in VR without Instrumentation. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. ACM, 1109–1121. <https://doi.org/10.1145/3472749.3474810> 4
- [69] George W. Fitzmaurice and William Buxton. 1997. An Empirical Evaluation of Graspable User Interfaces: Towards Specialized, Space-Multiplexed Input. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing*

- Systems (CHI '97)*. ACM, 43–50. <https://doi.org/10.1145/258549.258578>
5, 12, 20, 24, 73
- [70] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. 1995. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM & Addison-Wesley, 442–449. <https://doi.org/10.1145/223904.223964> 75
- [71] Sean Follmer, Micah Johnson, Edward Adelson, and Hiroshi Ishii. 2011. DeForm: An Interactive Malleable Surface for Capturing 2.5D Arbitrary Objects, Tools and Touch. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 527–536. <https://doi.org/10.1145/2047196.2047265> 79
- [72] Mike Fraser, Michael R. McCarthy, Muneeb Shaukat, and Phillip Smith. 2007. Seconds Matter: Improving Distributed Coordination Bytracking and Visualizing Display Trajectories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, 1303–1312. <https://doi.org/10.1145/1240624.1240822> 20
- [73] Jérémy Frey, Renaud Gervais, Stéphanie Fleck, Fabien Lotte, and Martin Hachet. 2014. Teegi: Tangible EEG Interface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 301–308. <https://doi.org/10.1145/2642918.2647368> 5
- [74] Bernd Fröhlich and John Plate. 2000. The Cubic Mouse: A New Device for Three-Dimensional Input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, 526–531. <https://doi.org/10.1145/332040.332491> 20
- [75] Renaud Gervais, Jérémy Frey, Alexis Gay, Fabien Lotte, and Martin Hachet. 2016. TOBE: Tangible Out-of-Body Experience. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, 227–235. <https://doi.org/10.1145/2839462.2839486> 1, 5, 76
- [76] Renaud Gervais, Joan Sol Roo, and Martin Hachet. 2016. Tangible Viewports: Getting Out of Flatland in Desktop Environments. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, 176–184. <https://doi.org/10.1145/2839462.2839468> 79
- [77] Tom Giraud, Brian Ravenet, Chi Tai Dang, Jacqueline Nadel, Elise Prigent, Gael Poli, Elisabeth Andre, and Jean-claude Martin. 2021. “Can You Help Me Move

- This over There?": Training Children with ASD to Joint Action through Tangible Interaction and Virtual Agent. In *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '21)*. ACM, Article 27, 12 pages. <https://doi.org/10.1145/3430524.3440646> 93
- [78] Alessandro Gnoli, Anthony Perritano, Paulo Guerra, Brenda Lopez, Joel Brown, and Tom Moher. 2014. Back to the Future: Embodied Classroom Simulations of Animal Foraging. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction (TEI '14)*. ACM, 275–282. <https://doi.org/10.1145/2540930.2540972> 6, 73, 90
- [79] Antonio Gomes, Calvin Rubens, Sean Braley, and Roel Vertegaal. 2016. Bit-Drones: Towards Using 3D Nanocopter Displays as Interactive Self-Levitating Programmable Matter. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 770–780. <https://doi.org/10.1145/2858036.2858519> ix, 4, 5, 6, 73, 100
- [80] Tovi Grossman, Daniel Wigdor, and Ravin Balakrishnan. 2004. Multi-Finger Gestural Interaction with 3d Volumetric Displays. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, 61–70. <https://doi.org/10.1145/1029632.1029644> 19
- [81] Casey Grote, Evan Segreto, Johanna Okerlund, Robert Kincaid, and Orit Shaer. 2015. Eugenie: Multi-Touch and Tangible Interaction for Bio-Design. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 217–224. <https://doi.org/10.1145/2677199.2680605> 92, 93
- [82] Carl Gutwin and Saul Greenberg. 2004. The importance of awareness for team cognition in distributed collaboration. In *Team cognition: Understanding the factors that drive process and performance*. APA, 177–201. <https://doi.org/10.1037/10690-009> 12
- [83] Charlotte Häger-Ross and Marc H. Schieber. 2000. Quantifying the Independence of Human Finger Movements: Comparisons of Digits, Hands, and Movement Frequencies. *Journal of Neuroscience* 20, 22 (2000), 8542–8550. <https://doi.org/10.1523/JNEUROSCI.20-22-08542.2000> arXiv:<https://www.jneurosci.org/content/20/22/8542.full.pdf> 52
- [84] Changyo Han, Ryo Takahashi, Yuchi Yahagi, and Takeshi Naemura. 2020. Pneu-Module: Using Inflatable Pin Arrays for Reconfigurable Physical Controls on Pressure-Sensitive Touch Surfaces. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–14. <https://doi.org/10.1145/3313831.3376838> 73, 79

- [85] Mark Hancock, Otmar Hilliges, Christopher Collins, Dominikus Baur, and Sheelagh Carpendale. 2009. Exploring Tangible and Direct Touch Interfaces for Manipulating 2D and 3D Information on a Digital Table. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, 77–84. <https://doi.org/10.1145/1731903.1731921> 5, 12, 20, 24, 32, 73, 93
- [86] Björn Hartmann, Meredith Ringel Morris, Hrvoje Benko, and Andrew D. Wilson. 2010. Pictionaire: Supporting Collaborative Design Work by Integrating Physical and Digital Artifacts. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*. ACM, 421–424. <https://doi.org/10.1145/1718918.1718989> 11, 73
- [87] Khalad Hasan, David Ahlström, Junhyeok Kim, and Pourang Irani. 2017. Air-Panes: Two-Handed Around-Device Interaction for Pane Switching on Smartphones. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, 679–691. <https://doi.org/10.1145/3025453.3026029> 49
- [88] Doris Hausen, Sebastian Boring, Clara Lueling, Simone Rodestock, and Andreas Butz. 2012. StaTube: Facilitating State Management in Instant Messaging Systems. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, 283–290. <https://doi.org/10.1145/2148131.2148191> 79
- [89] Kirstie Hawkey, Melanie Kellar, Derek Reilly, Tara Whalen, and Kori M. Inkpen. 2005. The Proximity Factor: Impact of Distance on Co-Located Collaboration. In *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work (GROUP '05)*. ACM, 31–40. <https://doi.org/10.1145/1099203.1099209> 17
- [90] Fabian Hennecke, Raphael Wimmer, Eduard Vodicka, and Andreas Butz. 2012. Vertibles: Using Vacuum Self-Adhesion to Create a Tangible User Interface for Arbitrary Interactive Surfaces. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, 303–306. <https://doi.org/10.1145/2148131.2148194> 16, 20, 21, 23, 47, 73, 79, 92
- [91] Otmar Hilliges, Shahram Izadi, Andrew D. Wilson, Steve Hodges, Armando Garcia-Mendoza, and Andreas Butz. 2009. Interactions in the Air: Adding Further Depth to Interactive Tabletops. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 139–148. <https://doi.org/10.1145/1622176.1622203> 17
- [92] Uta Hinrichs, Simon Butscher, Jens Müller, and Harald Reiterer. 2016. Diving in at the Deep End: The Value of Alternative In-Situ Approaches for Systematic

- Library Search. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 4634–4646. <https://doi.org/10.1145/2858036.2858549> 84, 91
- [93] Eve Hoggan, John Williamson, Antti Oulasvirta, Miguel Nacenta, Per Ola Kristensson, and Anu Lehtiö. 2013. Multi-Touch Rotation Gestures: Performance and Ergonomics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 3047–3050. <https://doi.org/10.1145/2470654.2481423> 36
- [94] Jonathan Hook, Stuart Taylor, Alex Butler, Nicolas Villar, and Shahram Izadi. 2009. A Reconfigurable Ferromagnetic Input Device. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 51–54. <https://doi.org/10.1145/1622176.1622186> 79
- [95] Eva Hornecker and Jacob Buur. 2006. Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, 437–446. <https://doi.org/10.1145/1124772.1124838> 16
- [96] Takuma Hosokawa, Yasuhiko Takeda, Norio Shioiri, Mitsunori Hirano, and Kazuhiko Tanaka. 2008. Tangible Design Support System Using RFID Technology. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction (TEI '08)*. ACM, 75–78. <https://doi.org/10.1145/1347390.1347408> 6, 15, 73
- [97] Kori Inkpen, Kirstie Hawkey, Melanie Kellar, Regan Mandryk, Karen Parker, Derek Reilly, Stacey Scott, and Tara Whalen. 2005. Exploring display factors that influence co-located collaboration: angle, size, number, and user arrangement. In *Proc. HCI international*, Vol. 2005. 10 pages. https://people.ece.ubc.ca/hawkey/HCII2005_Inkpen.pdf 13
- [98] Yoshio Ishiguro, Ali Israr, Alex Rothera, and Eric Brockmeyer. 2014. Uminari: Freeform Interactive Loudspeakers. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14)*. ACM, 55–64. <https://doi.org/10.1145/2669485.2669521> 5
- [99] Hiroshi Ishii and Brygg Ullmer. 1997. 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)*. ACM, 234–241. <https://doi.org/10.1145/258549.258715> 14, 20
- [100] Shahram Izadi, Steve Hodges, Stuart Taylor, Dan Rosenfeld, Nicolas Villar, Alex Butler, and Jonathan Westhues. 2008. 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)*. ACM, 269–278. <https://doi.org/10.1145/1449715.1449760> 17, 93
- [101] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen. 1994. Integrality and Separability of Input Devices. *ACM Trans. Comput.-Hum. Interact.* 1, 1 (March 1994), 3–26. <https://doi.org/10.1145/174630.174631> 37, 38, 60
- [102] Mikkel R. Jakobsen and Kasper Hornbæk. 2014. Up Close and Personal: Collaborative Work on a High-Resolution Multitouch Wall Display. *ACM Trans. Comput.-Hum. Interact.* 21, 2, Article 11 (feb 2014), 34 pages. <https://doi.org/10.1145/2576099> 14
- [103] Mikkel R. Jakobsen and Kasper Hornbæk. 2014. Up Close and Personal: Collaborative Work on a High-Resolution Multitouch Wall Display. *ACM Trans. Comput.-Hum. Interact.* 21, 2, Article 11 (feb 2014), 34 pages. <https://doi.org/10.1145/2576099> 17, 42, 46
- [104] Mikkel R. Jakobsen and Kasper Hornbæk. 2015. Is Moving Improving? Some Effects of Locomotion in Wall-Display Interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 4169–4178. <https://doi.org/10.1145/2702123.2702312> 17, 19
- [105] Mikkel R. Jakobsen and Kasper Hornbæk. 2016. Negotiating for Space? Collaborative Work Using a Wall Display with Mouse and Touch Input. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 2050–2061. <https://doi.org/10.1145/2858036.2858158> 17
- [106] Mikkel R. Jakobsen, Yvonne Jansen, Sebastian Boring, and Kasper Hornbæk. 2015. Should I Stay or Should I Go? Selecting Between Touch and Mid-Air Gestures for Large-Display Interaction. In *IFIP Conference on Human-Computer Interaction (INTERACT '15)*. Springer, 455–473. https://doi.org/10.1007/978-3-319-22698-9_31 17, 21, 54, 69
- [107] Mikkel R. Jakobsen, Yonas Sahlemariam Haile, Søren Knudsen, and Kasper Hornbæk. 2013. Information Visualization and Proxemics: Design Opportunities and Empirical Findings. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2386–2395. <https://doi.org/10.1109/TVCG.2013.166> 13, 19
- [108] Raphaël James, Anastasia Bezerianos, and Olivier Chapuis. 2023. Evaluating the Extension of Wall Displays with AR for Collaborative Work. In *Proceedings*

- of the international conference on Human factors in computing systems (CHI '23)*. ACM, 17 pages. to appear. ix, 9
- [109] Raphaël James, Anastasia Bezerianos, Olivier Chapuis, Tim Dwyer, Maxime Cordeil, and Arnaud Prouzeau. 2020. Personal+Context Navigation: Combining AR and Shared Displays in Network Path-following. In *Proceedings of Graphics Interface (GI '20)*. CHCCS/SCDHM, 13 pages. <https://doi.org/10.20380/GI2020.27> 8, 9
- [110] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2012. Tangible Remote Controllers for Wall-Size Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2865–2874. <https://doi.org/10.1145/2207676.2208691> x, 5, 9, 19, 24, 25, 45, 47, 73, 76, 91
- [111] Yunwoo Jeong, Han-Jong Kim, Gyeongwon Yun, and Tek-Jin Nam. 2020. WIKA: A Projected Augmented Reality Workbench for Interactive Kinetic Art. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. ACM, 999–1009. <https://doi.org/10.1145/3379337.3415880> 93
- [112] Hans-Christian Jetter, Jens Gerken, Michael Zöllner, Harald Reiterer, and Natasa Milic-Frayling. 2011. Materializing the Query with Facet-streams: A Hybrid Surface for Collaborative Search on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 3013–3022. <https://doi.org/10.1145/1978942.1979390> 44
- [113] Brett Jones, Rajinder Sodhi, David Forsyth, Brian Bailey, and Giuliano Maciocci. 2012. Around Device Interaction for Multiscale Navigation. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, 83–92. <https://doi.org/10.1145/2371574.2371589> 16
- [114] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The ReactTable: Exploring the Synergy between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, 139–146. <https://doi.org/10.1145/1226969.1226998> 3, 6, 11, 20, 24, 42, 44, 47, 73
- [115] Ricardo Jota, Miguel A. Nacenta, Joaquim A. Jorge, Sheelagh Carpendale, and Saul Greenberg. 2010. A Comparison of Ray Pointing Techniques for Very Large Displays. In *Proceedings of Graphics Interface 2010 (GI '10)*. CIPS, 269–276. <https://dl.acm.org/doi/abs/10.5555/1839214.1839261> 56

- [116] Jamil L. Joundi, Arno Penders, Johanna Renny Octavia, and Jelle Saldien. 2019. The Design of an Interactive Surface for Supporting Rehabilitation of Children with Developmental Coordination Disorder. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '19)*. ACM, 335–344. <https://doi.org/10.1145/3294109.3295648> 101
- [117] Martin Kaltenbrunner, Till Bovermann, Ross Bencina, and Enrico Costanza. 2005. TUIO - A Protocol for Table Based Tangible User Interfaces. In *Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation (GW '05)*. 5 pages pages. 40
- [118] Hirokazu Kato, Mark Billinghurst, Ivan Poupyrev, K. Imamoto, and K. Tachibana. 2000. Virtual object manipulation on a table-top AR environment. In *Proceedings of the International Symposium on Augmented Reality (ISAR 2000)*. IEEE & ACM, 111–119. <https://doi.org/10.1109/ISAR.2000.880934> 20
- [119] Rebecca Kleinberger and Alisha Panjwani. 2018. Digitally Enchanted Wear: A Novel Approach in the Field of Dresses as Dynamic Digital Displays. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '18)*. ACM, 27–34. <https://doi.org/10.1145/3173225.3173250> 4
- [120] Varsha Koushik, Darren Guinness, and Shaun K. Kane. 2019. StoryBlocks: A Tangible Programming Game To Create Accessible Audio Stories. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–12. <https://doi.org/10.1145/3290605.3300722> 5, 85
- [121] Sven Kratz and Michael Rohs. 2009. HoverFlow: Expanding the Design Space of around-Device Interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, Article 4, 8 pages. <https://doi.org/10.1145/1613858.1613864> 16
- [122] Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. 2011. CapWidgets: Tangible Widgets Versus Multi-touch Controls on Mobile Devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, 1351–1356. <https://doi.org/10.1145/1979742.1979773> 25
- [123] Ricardo Langner, Ulrike Kister, and Raimund Dachsel. 2019. Multiple Coordinated Views at Large Displays for Multiple Users: Empirical Findings on User Behavior, Movements, and Distances. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (jan 2019), 608–618. <https://doi.org/10.1109/TVCG.2018.2865235> 18, 19, 45, 73, 87

- [124] Seung Ah Lee, Alice M. Chung, Nate Cira, and Ingmar H. Riedel-Kruse. 2015. Tangible Interactive Microbiology for Informal Science Education. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 273–280. <https://doi.org/10.1145/2677199.2680561> 76
- [125] Sang-won Leigh, Philipp Schoessler, Felix Heibeck, Pattie Maes, and Hiroshi Ishii. 2015. THAW: Tangible Interaction with See-Through Augmentation for Smartphones on Computer Screens. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 89–96. <https://doi.org/10.1145/2677199.2680584> 1, 76
- [126] Jakob Leitner and Michael Haller. 2011. Geckos: Combining Magnets and Pressure Images to Enable New Tangible-Object Design and Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 2985–2994. <https://doi.org/10.1145/1978942.1979385> 6, 7, 16, 20, 21, 44, 47, 73, 92
- [127] Rong-Hao Liang, Liwei Chan, Hung-Yu Tseng, Han-Chih Kuo, Da-Yuan Huang, De-Nian Yang, and Bing-Yu Chen. 2014. GaussBricks: Magnetic Building Blocks for Constructive Tangible Interactions on Portable Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 3153–3162. <https://doi.org/10.1145/2556288.2557105> 85
- [128] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-Free near-Surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 1391–1400. <https://doi.org/10.1145/2470654.2466185> 15, 100
- [129] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-free Near-surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 1391–1400. <https://doi.org/10.1145/2470654.2466185> 91, 93
- [130] Rong-Hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. 2014. GaussStones: Shielded Magnetic Tangibles for Multi-token Interactions on Portable Displays. In *Proceedings of the 27th ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 365–372. <https://doi.org/10.1145/2642918.2647384> 6, 73, 93

- [131] Rong-Hao Liang, Bin Yu, Mengru Xue, Jun Hu, and Loe M. G. Feijs. 2018. BioFidget: Biofeedback for Respiration Training Using an Augmented Fidget Spinner. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–12. <https://doi.org/10.1145/3173574.3174187> 5
- [132] David Lindlbauer, Joerg Mueller, and Marc Alexa. 2016. Changing the Appearance of Physical Interfaces Through Controlled Transparency. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, 425–435. <https://doi.org/10.1145/2984511.2984556> 5
- [133] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, and Eric Lecolinet. 2016. Shared Interaction on a Wall-Sized Display in a Data Manipulation Task. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 2075–2086. <https://doi.org/10.1145/2858036.2858039> 43, 101
- [134] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, and Eric Lecolinet. 2017. CoReach: Cooperative Gestures for Data Manipulation on Wall-Sized Displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, 6730–6741. <https://doi.org/10.1145/3025453.3025594> 24, 46
- [135] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, Eric Lecolinet, and Wendy E. Mackay. 2014. 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)*. ACM, 4147–4156. <https://doi.org/10.1145/2556288.2557020> 17, 19, 42
- [136] Mingyu Liu, Mathieu Nancel, and Daniel Vogel. 2015. Gunslinger: Subtle Arms-down Mid-Air Interaction. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, 63–71. <https://doi.org/10.1145/2807442.2807489> 18, 19, 45
- [137] Takuya Maekawa, Yuichi Itoh, Norifumi Kawai, Yoshifumi Kitamura, and Fumio Kishino. 2009. MADO Interface: A Window like a Tangible User Interface to Look into the Virtual World. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 175–180. <https://doi.org/10.1145/1517664.1517704> 6, 73
- [138] Pierre Mahieux, Romain Biannic, Sébastien Kubicki, and Ronan Querrec. 2022. SABLIER: A Tangible Interactor to Navigate through Space and Time. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. ACM, Article 503, 12 pages. <https://doi.org/10.1145/3491102.3517567> 5

- [139] Shahzad Malik, Abhishek Ranjan, and Ravin Balakrishnan. 2005. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In *Proceedings of the 18th annual ACM symposium on User interface software and technology (UIST '05)*. ACM, 43–52. 9
- [140] Roozbeh Manshaei, Nauman Baig, Sean Delong, Shahin Khayyer, Brien East, and Ali Mazalek. 2017. Tangible MtDNA: A Tangible Tabletop System for Exploring Genetic Mutations on Mitochondrial DNA Cancer Data. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)*. ACM, 101–110. <https://doi.org/10.1145/3024969.3025005> 5, 86, 87, 93
- [141] Valérie Maquil, Christian Moll, and João Martins. 2017. In the Footsteps of Henri Tudor: Creating Batteries on a Tangible Interactive Workbench. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces (ISS '17)*. ACM, 252–259. <https://doi.org/10.1145/3132272.3134115> 5
- [142] Karola Marky, Martin Schmitz, Verena Zimmermann, Martin Herbers, Kai Kunze, and Max Mühlhäuser. 2020. 3D-Auth: Two-Factor Authentication with Personalized 3D-Printed Items. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–12. <https://doi.org/10.1145/3313831.3376189> 73
- [143] Nicolai Marquardt, Ricardo Jota, Saul Greenberg, and Joaquim A. Jorge. 2011. The Continuous Interaction Space: Interaction Techniques Unifying Touch and Gesture on and above a Digital Surface. In *IFIP Conference on Human-Computer Interaction (INTERACT '11)*. Springer, 461–476. https://doi.org/10.1007/978-3-642-23765-2_32 17, 47, 49
- [144] Mark Marshall, Thomas Carter, Jason Alexander, and Sriram Subramanian. 2012. Ultra-Tangibles: Creating Movable Tangible Objects on Interactive Tables. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2185–2188. <https://doi.org/10.1145/2207676.2208370> 75
- [145] Asier Marzo, Richard McGeehan, Jess McIntosh, Sue Ann Seah, and Sriram Subramanian. 2015. Ghost Touch: Turning Surfaces into Interactive Tangible Canvases with Focused Ultrasound. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 137–140. <https://doi.org/10.1145/2817721.2817727> 76
- [146] Maurice R. Masliah and Paul Milgram. 2000. Measuring the Allocation of Control in a 6 Degree-of-Freedom Docking Experiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, 25–32. <https://doi.org/10.1145/332040.332403> 38

- [147] Fabrice Matulic and Daniel Vogel. 2018. Multiray: Multi-Finger Raycasting for Large Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–13. <https://doi.org/10.1145/3173574.3173819> x, 6, 18, 19, 46
- [148] Christopher McAdam and Stephen Brewster. 2011. Using Mobile Phones to Interact with Tabletop Computers. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. ACM, 232–241. <https://doi.org/10.1145/2076354.2076395> 73, 76
- [149] David C. McCallum and Pourang Irani. 2009. ARC-Pad: Absolute+relative Cursor Positioning for Large Displays with a Mobile Touchscreen. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 153–156. <https://doi.org/10.1145/1622176.1622205> 19, 47
- [150] Meghna Mehta, Ahmed Sabbir Arif, Apurva Gupta, Sean DeLong, Roozbeh Manshaei, Graceline Williams, Manasvi Lalwani, Sanjay Chandrasekharan, and Ali Mazalek. 2016. Active Pathways: Using Active Tangibles and Interactive Tabletops for Collaborative Modeling in Systems Biology. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, 129–138. <https://doi.org/10.1145/2992154.2992176> 76
- [151] Haipeng Mi and Masanori Sugimoto. 2011. HATs: Interact Using Height-Adjustable Tangibles in Tabletop Interfaces. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. ACM, 71–74. <https://doi.org/10.1145/2076354.2076368> 93
- [152] Hila Mor, Tianyu Yu, Ken Nakagaki, Benjamin Harvey Miller, Yichen Jia, and Hiroshi Ishii. 2020. Venous Materials: Towards Interactive Fluidic Mechanisms. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–14. <https://doi.org/10.1145/3313831.3376129> 5
- [153] Rafael Morales, Asier Marzo, Sriram Subramanian, and Diego Martínez. 2019. LeviProps: Animating Levitated Optimized Fabric Structures Using Holographic Acoustic Tweezers. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. ACM, 651–661. <https://doi.org/10.1145/3332165.3347882> 5, 76
- [154] Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. TouchTokens: Guiding Touch Patterns with Passive Tokens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 4189–4202. <https://doi.org/10.1145/2858036.2858041> 20, 24, 25, 28, 53

- [155] Tao Morisaki, Ryoma Mori, Ryosuke Mori, Yasutoshi Makino, Yuta Itoh, Yuji Yamakawa, and Hiroyuki Shinoda. 2019. Hopping-Pong: Changing Trajectory of Moving Object Using Computational Ultrasound Force. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces (ISS '19)*. ACM, 123–133. <https://doi.org/10.1145/3343055.3359701> 4, 5
- [156] M. R. Morris, A. J. B. Brush, and B. R. Meyers. 2008. A field study of knowledge workers' use of interactive horizontal displays. In *3rd International Workshop on Horizontal Interactive Human Computer Systems (Tabletop '08)*. IEEE, 105–112. <https://doi.org/10.1109/TABLETOP.2008.4660192> 12
- [157] Thomas Muender, Anke V. Reinschluessel, Sean Drewes, Dirk Wenig, Tanja Döring, and Rainer Malaka. 2019. Does It Feel Real? Using Tangibles with Different Fidelities to Build and Explore Scenes in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–12. <https://doi.org/10.1145/3290605.3300903> 4, 5
- [158] Jörg Müller, Gilles Bailly, Thor Bossuyt, and Niklas Hillgren. 2014. MirrorTouch: Combining Touch and Mid-Air Gestures for Public Displays. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, 319–328. <https://doi.org/10.1145/2628363.2628379> 18
- [159] Jens Müller, Tobias Schwarz, Simon Butscher, and Harald Reiterer. 2014. Back to Tangibility: A Post-WIMP Perspective on Control Room Design. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, 57–64. <https://doi.org/10.1145/2598153.2598161> 84, 92
- [160] Brad A. Myers, Rishi Bhatnagar, Jeffrey Nichols, Choon Hong Peck, Dave Kong, Robert Miller, and A. Chris Long. 2002. Interacting at a Distance: Measuring the Performance of Laser Pointers and Other Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, 33–40. <https://doi.org/10.1145/503376.503383> 19
- [161] Miguel A. Nacenta, Patrick Baudisch, Hrvoje Benko, and Andy Wilson. 2009. Separability of Spatial Manipulations in Multi-Touch Interfaces. In *Proceedings of Graphics Interface 2009 (GI '09)*. CIPS, 175–182. <https://dl.acm.org/doi/abs/10.5555/1555880.1555919> 44
- [162] Miguel A. Nacenta, Carl Gutwin, Dzmitry Aliakseyeu, and Sriram Subramanian. 2009. There and Back Again: Cross-Display Object Movement in Multi-Display Environments. *Human-Computer Interaction* 24, 1-2 (2009), 170–229. <https://doi.org/10.1080/07370020902819882> arXiv:<https://doi.org/10.1080/07370020902819882> 17

- [163] Miguel A. Nacenta, David Pinelle, Dane Stuckel, and Carl Gutwin. 2007. The Effects of Interaction Technique on Coordination in Tabletop Groupware. In *Proceedings of Graphics Interface 2007 (GI '07)*. ACM, 191–198. <https://doi.org/10.1145/1268517.1268550> 20
- [164] Bruno Nadeau and Amanda Williams. 2009. Tactful Interaction: Exploring Interactive Social Touch through a Collaborative Tangible Installation. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 147–152. <https://doi.org/10.1145/1517664.1517700> 5, 73
- [165] Ken Nakagaki, Jordan L Tappa, Yi Zheng, Jack Forman, Joanne Leong, Sven Koenig, and Hiroshi Ishii. 2022. (Dis)Appearables: A Concept and Method for Actuated Tangible UIs to Appear and Disappear Based on Stages. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. ACM, Article 506, 13 pages. <https://doi.org/10.1145/3491102.3501906> 84, 91
- [166] Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P. Irani, and Michel Beaudouin-Lafon. 2013. High-Precision Pointing on Large Wall Displays Using Small Handheld Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 831–840. <https://doi.org/10.1145/2470654.2470773> 19
- [167] Mathieu Nancel, Julie Wagner, Emmanuel Pietriga, Olivier Chapuis, and Wendy Mackay. 2011. Mid-Air Pan-and-Zoom on Wall-Sized Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 177–186. <https://doi.org/10.1145/1978942.1978969> 16, 19, 24, 46
- [168] Diana Nowacka, Karim Ladha, Nils Y. Hammerla, Daniel Jackson, Cassim Ladha, Enrico Rukzio, and Patrick Olivier. 2013. Touchbugs: Actuated Tangibles on Multi-Touch Tables. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 759–762. <https://doi.org/10.1145/2470654.2470761> 5
- [169] Rafael Nunes, Fabio Rito, and Carlos Duarte. 2015. TACTIC: An API for Touch and Tangible Interaction. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 125–132. <https://doi.org/10.1145/2677199.2680592> 73, 92, 94
- [170] Halla Olafsdottir, Vladimir M Zatsiorsky, and Mark L Latash. 2005. Is the thumb a fifth finger? A study of digit interaction during force production tasks. *Experimental brain research* 160, 2 (2005), 203–213. <https://doi.org/10.1007/s00221-004-2004-0> 52

- [171] Halla B. Olafsdottir, Theophanis Tsandilas, and Caroline Appert. 2014. Prospective Motor Control on Tabletops: Planning Grasp for Multitouch Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2893–2902. <https://doi.org/10.1145/2556288.2557029> 36
- [172] Patrick Parzer, Adwait Sharma, Anita Vogl, Jürgen Steimle, Alex Olwal, and Michael Haller. 2017. SmartSleeve: Real-Time Sensing of Surface and Deformation Gestures on Flexible, Interactive Textiles, Using a Hybrid Gesture Detection Pipeline. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, 565–577. <https://doi.org/10.1145/3126594.3126652> 4
- [173] Esben Warming Pedersen and Kasper Hornbæk. 2011. Tangible Bots: Interaction with Active Tangibles in Tabletop Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 2975–2984. <https://doi.org/10.1145/1978942.1979384> 5
- [174] Nadya Peek, David Pitman, and Richard The. 2009. Hangsters: Tangible Peripheral Interactive Avatars for Instant Messaging. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 25–26. <https://doi.org/10.1145/1517664.1517675> 5, 101
- [175] Gary Perelman, Marcos Serrano, Christophe Bortolaso, Celia Picard, Mustapha Derras, and Emmanuel Dubois. 2019. Combining Tablets with Smartphones for Data Analytics. In *IFIP Conference on Human-Computer Interaction (INTERACT '19)*. Springer, 439–460. https://doi.org/10.1007/978-3-030-29390-1_24 21
- [176] Daniela Petrelli, Alessandro Soranzo, Luigina Ciolfi, and John Reidy. 2016. Exploring the Aesthetics of Tangible Interaction: Experiments on the Perception of Hybrid Objects. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, 100–108. <https://doi.org/10.1145/2839462.2839478> 4
- [177] Pol Pla and Pattie Maes. 2013. Display Blocks: A Set of Cubic Displays for Tangible, Multi-Perspective Data Exploration. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13)*. ACM, 307–314. <https://doi.org/10.1145/2460625.2460677> 5
- [178] Dominic Potts, Martynas Dabravalskis, and Steven Houben. 2022. Tangible-Touch: A Toolkit for Designing Surface-Based Gestures for Tangible Interfaces. In *Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '22)*. ACM, Article 39, 14 pages. <https://doi.org/10.1145/3490149.3502263> 73, 90

- [179] Brianna Potvin, Colin Swindells, Melanie Tory, and Margaret-Anne Storey. 2012. Comparing Horizontal and Vertical Surfaces for a Collaborative Design Task. *Adv. in Hum.-Comp. Int.* 2012, Article 6 (Jan. 2012), 1 pages. <https://doi.org/10.1155/2012/137686> 14
- [180] Sara Price, Taciana Pontual Falcão, Jennifer G. Sheridan, and George Roussos. 2009. The Effect of Representation Location on Interaction in a Tangible Learning Environment. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 85–92. <https://doi.org/10.1145/1517664.1517689> 91
- [181] Arnaud Prouzeau, Anastasia Bezerianos, and Olivier Chapuis. 2015. Road Traffic Monitoring on a Wall Display. In *Proceedings of the 27th Conference on l'Interaction Homme-Machine (IHM '15)*. ACM, Article 29, 6 pages. <https://doi.org/10.1145/2820619.2825009> 3
- [182] Fateme Rajabiyazdi, Jagoda Walny, Carrie Mah, John Brosz, and Sheelagh Carpendale. 2015. Understanding Researchers' Use of a Large, High-Resolution Display Across Disciplines. In *Proceedings of the International Conference on Interactive Tabletops & Surfaces (ITS '15)*. ACM, 107–116. <https://doi.org/10.1145/2817721.2817735> 42
- [183] Hanae Rateau, Yosra Rekik, Laurent Grisoni, and Joaquim Jorge. 2016. Talaria: Continuous Drag & Drop on a Wall Display. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, 199–204. <https://doi.org/10.1145/2992154.2992164> ix, 17, 18, 54, 63
- [184] Jun Rekimoto. 1997. Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology (UIST '97)*. ACM, 31–39. <https://doi.org/10.1145/263407.263505> 20, 40, 42
- [185] Jun Rekimoto, Brygg Ullmer, and Haruo Oba. 2001. DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, 269–276. <https://doi.org/10.1145/365024.365115> 15
- [186] Simon Robinson, Céline Coutrix, Jennifer Pearson, Juan Rosso, Matheus Fernandes Torquato, Laurence Nigay, and Matt Jones. 2016. Emergeables: Deformable Displays for Continuous Eyes-Free Mobile Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 3793–3805. <https://doi.org/10.1145/2858036.2858097> 76

- [187] Yvonne Rogers and Siân Lindley. 2004. Collaborating around vertical and horizontal large interactive displays: which way is best? *Interacting with Computers* 16, 6 (2004), 1133–1152. <https://doi.org/10.1016/j.intcom.2004.07.008> 14
- [188] Kimiko Ryokai, Elena Durán López, Noura Howell, Jon Gillick, and David Baman. 2018. Capturing, Representing, and Interacting with Laughter. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–12. <https://doi.org/10.1145/3173574.3173932> 5
- [189] Elena Sabinson, Isha Pradhan, and Keith Evan Green. 2021. Plant-Human Embodied Biofeedback (PheB): A Soft Robotic Surface for Emotion Regulation in Confined Physical Space. In *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '21)*. ACM, Article 89, 14 pages. <https://doi.org/10.1145/3430524.3446065> 101
- [190] Alexander Schick, Florian van de Camp, Joris Ijsselmuiden, and Rainer Stiefelhaugen. 2009. Extending Touch: Towards Interaction with Large-Scale Surfaces. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, 117–124. <https://doi.org/10.1145/1731903.1731927> 17
- [191] Dominik Schmidt, Fadi Chehimi, Enrico Rukzio, and Hans Gellersen. 2010. PhoneTouch: A Technique for Direct Phone Interaction on Surfaces. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, 13–16. <https://doi.org/10.1145/1866029.1866034> 21, 73
- [192] Dominik Schmidt, Raf Ramakers, Esben W. Pedersen, Johannes Jasper, Sven Köhler, Aileen Pohl, Hannes Rantzsch, Andreas Rau, Patrick Schmidt, Christoph Sterz, Yanina Yurchenko, and Patrick Baudisch. 2014. Kickables: Tangibles for Feet. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 3143–3152. <https://doi.org/10.1145/2556288.2557016> 101
- [193] Bertrand Schneider, Patrick Jermann, Guillaume Zufferey, and Pierre Dillenbourg. 2011. Benefits of a Tangible Interface for Collaborative Learning and Interaction. *IEEE Trans. Learn. Technol.* 4, 3 (July 2011), 222–232. <https://doi.org/10.1109/TLT.2010.36> 11, 27
- [194] Alexa F. Siu, Eric J. Gonzalez, Shenli Yuan, Jason B. Ginsberg, and Sean Follmer. 2018. ShapeShift: 2D Spatial Manipulation and Self-Actuation of Tabletop Shape Displays for Tangible and Haptic Interaction. In *Proceedings of the 2018*

- CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–13. <https://doi.org/10.1145/3173574.3173865> 76, 84
- [195] Martin Spindler. 2012. Spatially Aware Tangible Display Interaction in a Tabletop Environment. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. ACM, 277–282. <https://doi.org/10.1145/2396636.2396679> 93
- [196] Martin Spindler, Wolfgang Büschel, and Raimund Dachsel. 2012. Use Your Head: Tangible Windows for 3D Information Spaces in a Tabletop Environment. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. ACM, 245–254. <https://doi.org/10.1145/2396636.2396674> 93
- [197] Hemant Bhaskar Surale, Fabrice Matulic, and Daniel Vogel. 2019. Experimental Analysis of Barehand Mid-Air Mode-Switching Techniques in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–14. <https://doi.org/10.1145/3290605.3300426> 18
- [198] Ryo Suzuki, Rubaiat Habib Kazi, Li-yi Wei, Stephen DiVerdi, Wilmot Li, and Daniel Leithinger. 2020. RealitySketch: Embedding Responsive Graphics and Visualizations in AR through Dynamic Sketching. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. ACM, 166–181. <https://doi.org/10.1145/3379337.3415892> 76
- [199] Ryo Suzuki, Clement Zheng, Yasuaki Kakehi, Tom Yeh, Ellen Yi-Luen Do, Mark D. Gross, and Daniel Leithinger. 2019. ShapeBots: Shape-Changing Swarm Robots. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. ACM, 493–505. <https://doi.org/10.1145/3332165.3347911> 84, 93
- [200] Yujie Tao, Shan-Yuan Teng, and Pedro Lopes. 2021. Altering Perceived Softness of Real Rigid Objects by Restricting Fingerpad Deformation. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. ACM, 985–996. <https://doi.org/10.1145/3472749.3474800> 5, 76
- [201] Sureyya Tarkan, Vibha Sazawal, Allison Druin, Evan Golub, Elizabeth M. Bon-signore, Greg Walsh, and Zeina Atrash. 2010. Toque: Designing a Cooking-Based Programming Language for and with Children. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 2417–2426. <https://doi.org/10.1145/1753326.1753692> 5, 6, 73, 86, 87

- [202] Anne Treisman. 1985. Preattentive processing in vision. *Computer vision, graphics, and image processing* 31, 2 (1985), 156–177. [https://doi.org/10.1016/S0734-189X\(85\)80004-9](https://doi.org/10.1016/S0734-189X(85)80004-9) 100
- [203] Theophanis Tsandilas, Anastasia Bezerianos, and Thibaut Jacob. 2015. Sketch-Sliders: Sketching Widgets for Visual Exploration on Wall Displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 3255–3264. <https://doi.org/10.1145/2702123.2702129> 19, 45, 47
- [204] Tiffany Tseng, Coram Bryant, and Paulo Blikstein. 2010. Mechanix: An Interactive Display for Exploring Engineering Design through a Tangible Interface. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, 265–266. <https://doi.org/10.1145/1935701.1935757> ix, 7, 23, 92
- [205] Wen-Jie Tseng, Li-Yang Wang, and Liwei Chan. 2019. FaceWidgets: Exploring Tangible Interaction on Face with Head-Mounted Displays. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. ACM, 417–427. <https://doi.org/10.1145/3332165.3347946> 5
- [206] Philip Tuddenham, David Kirk, and Shahram Izadi. 2010. Graspables Revisited: Multi-Touch vs. Tangible Input for Tabletop Displays in Acquisition and Manipulation Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2223–2232. <https://doi.org/10.1145/1753326.1753662> 12, 20, 24, 33, 44
- [207] Philip Tuddenham, David Kirk, and Shahram Izadi. 2010. Graspables Revisited: Multi-Touch vs. Tangible Input for Tabletop Displays in Acquisition and Manipulation Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 2223–2232. <https://doi.org/10.1145/1753326.1753662> 101
- [208] Brygg Ullmer, Hiroshi Ishii, and Dylan Glas. 1998. MediaBlocks: Physical Containers, Transports, and Controls for Online Media. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, 379–386. <https://doi.org/10.1145/280814.280940> 74
- [209] Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. 2005. Token+constraint Systems for Tangible Interaction with Digital Information. *ACM Trans. Comput.-Hum. Interact.* 12, 1 (March 2005), 81–118. <https://doi.org/10.1145/1057237.1057242> 24

- [210] Nicolas Villar, Daniel Cletheroe, Greg Saul, Christian Holz, Tim Regan, Oscar Salandin, Misha Sra, Hui-Shyong Yeo, William Field, and Haiyan Zhang. 2018. Project Zanzibar: A Portable and Flexible Tangible Interaction Platform. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, 1–13. <https://doi.org/10.1145/3173574.3174089> 92
- [211] Simon Voelker, Christian Cherek, Jan Thar, Thorsten Karrer, Christian Thoresen, Kjell Ivar Øvergård, and Jan Borchers. 2015. PERCs: Persistently Trackable Tangibles on Capacitive Multi-Touch Displays. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and; Technology (UIST '15)*. ACM, 351–356. <https://doi.org/10.1145/2807442.2807466> ix, 7
- [212] Simon Voelker, Kosuke Nakajima, Christian Thoresen, Yuichi Itoh, Kjell Ivar Øvergård, and Jan Borchers. 2013. PUCs: Detecting Transparent, Passive Untouched Capacitive Widgets on Unmodified Multi-Touch Displays. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces (ITS '13)*. ACM, 101–104. <https://doi.org/10.1145/2512349.2512791> 15, 53, 90
- [213] Simon Voelker, Kjell Ivar Øvergård, Chat Wacharamanatham, and Jan Borchers. 2015. Knobology Revisited: A Comparison of User Performance between Tangible and Virtual Rotary Knobs. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces (ITS '15)*. ACM, 35–38. <https://doi.org/10.1145/2817721.2817725> 5, 12, 24, 47, 73
- [214] Simon Voelker, Kjell Ivar undefinedvergård, Chat Wacharamanatham, and Jan Borchers. 2015. Knobology Revisited: A Comparison of User Performance between Tangible and Virtual Rotary Knobs. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 35–38. <https://doi.org/10.1145/2817721.2817725> 32
- [215] Daniel Vogel and Ravin Balakrishnan. 2004. Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, 137–146. <https://doi.org/10.1145/1029632.1029656> ix, x, 13, 18, 19, 49
- [216] Daniel Vogel and Ravin Balakrishnan. 2005. Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, 33–42. <https://doi.org/10.1145/1095034.1095041> 18, 19, 46, 47
- [217] Ulrich von Zadow, Daniel Bösel, Duc Dung Dam, Anke Lehmann, Patrick Reipschläger, and Raimund Dachzelt. 2016. Miners: Communication and Awareness

- in Collaborative Gaming at an Interactive Display Wall. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, 235–240. <https://doi.org/10.1145/2992154.2992174> x, 6, 14, 16, 19, 20, 21, 46, 47, 75, 94
- [218] Ulrich von Zadow, Wolfgang Büschel, Ricardo Langner, and Raimund Dachzelt. 2014. SleeD: Using a Sleeve Display to Interact with Touch-Sensitive Display Walls. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14)*. ACM, 129–138. <https://doi.org/10.1145/2669485.2669507> 4, 16, 47
- [219] Manuela Waldner, Jörg Hauber, Jürgen Zauner, Michael Haller, and Mark Billingham. 2006. Tangible Tiles: Design and Evaluation of a Tangible User Interface in a Collaborative Tabletop Setup. In *Proceedings of the 18th Australia Conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments (OZCHI '06)*. ACM, 151–158. <https://doi.org/10.1145/1228175.1228203> 5, 12, 15, 40, 73
- [220] Torben Wallbaum, Maria Rauschenberger, Janko Timmermann, Wilko Heuten, and Susanne C.J. Boll. 2018. Exploring Social Awareness: A Design Case Study in Minimal Communication. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18)*. ACM, 1–10. <https://doi.org/10.1145/3170427.3174365> 5
- [221] Colin Ware and Marlon Lewis. 1995. The DragMag Image Magnifier. In *Conference Companion on Human Factors in Computing Systems (CHI '95)*. ACM, 407–408. <https://doi.org/10.1145/223355.223749> 42
- [222] Malte Weiss, Florian Schwarz, Simon Jakubowski, and Jan Borchers. 2010. Madgets: Actuating Widgets on Interactive Tabletops. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, 293–302. <https://doi.org/10.1145/1866029.1866075> 15, 100
- [223] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAP Widgets: Bridging the Gap between Virtual and Physical Controls on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 481–490. <https://doi.org/10.1145/1518701.1518779> 20, 24, 47, 53
- [224] Malte Weiss, Julie Wagner, Roger Jennings, Yvonne Jansen, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAPbook: Tangible Widgets on Multi-Touch Tables in Groupware Environments. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 297–300. <https://doi.org/10.1145/1517664.1517725> 11, 40

- [225] Pierre Wellner. 1991. The DigitalDesk Calculator: Tangible Manipulation on a Desk Top Display. In *Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology (UIST '91)*. ACM, 27–33. <https://doi.org/10.1145/120782.120785> 20
- [226] Amy Wibowo, Daisuke Sakamoto, Jun Mitani, and Takeo Igarashi. 2012. DressUp: A 3D Interface for Clothing Design with a Physical Mannequin. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, 99–102. <https://doi.org/10.1145/2148131.2148153> 86, 87
- [227] Alexander Wilberz, Dominik Leschtschow, Christina Trepkowski, Jens Maiero, Ernst Kruijff, and Bernhard Riecke. 2020. FaceHaptics: Robot Arm Based Versatile Facial Haptics for Immersive Environments. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, 1–14. <https://doi.org/10.1145/3313831.3376481> ix, 4, 5
- [228] Andrew D. Wilson and Hrvoje Benko. 2010. Combining Multiple Depth Cameras and Projectors for Interactions on, above and between Surfaces. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. ACM, 273–282. <https://doi.org/10.1145/1866029.1866073> 17
- [229] Andy Wu, Sam Mendenhall, Jayraj Jog, Loring Scotty Hoag, and Ali Mazalek. 2012. A Nested APi Structure to Simplify Cross-Device Communication. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, 225–232. <https://doi.org/10.1145/2148131.2148180> 73, 94
- [230] Andy Wu, Derek Reilly, Anthony Tang, and Ali Mazalek. 2010. Tangible Navigation and Object Manipulation in Virtual Environments. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, 37–44. <https://doi.org/10.1145/1935701.1935710> 91, 92
- [231] Anna Xambó, Brigid Drozda, Anna Weisling, Brian Magerko, Marc Huet, Travis Gasque, and Jason Freeman. 2017. Experience and Ownership with a Tangible Computational Music Installation for Informal Learning. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)*. ACM, 351–360. <https://doi.org/10.1145/3024969.3024988> 5
- [232] Nesra Yannier, Kenneth R. Koedinger, and Scott E. Hudson. 2015. Learning from Mixed-Reality Games: Is Shaking a Tablet as Effective as Physical Observation?.

In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 1045–1054. <https://doi.org/10.1145/2702123.2702397> 75

- [233] Neng-Hao Yu, Li-Wei Chan, Seng Yong Lau, Sung-Sheng Tsai, I-Chun Hsiao, Dian-Je Tsai, Fang-I Hsiao, Lung-Pan Cheng, Mike Chen, Polly Huang, and Yi-Ping Hung. 2011. TUIC: Enabling Tangible Interaction on Capacitive Multi-Touch Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 2995–3004. <https://doi.org/10.1145/1978942.1979386> 90
- [234] Neng-Hao Yu, Sung-Sheng Tsai, I-Chun Hsiao, Dian-Je Tsai, Meng-Han Lee, Mike Y. Chen, and Yi-Ping Hung. 2011. Clip-on Gadgets: Expanding Multi-Touch Interaction Area with Unpowered Tactile Controls. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 367–372. <https://doi.org/10.1145/2047196.2047243> 5, 73
- [235] Jian Zhao, R. William Soukoreff, and Ravin Balakrishnan. 2015. Exploring and Modeling Unimanual Object Manipulation on Multi-Touch Displays. *Int. J. Hum.-Comput. Stud.* 78, C (June 2015), 68–80. <https://doi.org/10.1016/j.ijhcs.2015.02.011> 36
- [236] Jamie Zigelbaum, Adam Kumpf, Alejandro Vazquez, and Hiroshi Ishii. 2008. Slurp: Tangibility Spatiality and an Eyedropper. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems (CHI EA '08)*. ACM, 2565–2574. <https://doi.org/10.1145/1358628.1358713> 73, 92, 93
- [237] Simone Zimmermann, Sonja Rümelin, and Andreas Butz. 2014. I Feel It in My Fingers: Haptic Guidance on Touch Surfaces. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction (TEI '14)*. ACM, 9–12. <https://doi.org/10.1145/2540930.2540938> 5
- [238] Guillaume Zufferey, Patrick Jermann, Aurélien Lucchi, and Pierre Dillenbourg. 2009. TinkerSheets: Using Paper Forms to Control and Visualize Tangible Simulations. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 377–384. <https://doi.org/10.1145/1517664.1517740> 6, 73

Titre: Interaction Tangible pour les Murs d'Écrans

Mots clés: Interaction Humain Machine, Interaction Tangible, Murs d'Écrans

Résumé:

Les murs d'écrans plongent les utilisateurs dans de larges espaces d'informations ultra-haute résolution. Ils sont bien adaptés à l'analyse de grands ensembles de données car les utilisateurs peuvent se déplacer physiquement pour explorer ce qui est affiché à l'écran. Ils facilitent également la collaboration car leur taille permet facilement d'accueillir plusieurs utilisateurs à la fois. Cependant, créer des interactions efficaces avec les murs d'écrans est un défi. Les périphériques traditionnels tel que le clavier et la souris montre vite leur limite dans un contexte où plusieurs utilisateurs interagissent et se déplacent librement.

La littérature en IHM propose des moyens d'interaction alternatifs. En particulier, l'interaction tangible, qui s'appuie sur la manipulation d'objets physiques pour interagir avec des scènes virtuelles, offre des avantages intéressants pour les murs d'écrans. La matérialité des contrôleurs offre un retour haptique qui permet de les manipuler sans les regarder. Leur forme suggère aussi la manière de les attraper et de les manipuler, guidant les utilisateurs sur leurs fonctionnalités. De nombreuses études empiriques comparant les contrôleurs tangibles à d'autres formes d'interaction montrent qu'ils procurent des gains significatifs en termes de vitesse et de précision de manipulation.

Cependant, fabriquer des contrôleurs tangibles pour les murs d'écrans est difficile. Tout d'abord, l'orientation de l'écran et sa taille doivent être prise en compte pour fabriquer des tangibles appropriés. De plus, les utilisateurs sont mobiles : ils s'éloignent pour obtenir un angle de vue plus large, se rapprochent pour voir plus de détails, ou ajustent leur position en fonction de celle des autres utilisateurs. Les contrôleurs tangibles doivent donc être conçus pour être utilisables quelle que soit la position de l'utilisateur dans la pièce. Enfin, un mur d'écrans est souvent situé dans un contexte rassemblant d'autres dispositifs (tables interactives, ordinateurs, etc.). Dans ce cas, il est nécessaire de prendre en compte l'ensemble du

contexte, contraignant la forme des tangibles et les technologies sous-jacentes.

Mon travail de thèse propose trois contributions pour faciliter l'interaction tangible avec les murs d'écrans.

Mon premier projet, *WallTokens*, propose des tangibles qui permettent d'interagir sur la surface des murs d'écrans. Les *WallTokens* sont équipés d'un mécanisme qui permet aux utilisateurs de les attacher et de les détacher facilement de la surface du mur. Cela permet de les laisser en place lorsque les utilisateurs veulent libérer leur main pour d'autres tâches. Nous présentons deux études évaluant la facilité d'utilisation et l'efficacité des *WallTokens*. Nos résultats montrent qu'ils sont plus précis et plus confortables que les interactions tactiles pour effectuer des manipulations de bas niveau sur mur d'écrans.

Mon deuxième projet, *SurfAirs*, propose des tangibles permettant des interactions avec les murs d'écrans en surface, quand les utilisateurs ont besoin de détails et de précision, mais aussi à distance quand ils ont besoin d'un grand angle de vue. Les *SurfAirs* permettent également une transition continue entre ces deux modes d'interaction. Nous présentons deux études qui comparent les *SurfAirs* avec des gestes à main nue pour effectuer des tâches de manipulation de bas niveau. Les *SurfAirs* sont plus performants que les gestes à main nue en termes de précision et de vitesse et les utilisateurs les préfèrent. Le troisième projet propose une étude de la littérature sur l'utilisation de contrôleur tangible avec des écrans. Chaque article étudié est classifié selon 12 dimensions qui reflètent les aspects de la conception du contrôleur et de l'écran. Nous proposons un outil Web qui permet l'exploration de notre corpus d'articles à travers ces dimensions de classification. Nous discutons ensuite les défis qui sous-tendent la conception de contrôleurs tangibles dans un environnement multi-écrans.

Title: Tangible Interaction for Wall Displays

Keywords: Human-Computer Interaction, Tangible Interaction, Wall-Displays

Summary:

Wall displays immerse users in large, high-resolution information spaces. They are well suited for data analysis, as users only need to move around the physical space to explore the virtual information space displayed on the wall. They also facilitate collaboration as their large physical size can accommodate multiple users. However, designing effective ways of interacting with wall displays is challenging. Traditional input devices, such as mice and keyboards, quickly show their limitations in an environment where multiple users can interact and move freely.

HCI literature offers interesting alternatives to traditional input techniques. In particular, Tangible User Interactions (TUIs), where users rely on custom tangible objects to interact with the virtual scene, have proved efficient with different types of displays ranging from smartphones to tabletops. Tangible controllers have natural advantages such as the haptic feedback they provide that enables eyes-free manipulations. They also afford specific grasps and manipulations, guiding users on what they can do with them. Empirical studies that compare tangibles to other forms of input also report quantitative gains in regarding manipulation speed and precision in different hardware setups.

However, designing tangible controllers for wall displays is difficult. First, the large size and vertical orientation of walls must be taken into account to design tangibles with a suitable form factor. Second, users move in space. They move away to get a wider view, move closer to see details, or adjust their physical position based on other users and objects in the room. This means that tangible controllers must be usable regardless of the user's position in the room, which has some impact on design and engineering aspects. Finally, a wall display is often located in an environment that feature other devices and displays. In such cases, designing tangible controllers for a wall display requires to consider the whole multi-display environ-

ment, which constrains even more the tangibles' form factor and the underlying technologies.

My thesis work makes three contributions towards enabling tangible interaction with wall displays.

The first project, WallTokens, contributes tangibles for enabling *on surface* interaction with wall displays. WallTokens are low-cost, passive controllers that users can manipulate directly on the wall's surface. WallTokens have a mechanism that allows users to easily attach and detach them from the wall surface, so that when users are done interacting, they can leave them in place and free their hands for other purposes. We report on two studies assessing WallTokens' usability, showing that they are more precise and comfortable than bare-hand gestures to perform low-level manipulations on walls.

The second project, SurfAirs, contributes tangibles that support not only *on surface* interaction but also *distant* interaction with wall displays. We present two possible designs for versatile tangible controllers that can be used both on the wall surface when users need precision and detail, and in the air when they need a wide viewing angle. SurfAirs support both types of input, as well as smooth transitions between the two. We report on two studies that compare SurfAir prototypes with bare hand gestures for performing low-level manipulation tasks. SurfAirs outperform bare hand gestures regarding accuracy, speed and user preference.

The third project contributes a survey about the use of physical controllers to interact with a physical display. Each project is described along twelve dimensions that capture the design aspects of the controller, the properties of the display and how they communicate with each other. We contribute a Web page to explore this list of references along the different dimensions, and use it to discuss the challenges that underlie the design of tangible controllers in a multi-display environment.

