

# AirConstellations: In-Air Device Formations for Cross-Device Interaction via Multiple Spatially-Aware Armatures

Nicolai Marquardt\*  
Microsoft Research, UCL

Nathalie Henry Riche  
Microsoft Research

Christian Holz  
ETH Zürich

Hugo Romat  
ETH Zürich, Microsoft

Michel Pahud  
Microsoft Research

Frederik Brudy  
Autodesk Research

David Ledo  
Autodesk Research

Chunjong Park  
University of Washington

Molly Jane Nicholas  
University of California, Berkeley

Teddy Seyed  
Microsoft Research

Eyal Ofek  
Microsoft Research

Bongshin Lee  
Microsoft Research

William A. S. Buxton  
Microsoft Research

Ken Hinckley†  
Microsoft Research

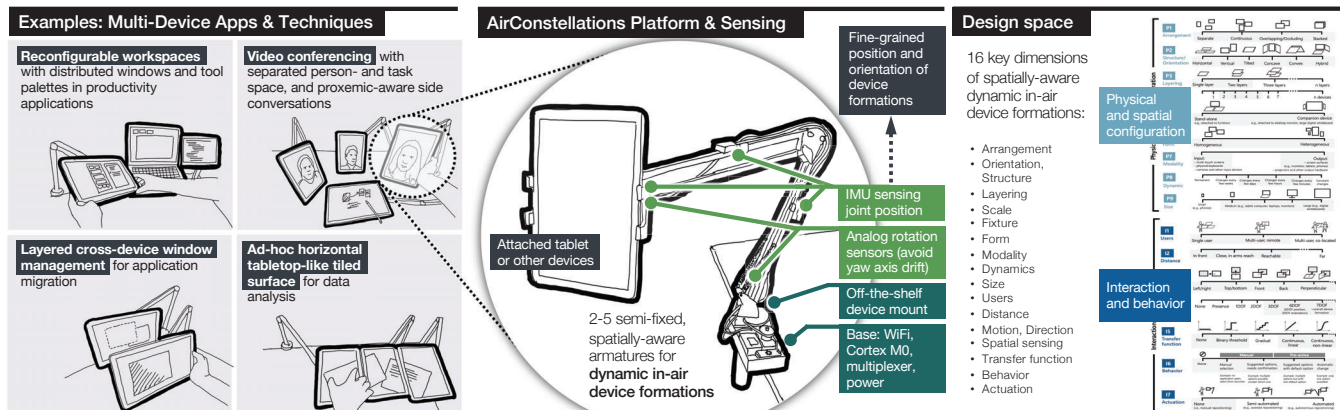


Figure 1: AirConstellations enables flexible cross-device interactions where users can move, re-orient, and pose devices together to form *dynamic in-air device formations*.

## ABSTRACT

AirConstellations supports a unique *semi-fixed* style of cross-device interactions via multiple self-spatially-aware armatures to which users can easily attach (or detach) tablets and other devices. In particular, AirConstellations affords highly flexible and *dynamic device formations* where the users can bring multiple devices together in-air – with 2-5 armatures poseable in 7DoF within the

same workspace – to suit the demands of their current task, social situation, app scenario, or mobility needs. This affords an interaction metaphor where relative orientation, proximity, attaching (or detaching) devices, and continuous movement into and out of ad-hoc ensembles can drive context-sensitive interactions. Yet all devices remain self-stable in useful configurations even when released in mid-air.

We explore flexible physical arrangement, feedforward of transition options, and layering of devices in-air across a variety of multi-device app scenarios. These include video conferencing with flexible arrangement of the person-space of multiple remote participants around a shared task-space, layered and tiled device formations with overview+detail and shared-to-personal transitions, and flexible composition of UI panels and tool palettes across devices for productivity applications. A preliminary interview study highlights user reactions to AirConstellations, such as for minimally disruptive device formations, easier physical transitions, and balancing “*seeing and being seen*” in remote work.

\*nicolai.marquardt@acm.org

†kenh@microsoft.com, surface-fleet@microsoft.com

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

UIST '21, October 10–14, 2021, Virtual Event, USA

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-8635-7/21/10...\$15.00

<https://doi.org/10.1145/3472749.3474820>

## CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI); Interaction devices.**

## KEYWORDS

cross-device computing; device formations; semi-fixed workspaces; multi-device; interaction techniques; platform; cross-device tracking; SurfaceFleet; device ecologies

### ACM Reference Format:

Nicolai Marquardt, Nathalie Henry Riche, Christian Holz, Hugo Romat, Michel Pahud, Frederik Brudy, David Ledo, Chunjong Park, Molly Jane Nicholas, Teddy Seyed, Eyal Ofek, Bongshin Lee, William A. S. Buxton, and Ken Hinckley. 2021. AirConstellations: In-Air Device Formations for Cross-Device Interaction via Multiple Spatially-Aware Armatures. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), October 10–14, 2021, Virtual Event, USA*. ACM, New York, NY, USA, 17 pages. <https://doi.org/10.1145/3472749.3474820>

## 1 INTRODUCTION

People increasingly own and use multiple devices in their personal workspaces, a long-term trend (e.g., [81]) accelerated by remote work during the COVID-19 pandemic [90]. In the course of a single day, users may go from a teleconference to data analysis to editing spreadsheets and back again — with unpredictable intrusions of home life (e.g., partners, children, pets, deliveries) driving task shifts or spur-of-the-moment reconfigurations. Within these personal device ecologies, cross-device computing leverages the complementary roles and micro-mobility [62] afforded by individual devices to support a wide range of application scenarios, use cases, and communication [8].

In these emerging cross-device ecologies, most existing work either uses *fixed* setups with combinations of interactive surfaces embedded in the environment for individual and small-group use (e.g., [7, 39, 102, 103]), or *mobile* cross-device setups that link devices such as tablets, phones, or other digital surfaces into ad-hoc, lightweight federations (e.g., [13, 32, 60]). Physical reconfigurations to better adapt to changes in task, users, or other context [56, 78, 90] remain difficult in fixed spaces — and while easier in mobile setups, there are often only limited options to spatially arrange devices around a workspace, so they often remain either hand-held or placed flat on a desk surface. Furthermore, for most mobile setups, sensing where devices are in space (which drives many cross-device interaction techniques [8]) remains challenging (e.g., [29, 74]).

In between these two classes of *fixed vs. mobile* multi-device use, we can position a third category of *semi-fixed* cross-device setups, akin to Hall’s semi-fixed architectural features [31]. The goal is to preserve the modularity and rapid reconfigurability of mobile setups, while at the same time allowing physical rearrangement of diverse configurations as the task and situation demand, facilitating people’s rich practices of spatially organizing information [46]. There have been several investigations into more flexible device configurations, such as through actuated furniture [2, 27, 28, 58] or re-connectable devices [64, 73, 87, 88]. Yet, preserving high degrees of modular reconfigurability, affording fluid and lightweight changes of physical workspace device configurations, remains a challenge. This becomes even more difficult for interactions on a

*desktop*-scale, with fast-changing working practices [90] and due to limited screen real-estate in the presence of multi-tasking, with varied applications side-by-side with one another.

AirConstellations contributes a new technical approach, platform, and techniques for in-air, poseable multi-device assemblages — devices not only flat on a table, but floating in front, at varying angles, or huddled together (Figure 1). AirConstellations workspaces are composed of 2–5 flexible armatures with attached devices (such as tablets), allowing easy physical movement and positioning of devices around a workspace. In particular, we retrofit multiple off-the-shelf device mounts with (low-cost and robust) sensors to determine the spatial location and orientation of all devices, which we then combine into a 7DoF model of device assemblages around the workspace. This sensor augmentation opens up new possibilities for sensor-driven application behaviors. To inform the design of this new category of mid-air, spatially-aware cross-device setups, we synthesized a design space (cf. Figure 3) — along 16 dimensions of *hardware configuration* (arrangement, structure/orientation, layering, scale, fixture, form, modality, dynamics, and size) and *interaction* (users, distance, motion/direction, spatial sensing, transfer function, behavior, and actuation).

We then contribute a set of applications and interaction behaviors, illustrating techniques across different segments of this device formation design space. In particular, we use the sensed information of 3D position and orientation of all connected devices to drive applications (with a low transaction cost) responding to any changes of orientation or proximity between devices. Our applications include a video conferencing app facilitating fluid separation of task- and person-space [10], the use of layered and tiled device formations with techniques such as overview+detail and shared-to-personal transitions, and flexible workspaces and tool palettes in productivity applications. We also illustrate the use of ad-hoc operations for managing windows and files across devices, and explore feedforward and feedback to continuously reveal interaction possibilities, so that the user can discover, select between different options, and act on (or choose to ignore) AirConstellations’ spatially-aware behaviors. A preliminary (remote) interview study with eight participants suggests that AirConstellations could afford rapid transitions or adjustment of people’s workspaces while minimizing disruption, allow balancing of “seeing and being seen” during remote collaborations, and afford meaningful physical proximity-based techniques that facilitate sharing.

In summary, we contribute:

- *AirConstellations system*: Semi-fixed cross-device interaction with *dynamic in-air device formations* via multiple self-stable, spatially-aware armatures;
- *Design space* that articulates 16 key dimensions of (*physical & spatial*) *hardware configuration* and *interaction* for semi-fixed cross-device setups;
- *Application scenarios & interaction techniques* for the dynamic device formations afforded by AirConstellations, including layering, tiling, adjacency, and feedforward animations that surface capabilities to users; and
- *Initial user insights (from a remote interview study)* for the highly reconfigurable workspaces, applications, and techniques thus afforded.

Our work aims to show how multiple AirConstellations devices used together in the same operating volume can afford some aspects from mobile interaction (and micro-mobility concepts from social theory [62]) while simultaneously offering some properties of semi-fixed device arrangements in a way that is flexible and delightful, allowing the effort of forming a particular device arrangement to repay itself over time because it stays put in mid-air once released. Such devices, in combination with appropriate applications, feedback, interaction techniques, and distributed systems foundations such as SurfaceFleet [9], may open up an intriguing new design space of highly dynamic and easily reconfigurable personal workspaces with dynamic in-air device formations.

## 2 BACKGROUND

We synthesize work in cross-device computing, in particular for fixed, instrumented environments and mobiles, ad-hoc collaborative interactions, challenges of spatial tracking, and work towards reconfigurable workspaces.

### 2.1 Multi-Surface Workspaces

Large interactive surfaces can form parts of effective individual or collaborative workspaces (often inspired by Wellner’s seminal *DigitalDesk* [102]). Varying physical form factors and orientation can afford collaborative work from horizontal [17, 20], tilted [16, 38, 69], to vertical [103] surfaces. Non-planar, bent interaction surfaces [101, 105] can facilitate reach and visibility of digital information.

To better support collaborative work, *multiple* interactive surfaces can be combined in interactive environments [87], such as for collaborative astrophysics explorations [103], work in design studios [21], facilitating video-mediated communication [51], or data analytics [15, 85]. Most combine horizontal and vertical displays, some use projectors [104], or other forms of output (cf. taxonomies [6, 8]). Following Hall’s terminology [31], these interactive environments could be categorized as *fixed features* (Figure 2), parts of a space that are not or only rarely reconfigured or changed. To make interactive spaces adapt to changes in context, some follow a hybrid approach [3, 4, 37, 40], for example, by including mobile devices and laptops in a “*walk up and share environment*” [103].

### 2.2 Ad-hoc Interactions Across Mobile Devices

In contrast to *fixed* interactive spaces, *mobile* setups are at the opposite end of the mobility spectrum (Figure 2) – built to afford high degrees of flexibility and dynamic configurations for different contexts. Some of these setups allow users to combine mobile devices (e.g., phones, tablets, e-readers, smartwatches) in unified interactions spanning across multiple devices [9, 61]. Such setups can then support a wide range of user activities & tasks, including active reading [13, 34], sharing personal media [60], sensemaking [106], and ad-hoc data exploration [52].

Interaction techniques (e.g., pen/touch input, hand gestures, or device motion) can facilitate exploration, transfer, and other content engagement across devices (cf. [8, Table 3]). Rekimoto’s seminal work [76] introduced pick-and-drop for transfer. Many other techniques followed using, for example, stitching [36], 3D hand motion [75], tapping and dragging [32], tilting and portals [66], redirecting content [108], eye-tracking [99], or conduit gestures [12]

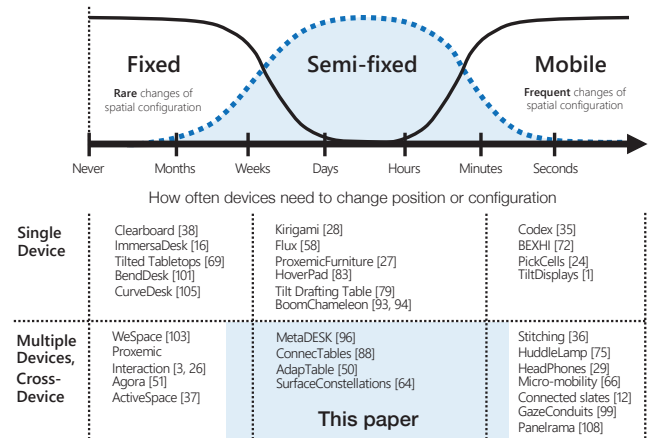


Figure 2: *Semi-fixed* cross-device configurations positioned between fixed and mobile setups.

for fluid data transfer in cross-device ecologies. Related techniques include spatial information lenses [19] and merged displays [60, 82].

To build techniques spanning across devices requires flexible software infrastructure. Therefore, previous work explored approaches for developing dynamic cross-device applications, for example, by introducing shareable dynamic media [47], using scripting for cross-device behaviors [14], generating semi-automatic adaptations based on capabilities [70], and debugging cross-device applications [71].

### 2.3 Reconfigurability

Reconfiguration of a workspace or device setup can be beneficial for various use cases: better supporting a particular social interaction, adapting small workspaces or spaces with multiple uses [90], or changing a setup for a different activity. We apply Hall’s [31] term of *semi-fixed* features to describe such reconfigurable device formations. One approach for such reconfigurability is the integration of actuated, shape-changing mechanisms – e.g., surfaces changing from horizontal to vertical [27], drafting-table orientation [58], or shape changes of the surface area depending on tasks [27, 28]. On a smaller scale, HoverPad [83] uses an actuated screen above a tabletop, and Living Desktop [2] introduces mechanically actuated monitors and keyboards that reconfigure depending on a task or social context.

In some cases, the boundaries between technology and interior architecture blur, when devices or displays become furniture-like modular blocks. For example, AdapTable [50] uses smaller tabletop-blocks, UbiWall [45] allows modular display cubes to be reassembled when needed, and Foxels [73] integrates various input and output capabilities in furniture cubes. Similarly, i-LAND [87] explores such device-furniture hybrids, for example, with ConnecTables [88] where display spaces merge when moving tables in direct proximity. u-Texture [48] introduced grid-like structures and links that allow surfaces to be assembled in different horizontal and vertical arrangements, and SurfaceConstellations [64] uses 3D-printed brackets to form multi-device workspaces. Shape-changing [1, 92] and modular [24] mobile devices allow reconfigurability at the smallest scale.

## 2.4 Sensing Location and Orientation

For most cross-device setups, one particular challenge is to infer the position and orientation of the devices to enable any spatially aware cross-device interaction techniques (e.g., easily re-directing input to devices, directional gestures transferring content). This is why cross-device research [8] investigates strategies to sense inter-device proximity and orientation. Some use computer vision of RGB [29], depth cameras [75], marker-based motion capture [65, 107], marker recognition [59], polarization filters [74], or eye tracking [99]. Other techniques apply short-range infrared sensing [68], radio-based sensing such as bluetooth [22, 42], or near-field communication (NFC, RFID) [77]. Hybrid approaches are promising, for example, fusing radio- and camera-based sensing [66], or bluetooth-radio and acoustic stereo signal positioning [41]. In addition, aside from automated tracking, synchronous gestures [33] and stitching [36] are effective techniques for quickly forming device federations. Reliable and robust sensing, however, remains a “*major challenge*” [8]: vision-based sensing achieves relatively high spatial resolution but faces difficulties with occlusion, lighting conditions, and potentially undesirable always-on camera tracking; and radio-based sensing struggles with achieving high spatial resolution or accuracy. Furthermore, many of these setups require outside-in-tracking [8], which can limit their use and deployability.

## 2.5 Using Spatial Changes as Sensor Input

We can leverage information about any changes of the spatial configuration of a device (or devices) to function as input, affecting application behavior. For example, *Codex* [35] (and similar approaches that followed [80]) senses the opening angle of the connecting hinge to inform context switching and other adaptations of dual-surface applications. *BEXHI* [72], as one example of *deformable* interfaces, uses stretching and bending as direct input. For stationary setups, sensing input of a display’s tilting angle can drive context-appropriate transitions [79], and tracked vertical up and down movements of a tablet can facilitate 3D drawing [53].

Beyond the sensing of a single degree-of-freedom movement (e.g., up/down, tilting angle), tracking the full position (3DoF for x/y/z position) and orientation (additional 3DoF of yaw/pitch/roll orientation) of a device in space allows for the use for rich 3D spatial interaction. Most notable is activeLENS [96], a spatially tracked arm-mounted display, functioning as a window into the hybrid digital-tangible geospace of metaDESK (sensed through a 6DOF magnetic tracker). BoomChameleon [21, 93, 94] introduced a similar setup of a display mounted on a tracked mechanical boom, for rich 3D viewpoint control and spatial annotations.

We apply this particular combination of *spatial awareness* and *high degrees of reconfigurability* of spatially-aware, arm-mounted displays to *multi-device interactions*. As summarized in Figure 2, our work is positioned in the space between fixed and mobile device setups, where we contribute a new approach and platform for dynamic in-air device formations. Through this modularity and high degree of reconfigurability, such semi-fixed setups have the potential to effectively support changing working practices, such as the need for increased work at home, with possibly smaller or shared workspaces [90], or demands for more flexible device and furniture configurations in offices for collaboration [56].

## 3 DESIGN SPACE

To better understand the parameters for creating dynamic in-air device formations, we synthesized a design space along dimensions that represent varying interaction techniques, sensing fidelity, physical characteristics, and spatial arrangements (Figure 3). We incorporated other classifications, in particular the dimensions of flexible dual-device configurations [35, Fig. 2], articulated orientation angles of tilted drawing boards [79, Fig. 3], mappings of the field of cross-device interaction [8, Fig. 2 and Table 3], modular assemblies of multi-device workspaces [64, Fig. 3], and the disparate but related dimensions of graspable user interfaces [20, Table 1]. We identified the following key dimensions relevant for the design of spatially-aware dynamic in-air device formations:

- P1 ARRANGEMENT** — The spatial configuration of devices: separated [3, 103], continuous surface [75, 82] (e.g., edge to edge), overlapping [23, 37], and/or occluding.
- P2 STRUCTURE/ORIENTATION** — The physical orientation of devices in space: horizontal [17, 18, 88], vertical [64], tilted [79], concave [101, 105] vs. convex or both [35, 44], or hybrid [28, 64].
- P3 LAYERING** — Spatial layering of devices around a person’s workspace, from *one* to *n* layers [23].
- P4 SCALE** — The number of devices combined in one cross-device formation.
- P5 FIXTURE** — Are devices configured as a stand-alone [64] setup, for example, attached to a desk, or do they function as a companion [87] for another device (e.g., smaller devices attached to a large interactive whiteboard).
- P6 FORM** — From heterogeneous setups (using similar devices) [75, 108] to heterogeneous configurations (devices of varying size or different capabilities) [3, 37, 103].
- P7 MODALITY** — Which input or output devices form part of the multi-device setup. Input modalities can be multi-touch (e.g., many of our examples use multi-touch tablets), physical keyboards, or cameras. Output modalities can be screens, projectors, or other hardware.
- P8 DYNAMICS** — How often is the spatial configuration of devices changed: from permanent setups, to changes every few weeks/days/hours/minutes, to continuous changes (cf. Figure 2).
- P9 SIZE** — The size of devices included in the formation: small (e.g., phone), medium (e.g., tablet, laptop, monitor), large (e.g., digital whiteboard).
- I1 USERS** — Workspace for single user interactions, multi-user remote collaboration, or multi-user co-located interactions.
- I2 DISTANCE** — How close are the devices physically positioned in relationship to the user: directly in front, in arms reach, further away but reachable, or far.
- I3 MOTION/DIRECTION** — Different directions of approach when moving one device closer to another: from *left* or *right* side; *top* or *bottom* of the device; approaching from the *front* or *back*; or *perpendicular*.
- I4 SPATIAL SENSING** — The fidelity of spatial sensing available: from no sensing to sensing of presence, 1DoF [58, 69, 79], 2DoF [57], 3DoF, 6DoF [83, 93], and 7+DoF (including overall formation).

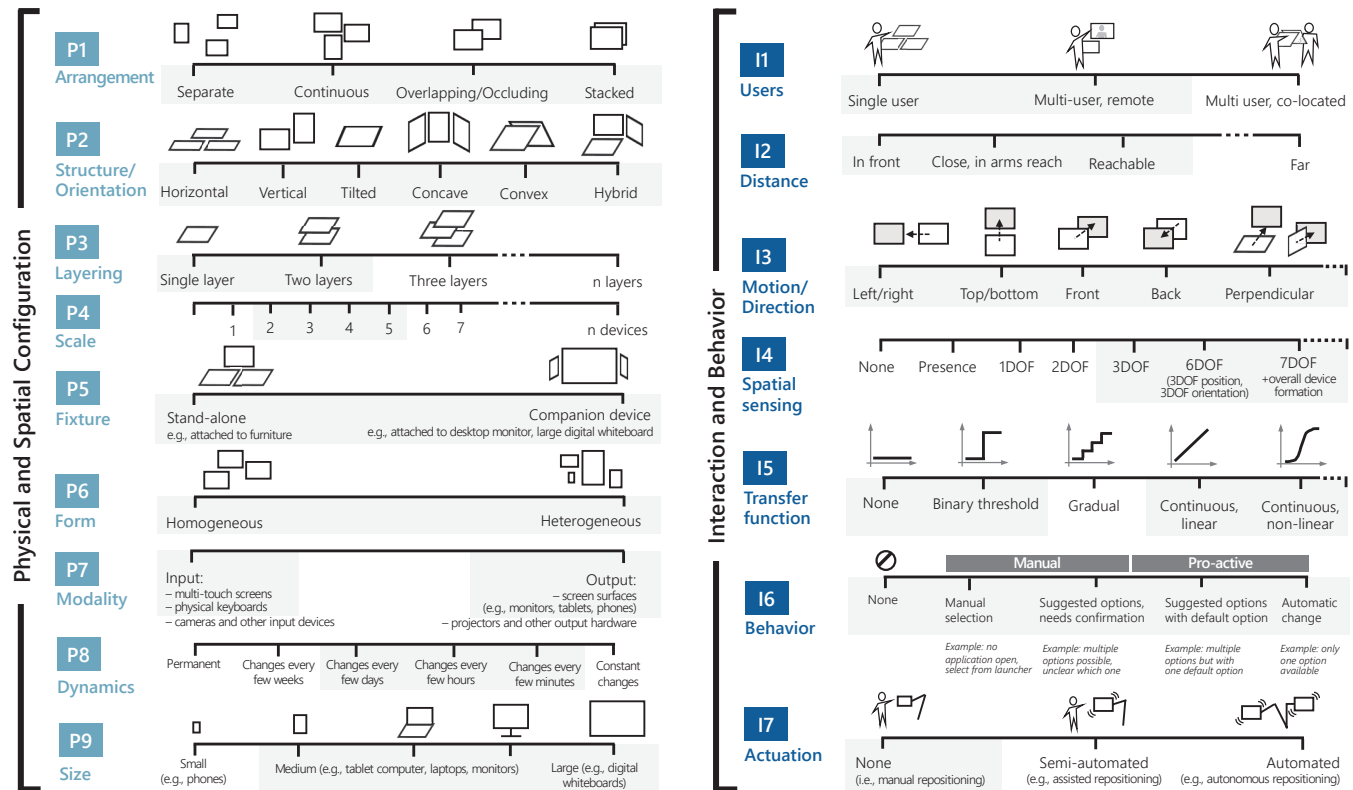


Figure 3: Design space of dynamic in-air device formations: (left) physical and spatial configuration and (right) interaction and behavior (gray background indicates design dimensions we illustrate through examples in this paper).

**I5 TRANSFER FUNCTION** — How is the sensor input mapped to any interactive application behavior: no mapping, binary threshold (e.g., crossing a distance threshold between two devices triggers action), gradual [63], linear continuous [3], or non-linear continuous [79].

**I6 BEHAVIOR** — The way applications implement different action behaviors in response to sensed input: *manual selection* (e.g., if there is no application opened), *seeing suggested options that require confirmation* (e.g., when there is no clear preference in the system of a default choice), *showing options but pro-actively triggering the default option*, and *pro-actively automatically triggering an action* (for example when only one option is available).

**I7 ACTUATION** — Is any actuation mechanism used to reconfigure the position of devices in a formation. In the scope of this paper, we illustrate examples where users manually reconfigure formation. Other approaches are feasible, such as semi-automated (e.g., assisted repositioning) or automated (autonomous position changes using robotic arms) formations (similar to approaches in [2, 27, 28, 58, 83, 86]).

## 4 AIRCONSTELLATIONS PLATFORM

To enable easy reconfiguration of device setups, we designed a hardware platform that consists of off-the-shelf mount armatures (to hold tablets, phones, keyboards, and any other devices) that

are retrofitted with sensors to continuously track the location and orientation of each attached device. We then used the tracking data from these sensors to drive a number of novel interaction techniques (introduced shortly). In this section, we explain the technical implementation of our platform enabling dynamic in-air device formations for cross-device computing.

### 4.1 Concept: Self-stable Armatures for In-Air Device Formations

For the flexible positioning of devices around a user, we needed a mechanical hardware construction that physically holds devices in place. To achieve this we use an armature design of a flexible articulated arm — similar to activeLENS [96] and BoomChamelon [93, 94] — but extend this concept to *multi-device* use and composition of device formations. In particular, we optimized our platform for easy reconfigurability in *small-scale, multi-device setups* — multiple displays and other devices that can be freely positioned in the same working volume. Due to size constraints of using this in desk-sized workspaces, we decided not to use a counter-balanced mechanical boom (which are often used for holding and positioning heavier cameras in movie productions), but instead use a 3-joint flexible tablet mounting arm [95] as the basis of our design. The devices we use in our setups (tablets, keyboards) are lighter and don't require a fully counter-balanced mechanical construction.

## 4.2 Adding Spatial Awareness: Retrofitting Armatures with Sensors

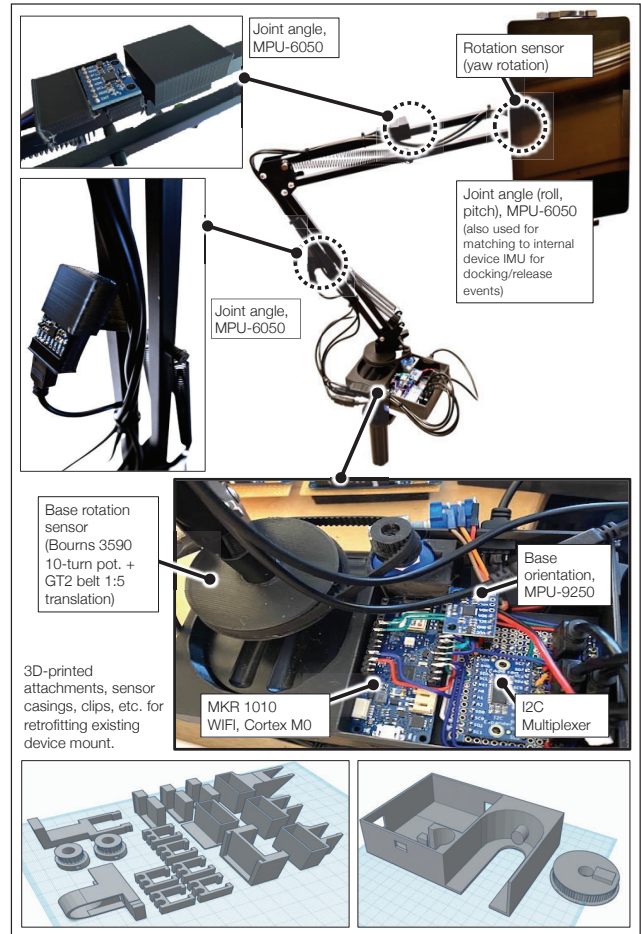
We retrofit these off-the-shelf armatures with sensors to continuously track the position and orientation of each device. The technical setup of a single AirConstellations arm is illustrated in Figure 4.

In each AirConstellations arm we use an *Arduino MKR 1010 Wifi* (SAMD21 Cortex-M0+) microcontroller, with a 1-to-8 I2C multiplexer (TCA9548A), connecting to four IMUs (Inertial Measurement Units), all either MPU6050 or MPU9250. We use one IMU for each of the two main joints of the arm, one for the base (so we can sense the placement of the base) and the other for the joint that connects to the tablet at the end of the mount (Figure 4). To eliminate yaw-axis drift (a challenge with most IMU-based sensing), we do not use any yaw-axis measurement from the IMUs but added mechanical rotation sensors. We use high-precision 10-turn rotation sensors (Bourns 3510) with a connected 1:5 gear translation in the base, and an additional rotation sensor connected to the top joint where the device gets connected. Our algorithm then combines the *yaw-axis* rotation measures (from the base and head rotation axis) with the pitch-axis measures provided by the IMUs for determining the 3DoF position of the device in space, together with 3DoF orientation of the device around the head of the mounting arm. Using off-the-shelf armature mounts and low-cost sensing hardware makes this approach scalable and easy to replicate. We release the STL designs for the 3D-printed sensor enclosures, attachments, and other fixtures (shown at the bottom of Figure 4).<sup>1</sup>

We built five AirConstellations hardware prototypes, each with one device attached (using tablet devices such as the Microsoft *Surface Go*, *Surface Pro*, and *Surface Book*, all connected through local WiFi, and Bluetooth peripherals such as keyboards). The armatures can be attached to any surface, such as a desk or a small table. On first use, the position of the location where each of the armatures is attached needs to be configured in our software (i.e., the relative location and distance from one base to the next). After this one-time configuration, all AirConstellations armatures continuously track the position and orientation of the attached device when moved, and send their position information via UDP packets to a PC merging all the position information and running our prototyping software (Figure 5).

## 4.3 From Mobile to Semi-fixed

We integrated a simple detection for when devices are *added* or *removed* from a formation. For example, users can take their mobile devices and join them to a semi-fixed AirConstellations formation, like docking their phone or tablet into one of the armatures. The device is recognized by the armature mount, where we compare the sensed orientation of the mount to the orientation sensor of the device (i.e., the internal IMU) to recognize docking or removing, and the mobile device then becomes part of or is removed from the current setup. Overall this orientation matching approach works well, but could lead to false recognition due to identical device/mount orientation — other techniques could be used instead, such as an RFID-based detection of devices attached to the armatures (e.g., RFID reader in the armature, tags attached to the back of devices).



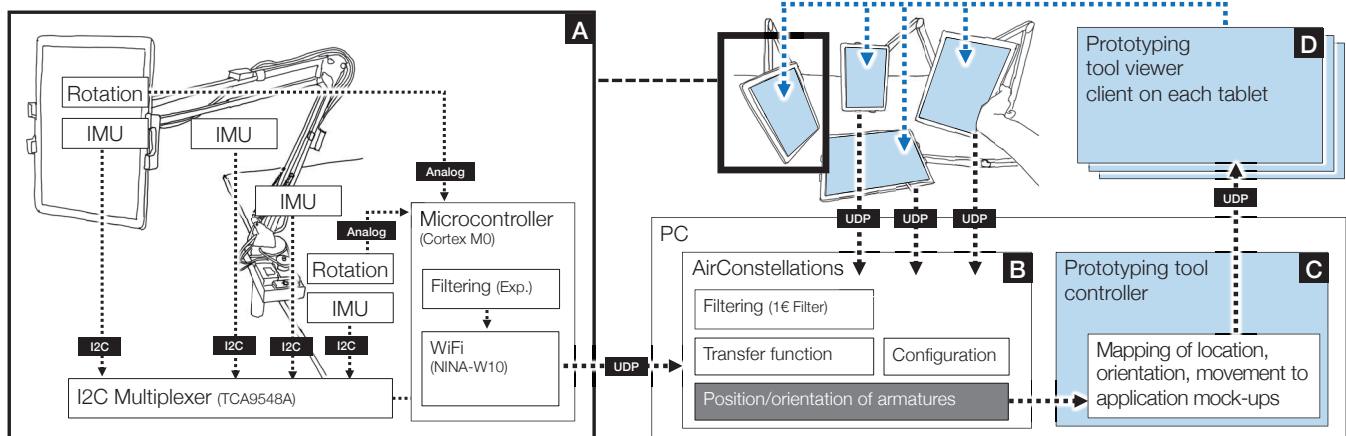
**Figure 4: AirConstellations technical implementation: adding spatial sensing to a 3-joint flexible armature.**

This docking behavior is similar to Codex [35], where devices can be disconnected from the connecting joint when needed. Mobile devices can join semi-fixed device formations when needed (moving along the design space dimension [4] from left to right), and disconnected when again needed as mobile, portable devices.

## 4.4 Stabilization and Software Filtering

We use multiple levels of software filtering to stabilize the sensor signal and minimize jitter when using the sensed position of the devices to drive application responses. On the microcontroller, we do 12-bit sampling of the analog rotation sensors, oversampling at 110Hz, and use an exponential filter as a first-level filtering for all measured orientation angles — from the two rotation sensors and the four IMUs. The base then connects to a PC over WiFi (using the microcontroller’s NINA-W10 radio of the *u-blox* chip), and streams all sensor values at 60 Hz via UDP to a C# software tool (Figure 5). In the software (Figure 5b), we use an additional filter level over all sensor values using the 1€ filter [11] ( $\beta = 0.007$ ,  $d_{cutoff} = 1.1$ ,  $f_{c_{min}} = 0.02$ ).

<sup>1</sup><https://github.com/nicmarquardt/airconstellations>



**Figure 5: Hardware and software implementation: (a) detailed diagram of the sensors on each armature, (b) the AirConstellations software which filters the input from each armature according to the user-defined configuration, (c) the prototyping tool that maps the sensor data to an application mock-up, and (d) the viewer software of the prototyping tool on a client receives sensor data and updates the screen based on sensor data.**

#### 4.5 Rapid Prototyping Software for Dynamic Device Formation Application Behaviors

We built a rapid prototyping tool (C# Windows software) to explore application scenarios with our dynamic in-air device formations. The goal of the prototyping tool was to allow quick and lightweight experimentation with dynamic application behaviors – that is, exploring concepts of how applications running on the devices connected to AirConstellations could dynamically respond to users’ interactions with the formation, when they move devices in and out of formations, change the orientation of devices, and so on. Our prototyping tool (summarized in Figure 5c-d) created a scaffolding that empowered us to create interaction-driven animations by (1) connecting multiple devices together, (2) processing the sensor data, and (3) enabling translating the sensor data into rich continuous animations as the interactions are taking place.

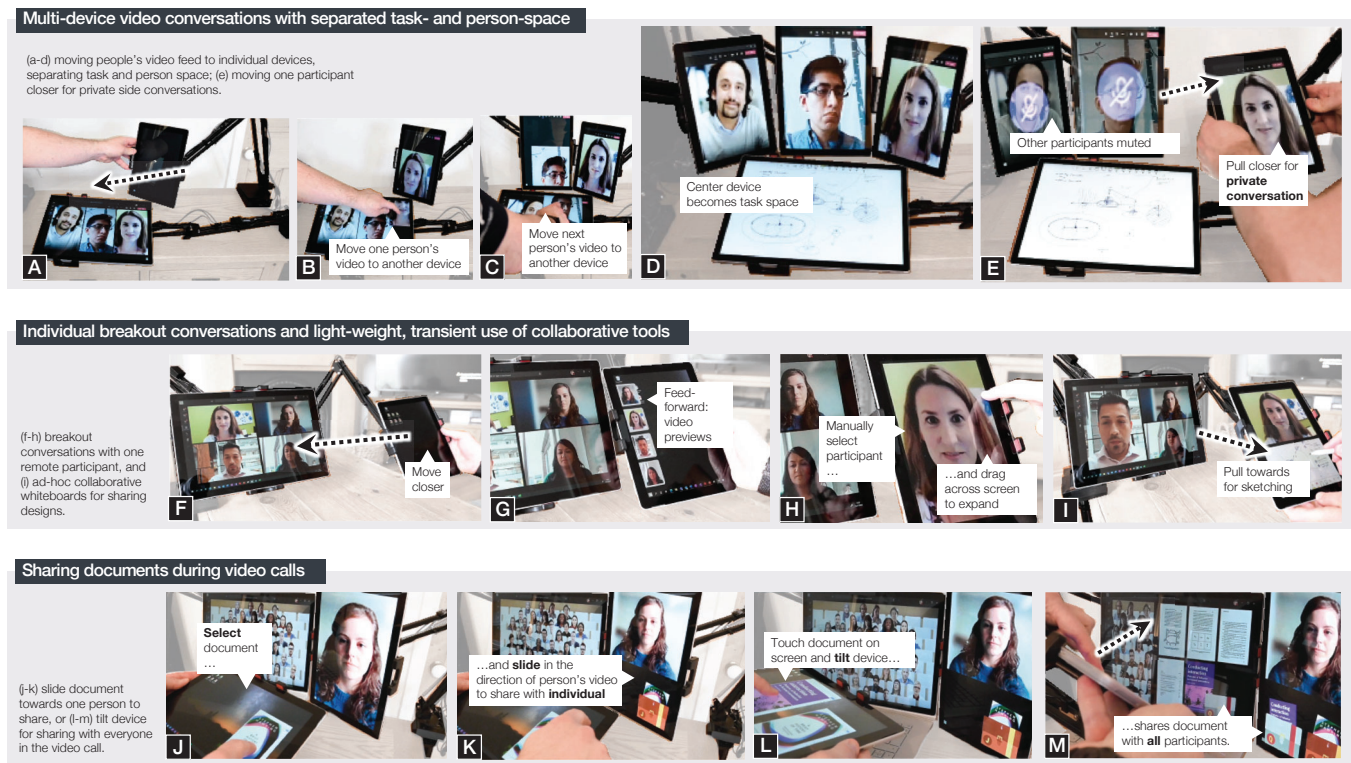
**Designing animations of application behaviors.** To visually author these kinds of real-time behaviors – without the overhead of full application implementations – we designed our tool so it uses interactive animations of native application behaviors that can be implemented with a wide variety of prototyping tools, such as PowerPoint animations, or video mock-ups from AfterEffects. We built on existing approaches, such as Astral [55] and the one by Romat et al. [79], and applied them to *cross-device* prototyping. Consequently, our tool facilitates quick exploration, testing of alternatives, and fine-tuning dynamic behaviors. After designing a new application behavior (e.g., a PowerPoint animation showing feedback of devices approaching each other), we import these as video sequences into our prototyping tool (Figure 5c).

**Sensor-driven animations.** Our tool segments all prototype video animations into sequences of still frames, and – most importantly – provides detailed settings for specifying how the live sensor tracking data is mapped to the behaviors of the different prototype animations. To do this, the designer (1) selects the sensor input to control the animation (such as sensed position, movement along

a path, orientation changes, etc.), (2) chooses a transfer function (e.g., linear, non-linear, ease-in/out) that controls the mapping of sensor input to animation frame, and (3) specifies the animation sequence to show on individual devices. The prototyping tool also provides a number of optional settings to fine-tune the processing of the incoming sensor data (e.g., filter thresholds, cutoff values) or playback of the animation (e.g., reverse playback, select subset of the animation sequence). This prototyping approach allows exploring different facets of user flow (combining animations and states) within a single application. For example, one application sequence might be mapping the proximity of two devices of a formation to show the animation, another sequence might respond to changes in device orientation, and so on.

**Interactive playback of animations.** The prototyping viewer client software (Figure 5d) displays the different animated application sequences on the devices attached to AirConstellations armatures. Each of the viewer clients receives the real-time sensing data (as part of a UDP packet stream) of the AirConstellations platform. The client software then dynamically animates the application interface in response to the physical movement of the devices and physical changes of the formations (Figure 5).

**Prototyping and refinement.** We designed 52 prototype application behaviors, which demonstrate different applications and operating system functions in a cross-device setting. From these behaviors we selected, refined, and combined a representative sample of our design space which we discuss in the next section. This prototyping process propelled a rapid exploration of many techniques and application behaviors while enabling iteratively refining the designs as we increased our prototype fidelity. Because all of the resulting applications use real-time sensing input (moving devices, proximity, orientation, touch input, etc.), it is possible to experience the different nuances of how the spatially-aware applications can respond to dynamic changes of formations of people and devices.



**Figure 6: Multi-device video conferencing example applications: (a-e) multi-device Teams video conversation, with separated task and person space; (f-i) ad-hoc breakout conversations and collaborative whiteboards for sharing designs; (j-m) gestures for sharing documents with individual and groups during video calls.**

## 5 AIRCONSTELLATIONS APPLICATIONS AND INTERACTION TECHNIQUES

To illustrate the dimensions of the design space for dynamic in-air device formations, we now present a number of concrete application scenarios and interaction techniques. In particular, we demonstrate how AirConstellations seamlessly manages the locations and orientations of devices to enable applications to easily respond to spatial (re)configurations and inter-device proximity.<sup>2</sup>

### 5.1 Multi-Device Video Conferencing

Our multi-device video conference application allows dynamic reconfiguration of the task and person space. Thus, it facilitates transitions between group chats and face-to-face side conversations, incorporates light-weight use of tools such as whiteboards during video calls, and enables document sharing through both touch and tilt gestures.

**Device formations for separated person and task space.** This example illustrates the separation of person and task space [10] during video calls, related to the Hydra four-way video conferencing [84], and techniques on large surfaces [51, 89], blended interaction [40], and tilting single-device screens [79]. We translate

<sup>2</sup>We focus on demonstrating how users might work with the AirConstellations platform; primarily through the exploration of the design space, and a combination of novel and replicated examples [54].

this technique to dynamic in-air multi-device workspaces, offering more degrees of freedom for possible reconfigurations and adjustments. When moving a second device of our AirConstellations setup closer to a tablet running a video conference call (Figure 6a), the video feed of individual participants can be transferred from the horizontal screen to any nearby vertically positioned tablet using a sliding gesture into the direction of the device (Figure 6b-c). Once all three remote participants have been moved to a separate screen (recreating the experience of four people facing each other in a physical meeting), the fourth device in the center transitions into *task space mode* and opens the collaborative whiteboard sketching app of the team (Figure 6d). Thanks to the flexibility of the armature, one can position the camera optimally (with each of the three *person space* devices) independently from the ergonomic placement of the *task space* device (i.e., placing the whiteboard sketching app horizontally in front for sketching or writing on the surface).

**Ad-hoc proxemic-aware transitions to individual side conversations.** Changing from a group conversation to a two-person side conversation can be difficult to do in digital video conferencing compared to how easily and naturally we can initiate a side conversation with someone in a physical meeting. As shown in Figure 6, AirConstellations supports a lightweight technique to transition from group to individual conversations, inspired by proximity-aware physical surrogates [25] and leveraging proxemic interactions [26, 49]. During the video conversation, a user can take any



of the three speakers’ proxy view, separating them from the device formation [P1], and bring the screen closer towards themselves (moving along spectrum of [12], decreasing proxemic distance, moving that individual speaker from Hall’s *social* to the *personal* space [31]), to initiate a private side conversation (Figure 6e). When doing so, the audio channel to the other person remains open, but gets muted for all other participants, allowing for a conversation in private. All other meeting participants can continue the ongoing conversation (with audio/video). As the user moves the device back to the group of other devices, the side conversation ends.

**Feedforward vs. manual confirmation of application behavior.** For revealing multiple options for what should happen in response to spatially-aware device movements, we introduce *feedforward* techniques (illustrating different techniques of [16] and depending on sensor input [15]). As Vermeulen et al. state in their paper [98], “*feedforward [...] tells users what the result of their action will be.*” In this variation of the video conferencing app, when a user brings a second tablet closer (Figure 6f) a side-panel fades in with preview-icons of all four remote participants of the conversation (Figure 6g). This semi-transparent panel provides a feedforward preview of the options to move one of the remote participants’ video stream to the other device. The user can simply click&drag one of the camera preview icons from the sidebar (Figure 6h) to move the participant to the new device. If no option is selected, the panel fades out after five seconds.

**Light-weight, transient use of collaborative tools.** AirConstellations also supports ad-hoc, transient application behaviors during video conversations. For example, when separating the tablet away from the device formation and moving it closer, while tilted like a drawing board [P2], we fade in a shared Whiteboard from the bottom half of the screen (Figure 6h), giving easy access for shared writing and drawing. Moving the tablet back as part of the device formation makes the whiteboard canvas disappear.

**Sharing gestures for individuals and groups.** Our app facilitates sharing of documents during video conferencing. A user can share documents with other *individual* participants by using directional sliding gestures (Figure 6j) into the direction of the proxy-screen of another participant of the conversation, which creates a transient shared portfolio [9] (Figure 6k). A fade-in animation indicates this document sharing, where the shared portfolio slowly fades in when performing the sliding gesture — and the sharing can be revoked easily by reversing the gesture. Sharing documents with *all participants* during video conferencing can be done through a more expressive gesture — such as tilting devices. A user can select the document to share on the screen (Figure 6l), followed by tilting the device into the direction of the device formation with all remote participants (Figure 6m), which shares the document with everyone. By design this gesture requires more effort compared to the sliding gesture for sharing with only one individual — we match the effort of the gesture with its impact (i.e., minimal gesture for sharing with one person, more effortful gesture when sharing with all participants).

## 5.2 Productivity Applications

Proximity-dependent cross-device behaviors can facilitate managing multiple application windows, tool palettes, as well as help redirect input devices when using productivity applications (e.g., presentation software, email clients, or development tools).

**Cross-device hand-over of tool palettes.** When a person moves a tablet into the proximity of another device running an application (Figure 7a), tool palettes that can be transferred onto the second device are shown in a *semi-transparent overlay* on the side of the screen (Figure 7b). The person can then select and expand the tool palettes by dragging, spreading them out on screen (Figure 7c-d). For the tool palettes transfer technique, we again use feedforward to indicate action possibilities: we map the sliding-in/out of the palette window (Figure 7b) directly to the continuous movement [15] of the approaching device (and a touch gesture confirms the action, otherwise the preview slides away with no action).

**Multiple options and default behaviors.** When implementing proximity- and orientation-dependent application behaviors for device formations, there can be cases where multiple options are available to choose from (e.g., different actions resulting from bringing two devices closer together — we describe these cases in the BEHAVIOR dimension [16]). For example, when a user moves a tablet next to a laptop running a development tool, a sidebar suggests three possible outcomes: moving the code editor to the tablet, moving tool palettes, or no action (Figure 7e). In this case, a default option (moving code editor, shown in green) is automatically selected when continuing the movement of the tablet in the direction of the laptop (Figure 7f). When the developer later moves a second tablet closer, options are again shown (Figure 7g), but no default action is available — the user decides to manually move the tool palettes to the second screen (Figure 7h).

**Proximity-dependent input redirection.** Aside from digital tablets, we can use other input/output devices with AirConstellations [P7]. When attaching a physical keyboard to an AirConstellations arm, the input of that keyboard can be re-directed to different devices based on proximity. For example, when moving the physical keyboard closer to a tablet, the onscreen soft keyboard fades out and the device instead uses the physical keyboard (Figure 7i-j). In this case, keyboard input can always be redirected to the closest tablet — or alternatively we can use spatial gestures like *swiping* in the direction of the tablet the keyboard input should be directed to.

**Spatially-aware soft-keyboard extensions.** When moving another screen close to the physical keyboard, this device will then function as an extended keyboard to show pre-sets of emojis (Figure 7k). This kind of virtual keyboard extension can depend on the application currently in focus and other context: for example, with a chat messenger app the keyboard would show emojis and GIF images, for a video editor pre-sets of command functions, and for a vector drawing application it could show a visual clipboard.

**Transfer portals with whiteboard sketching applications.** Close proximity between devices can open ad-hoc file transfer portals (similar to [66, 100]). When moving a tablet *perpendicular* [13] towards the large display (Figure 7l), a transfer portal proxy is shown on both on the tablet and the large display — establishing a direct



**Figure 7: Productivity applications with AirConstellations: (a-d) distributing tool palettes across devices, (e-h) multiple available options for proximity-based application behavior, (i-k) proximity-dependent input redirection of physical keyboards, and spatially-aware soft-keyboard extensions, and (l-o) opening transfer portals across devices in a whiteboard application.**

virtual transfer link, which allows to move sketches from the tablet to the wall display with a sliding gesture (Figure 7m-o).

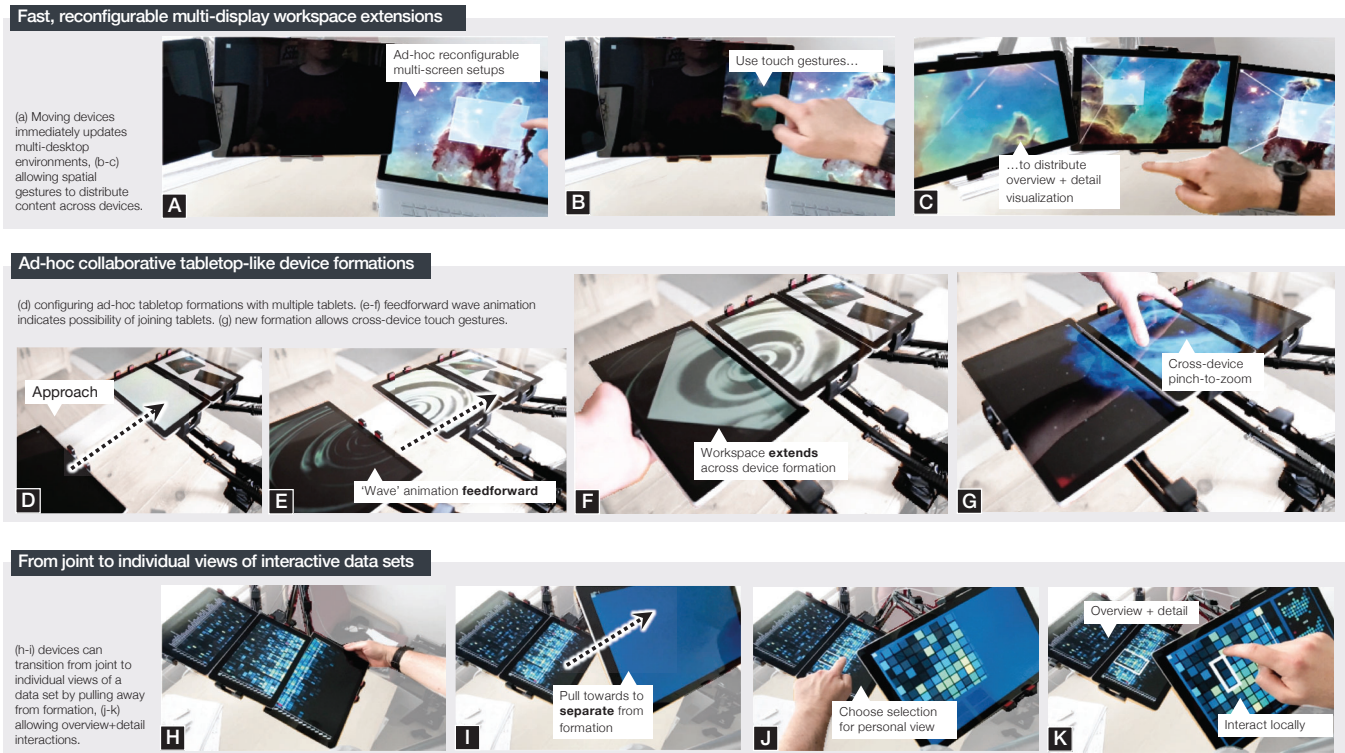
### 5.3 Workspace Formations for Data Analysis

We illustrate the use of ad-hoc device federations in the context of data analysis and information visualization applications (e.g., similar to the distributed and coordinated visualization views in VisPorter [15], VisTiles [52], or ReticularSpaces [4]). In particular, we show how to use feedforward animations to facilitate forming device formations, and how to enable ad-hoc transitions from joint to personal, individual views.

**Fast, reconfigurable multi-display workspace extensions.** Similar to traditional multi-monitor desktop setups [30], users can reposition any of the AirConstellations tablets so that they function as an additional, ad-hoc screen extension [P1]. Placing tablets next to each other (Figure 8a) adds the display to the current formation

arrangement – and updates the spatial configuration of the desktop layout. Users can then use touch gestures to extend content across the displays (like the high-pixel astrophysics image in Figure 8b). In our example, this enables multiple *overview+detail* views of different magnification levels (Figure 8c). The spatial awareness of AirConstellations means that physically rearranging devices does not require any manual reconfiguration of display arrangement in the system settings – and this lowers the barrier for more frequent physical reconfigurations, even when needed for only a short amount of time.

**Ad-hoc collaborative tabletop-like device formation.** We use another feedforward technique – which we combine with *slow motion feedback* [97] techniques for *previews* and *emphasizing change* – to indicate opportunities for joint device formations (Figure 8d). In particular, we show a *fluid wave/ripple animation* (Figure 8e) when a user brings horizontally oriented tablets close to another to indicate



**Figure 8: Multi-device formations workspaces for data analysis: (a-c) easily reconfigurable multi-display arrangements, (d-g) ad-hoc tabletop-like interactive surface with feedforward indicating possible merging of multiple tablet devices into a continuous surface, and (h-k) moving from shared to personal data views.**

the possibility to merge these devices into a tabletop-like formation (moving from *separate* to *continuous* formation [P1]). The intensity of the wave animation increases when moving closer (in a *non-linear* mapping [5]). When both devices are very close, we fade-in a second animation showing an expanding arrow indicating the new formed merged view of the devices (Figure 8f). Both animations fade out once the surfaces are physically touching, and this new formation can then be used like a continuous interactive surface, for example, using cross-device pinch-to-zoom [66] (Figure 8g).

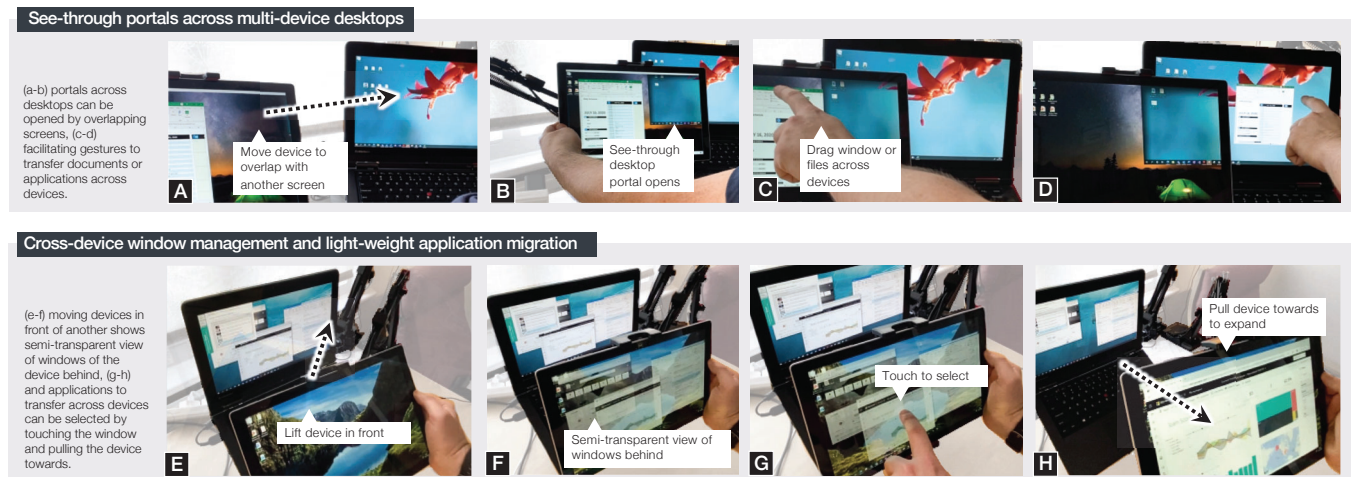
**From joint to individual views of interactive data sets.** Devices can transition from a joint device formation (similar to our previous example) to an individual, decoupled representation of the dataset. When analyzing a dataset on a horizontal multi-tablet formation (Figure 8h), a user can move one of the tablets apart from the formation, creating a separate, decoupled viewing window on the device (Figure 8i). Manually selecting parts of the dataset through direct touch on the shared view shows a magnified view of the data selection on the separate device (Figure 8j). The user can then directly interact with the dataset on the break-out device (Figure 8k).

#### 5.4 System-Wide Cross-Device Behaviors

We designed two additional techniques for system-wide cross-device application and window management: a window manager hand-over function and a see-through desktop portal for easy transfer of applications or files.

**Light-weight see-through portals across desktops.** Layering devices [P3] on top of or in front of each other can facilitate OS-level operations across devices. The see-through desktop portal is activated when one device is directly placed on top of another device. When a person slides the device on top of another screen, the part of the occluded desktop is shown on the device in front (Figure 9a-b). The user can then drag&drop files or windows from one desktop to another (Figure 9c-d). We use the fact that devices are physically *overlapping* (instead of being positioned *side-by-side*, cf. [P1] and [P3]) as a unique trigger for this OS-level function.

**Layering for cross-device window management and application migration.** We can use proximity and orientation of tablets to facilitate window management across devices. Our system recognizes when a person moves a tablet in *front* of another (Figure 9e), *occluding* the device in the back ([P1]-occluding and [I3]-front). This initiates the window-transfer view (Figure 9f): all application windows from the tablet in the back are shown semi-transparently on the tablet in front. A user can select one of the windows (Figure 9g) and pull the device closer to transfer the selected application to the tablet in front (Figure 9h). This example illustrates how we can use the fine-grained inter-device proximity and occlusion information to trigger a specific device operation (the window view), which is then confirmed through explicit user input (direct touch).



**Figure 9: System-wide cross-device behaviors: (a-d) see-through portals across multi-device desktops, (e-h) cross-device window management and light-weight application migration.**

## 6 PRELIMINARY INTERVIEW STUDY

To gather feedback on AirConstellations, we conducted 1-hour interviews with eight participants<sup>3</sup> (6 female, 2 male). We asked about their multi-device use and daily tasks, showed them a video with AirConstellations applications (similar to the video figure), collected their immediate impression on a scale from 1-negative to 5-positive, and followed up by gathering insights on the most and least compelling aspects.

**Most valued scenarios and techniques.** Participants tended to favor scenarios that reflected their most frequent activities. For example, P1, P6, and P7 spend a large portion of their day for data analysis, programming, and document editing, rating workspace and desktop management scenarios the highest. On the other hand, P2, P3, P4, P5, and P8 who use video meetings frequently rated remote collaboration scenarios as most compelling, relating to some of their needs when conducting remote collaboration. P3 commented that the multi-device video conferencing scenario “*is really where the innovation is*” and that the ad-hoc whiteboard technique during video calls as “*super useful to share and express ideas with other designers.*” P4 commented that assigning video feeds to different devices was “*taking the physical [meeting configuration] and turning it into the digital*” and that being able to look at participants seated at specific physical location was invaluable.

**Balancing seeing vs. being seen.** P2, P4, and P5 described tradeoffs they often made during video calls, having to optimize their setup for either better content viewing (seeing) or better webcam capture (being seen). P2 commented on the time spent to (re)adjusting her devices, angling some to avoid glare for content viewing, or for gaining a better angle for the webcam depending on lighting conditions throughout the day. P4 commented on the importance of eye contact during video calls but did not always switch webcams when switching devices (e.g., for screen sharing) for short periods of time as it was disruptive. Both noted that AirConstellations would

directly facilitate these adjustments, and suggested that additional sensors could perceive lighting conditions or devices being interacted on, to provide fluid webcam switch: “*posture, environment, ergonomics, orientation, source of lighting, all of that matters*” (P2).

**Organic setups and physical reconfigurability.** Participants described AirConstellations as “*mobile yet stable*” (P1), and they see the potential of the proposed dynamic in-air device formations reducing the cost of making physical adjustments to their workspaces, thus enabling “*more organic setups*” (P6) evolving even over short periods of time. Related, P1 mentioned that AirConstellations affords “*fine motor movements [screen angle adjustments] and broader types of movements [moving screens closer together].*” P7 commented that such fluidity could transform the way she works with multiple devices: “*It would make my life 100% more productive.*” For his use of video conferencing, P1 commented that with AirConstellations he could transition to a more optimal multi-device configuration (migrating his chat window, the video feed, or the webcam), even for a short period of time, as it appears “*effortless, [enabling] moving content without having to think.*”

**Replacing ‘clicks’ through physical device movement to facilitate transfers.** All participants commented on the hurdles of migrating data or applications between their currently used devices: “*today it requires a manual transfer of context [...] a bunch of clicks and a concerted effort to transfer content.*” (P1). Bringing devices closer to share data and apps via *desktop portal* or *transparent layer* was compelling to all participants: “*Extremely easy to use. It would feel like magic.*” (P7). P1 highlighted that “*a [single] physical interaction is more appealing than using a bunch of mouse and keyboard interactions.*” and that proximity sensing “*feels natural to move content [between devices].*”

**Unintentional interactions.** A few participants raised issues with regard to interactions launched from context sensing. “*There is a fine line to make it easy vs enabling them [the interactions] all the time. It should be part of the setup. I never want motion of my*

<sup>3</sup>Backgrounds: 3 researchers, 4 designers, 1 data scientist.

*device unless I say it.*” (P6). P2 echoed this concern by explaining that making some interactions “*too easy*” (by replacing a series of UI interaction with a single physical motion) would perhaps lead to unintentional application behaviors. For example, sharing a wrong document during a meeting or migrating a wrong app. These comments reinforce the importance of making interactions easily reversible, and of supporting effective visual feedforward and feedback.

**Consistency of interaction behaviors.** P7 pondered if the different behaviors are too numerous and prove somewhat inconsistent or hard to predict what would happen when moving a device. The concern that it could be disruptive led her to suggest that it might be better to select a small coherent and consistent subset.

**Directness & immediate feedback.** All participants externalized their delight, mentioning “*super cool*” (P5), “*pretty neat*” (P6), and “*magical*” (P7). They pointed out the fluid animations and direct feedback associated with physical interactions (e.g., making a window from device 1 pop up as device 2 approaches) as what made it delightful. They also responded enthusiastically to see-through/AR metaphors, such as the desktop portal and the flight simulator example: P5 commented that “*Cool. I probably said that a lot, but it’s like the VR headset, without wearing the headset!*”

**Tradeoffs with armature design & cost.** P2 commented that fixed external monitors on a desk were likely to be less “*wobbly*,” but saw value in creating temporary setups, noting that it would be important for them to be lightweight and easily attachable on any surface. Other participants commented on the need to own multiple of these arms to unlock the power of the multi-device scenarios they saw: “*Pretty neat, but the device arm device would have to be pretty cheap. A very low incremental expense.*” (P6). P1 also commented on the need to have many arms in multiple places, to avoid having to move them too frequently: “*If they were reasonably cheap and I would put them everywhere.*”

## 7 DISCUSSION

Here we reflect on the potential of dynamic in-air device formations, discuss limitations of the approach (e.g., tradeoffs with spatial-awareness or proactive application behaviors), and suggest directions of future work.

**Spatially-aware tracking vs. manual configuration.** Beyond the affordance of our armatures for plug & play use as well as quick rearrangement, the integrated tracking system affords mutual spatial awareness between multiple armatures. This allows our interaction techniques to leverage the spatial setup of devices as an implicit parameter of actions that users take. Users can put displays where they want them, in a direct manner in-air, rather than indirectly configuring display layouts through control panels and so forth. This directness and simplicity, and the corresponding reduction in time and effort, may lower the threshold for users to reconfigure or adjust displays more frequently, whether for ergonomic, social, or task-driven needs. At the same time, spatially-aware tracking might not always be needed or desired, such as for use cases that do not require knowledge of the precise locations ([14], [15]). The design space accommodates setups without integrated tracking, and a user could use instead a software-based setup similar to configuring monitor

setups in system preferences today. Alternatively, techniques from previous work could serve to accomplish this in a usable manner (e.g., stitching [36] or in world-in-miniature configuration view that support drag&drop device placement [7]).

**Transitions between fixed, semi-fixed, and mobile.** For quick transitions with the fluid use of devices in either semi-fixed or fully mobile roles, the simple clamp-based attachment of AirConstellations facilitates devices that come and go to varying degrees. In the semi-fixed case, users can easily pull a device partially away from a formation to focus on particular details in a visualization (Figure 8) or pull aside someone in a small-group video conference (Figure 6e), for example. But users can “grab and go” (detach a device) to go fully mobile (as shown in our supplementary video) at a moment’s notice. As such, a tablet, netbook, or smartphone could serve as a logical hub to migrate device, task, and interactive state between different work or home-office locations, including other AirConstellations setups.

**From single user to collaborative AirConstellations.** For future work, exploring the use of ad-hoc dynamic device formations for co-located collaborative applications will be interesting. New device formations could support collaborative tasks — using shared devices and users’ own personal devices in concert — merging *device formations* and *people formations* (cf. Kendon’s f-formations [43, 67]). In addition, in-air device formations have the potential to best support people’s dynamic and highly flexible use of interpersonal space for interaction [31], akin to use in *Proxemic Interaction* [26] or spatially configured displays affecting people’s behavior [91].

**Ecologies of devices and sustainability.** AirConstellations provide an opportunity to contribute to sustainable computing due to the modular nature of their assembly. Since AirConstellations separates affordance of a particular device, tracking, and the actual interaction with the display into separate components, our design can easily incorporate older devices into formations. For example, one could upcycle older tablets or touch monitors to be used as part of a formation, yet retain all the functionality we have demonstrated in our scenarios to create individual ecologies optimized for compartmentalized tasks.

**Tradeoffs of proactive vs. reactive application behaviors.** Sensing the proximity, relative orientation, or adjacency of displays affords certain automatically sensed transitions in application states or behaviors. However, people’s natural socio-spatial literacy [49] implies that such device motions and configurations are also already part of the vocabulary of human-human interaction. This represents a potential design tension in AirConstellations: should a particular transition happen automatically, or not? Several of our interaction techniques show that hybrid designs may be possible, where proximity of devices triggers feedforward of possible connection states and other relationships between devices. Users then have options to react to this feedback, to act on suggested transitions suitable to the current application and device arrangement, or to default to “*no change*” if they simply ignore such suggestions. Alternatively, users could manually configure their preferences for proactive, proximity-dependent behaviors, or use non-spatially responsive AirConstellations designs.

**Further use cases and hardware form factors.** There is also an opportunity to explore other armature form factors (e.g., similar to custom haptic feedback arms [5]). In our approach, we re-purposed existing armatures as much as possible, but custom-built designs may increase stability, reach, or accuracy of tracking. Further exploration could include servo-driven actuated changes of device formations (e.g., extending actuated desktops [2] or interactive surfaces [83, 86]). Dynamic in-air device formations could also be applied to many other use cases, such as in virtual- or augmented-reality applications (e.g., as a form of flexible, dynamic interaction surfaces), education, or specialized domain expert workspaces.

**Extending design space.** Our *design space* is a first-order approximation of the spectrum of design parameters relevant for dynamic in-air device formations. But, as with any such mapping, there are possible extensions and additional dimensions to consider (similar to how we incorporated dimensions of earlier design spaces [8, 20, 35, 64, 79]). Such extensions could be, among many others, a fine-grained categorization of *collaborative use cases*, differentiating levels of *actuation* (e.g., with links to the field of robotics), or an in-depth unpacking of the *roles* and *functions* of devices as part of formations.

## 8 CONCLUSION

Using the AirConstellations platform we presented in this paper, we have explored the potential of dynamic in-air device formations for the next generation of multi-device workspace setups in a variety of use-cases. A core characteristic of all AirConstellations setups is their ability to support and retain the affordances of mobile interaction, while at the same time allowing semi-fixed devices to be (re-)configured in different physical arrangements. Our applications scenarios showcase the potential of such highly dynamic workspace configurations, with spatially-aware behaviors, feedforward, and other interaction techniques for flexible and delightful use of technology best supporting tasks at hand. The scenarios we demonstrated represent only a subset of the overall design space of dynamic in-air device formations, and we hope our work will inspire many promising research avenues to explore.

## ACKNOWLEDGMENTS

We thank the Swiss Joint Research Center with Microsoft Research for funding Hugo Romat at ETH Zurich.

## REFERENCES

- [1] Jason Alexander, Andrés Lucero, and Sriram Subramanian. 2012. Tilt Displays: Designing Display Surfaces with Multi-Axis Tilting and Actuation. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '12)*. Association for Computing Machinery, New York, NY, USA, 161–170. <https://doi.org/10.1145/2371574.2371600>
- [2] Gilles Bailly, Sidharth Sahdev, Sylvain Malacria, and Thomas Pietrzak. 2016. LivingDesktop: Augmenting Desktop Workstation with Actuated Devices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 5298–5310.
- [3] Till Ballendat, Nicolai Marquardt, and Saul Greenberg. 2010. Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. Association for Computing Machinery, New York, NY, USA, 121–130. <https://doi.org/10.1145/1936652.1936676>
- [4] Jakob Bardram, Sofiane Gueddana, Steven Houben, and Søren Nielsen. 2012. ReticularSpaces: Activity-Based Computing Support for Physically Distributed and Collaborative Smart Spaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2845–2854. <https://doi.org/10.1145/2207676.2208689>
- [5] Gareth Barnaby and Anne Roudaut. 2019. Mantis: A Scalable, Lightweight and Accessible Architecture to Build Multiform Force Feedback Systems. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, New York, NY, USA, 937–948. <https://doi.org/10.1145/3332165.3347909>
- [6] Andrea Bellucci, Alessio Malizia, and Ignacio Aedo. 2014. Light on Horizontal Interactive Surfaces: Input Space for Tabletop Computing. *Comput. Surveys* 46, 3 (Jan. 2014), 32:1–32:42. <https://doi.org/10.1145/2500467>
- [7] Jacob T. Biehl and Brian P. Bailey. 2004. ARIS: An Interface for Application Relocation in an Interactive Space. In *Proceedings of Graphics Interface 2004 (GI '04)*. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 107–116.
- [8] Frederik Brudy, Christian Holz, Roman Rädle, Chi-Jui Wu, Steven Houben, Clemens Nylandsted Klokmose, and Nicolai Marquardt. 2019. Cross-Device Taxonomy: Survey, Opportunities and Challenges of Interactions Spanning Across Multiple Devices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–28. <https://doi.org/10.1145/3290605.3300792>
- [9] Frederik Brudy, David Ledo, Michel Pahud, Nathalie Henry Riche, Christian Holz, Anand Waghamare, Hemant Bhaskar Surale, Marcus Peinado, Xiaokuan Zhang, Shannon Joyner, Badrish Chandramouli, Umar Farooq Minhas, Jonathan Goldstein, William Buxton, and Ken Hinckley. 2020. SurfaceFleet: Exploring Distributed Interactions Unbounded from Device, Application, User, and Time. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 7–21. <https://doi.org/10.1145/3379337.3415874>
- [10] William A. S. Buxton. 1992. Telepresence: Integrating Shared Task and Person Spaces. In *Proceedings of the Conference on Graphics Interface '92*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 123–129.
- [11] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 1 € Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2527–2530. <https://doi.org/10.1145/2207676.2208639>
- [12] Nicholas Chen, Francois Guimbretiere, and Abigail Sellen. 2012. Designing a Multi-Slate Reading Environment to Support Active Reading Activities. *ACM Transactions on Computer-Human Interaction* 19, 3 (Oct. 2012), 18:1–18:35. <https://doi.org/10.1145/2362364.2362366>
- [13] Nicholas Chen, François Guimbretière, and Abigail Sellen. 2013. Graduate Student Use of a Multi-Slate Reading System. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 1799–1808. <https://doi.org/10.1145/2470654.2466237>
- [14] Pei-Yu (Peggy) Chi and Yang Li. 2015. Weave: Scripting Cross-Device Wearable Interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 3923–3932. <https://doi.org/10.1145/2702123.2702451>
- [15] Haeyong Chung, Chris North, Jessica Zeitz Self, Sharon Chu, and Francis Quek. 2014. VisPorter: Facilitating Information Sharing for Collaborative Sensemaking on Multiple Displays. *Personal Ubiquitous Comput.* 18, 5 (June 2014), 1169–1186. <https://doi.org/10.1007/s00779-013-0727-2>
- [16] Marek Czernuszenko, Dave Pape, Daniel Sandin, Tom DeFanti, Gregory L. Dawe, and Maxine D. Brown. 1997. The ImmersaDesk and Infinity Wall Projection-Based Virtual Reality Displays. *ACM SIGGRAPH Computer Graphics* 31, 2 (May 1997), 46–49. <https://doi.org/10.1145/271283.271303>
- [17] Paul Dietz and Darren Leigh. 2001. DiamondTouch: A Multi-User Touch Technology. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*. Association for Computing Machinery, New York, NY, USA, 219–226. <https://doi.org/10.1145/502348.502389>
- [18] K. Everitt, Chia Shen, K. Ryall, and C. Forlines. 2006. MultiSpace: Enabling Electronic Document Micro-Mobility in Table-Centric, Multi-Device Environments. In *First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)*. IEEE, USA, 8 pp.–. <https://doi.org/10.1109/TABLETOP.2006.23>
- [19] George W. Fitzmaurice. 1993. Situated Information Spaces and Spatially Aware Palmtop Computers. *Commun. ACM* 36, 7 (July 1993), 39–49. <https://doi.org/10.1145/159544.159566>
- [20] 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 Press/Addison-Wesley Publishing Co., USA, 442–449. <https://doi.org/10.1145/223904.223964>
- [21] George W. Fitzmaurice, Azam Khan, William Buxton, Gordon Kurtenbach, and Ravin Balakrishnan. 2003. Sentient Data Access via a Diverse Society of Devices: Today&#x2019;s Ubiquitous Computing Environment Cannot Benefit from the Traditional Understanding of a Hierarchical File System. *Queue* 1, 8 (Nov. 2003), 52–62. <https://doi.org/10.1145/966712.966721>

- [22] Hans Gellersen, Carl Fischer, Dominique Guinard, Roswitha Gostner, Gerd Kortuem, Christian Kray, Enrico Rukzio, and Sara Streng. 2009. Supporting Device Discovery and Spontaneous Interaction with Spatial References. *Personal and Ubiquitous Computing* 13, 4 (May 2009), 255–264. <https://doi.org/10.1007/s00779-008-0206-3>
- [23] Audrey Girouard, Aneesh Tarun, and Roel Vertegaal. 2012. DisplayStacks: Interaction Techniques for Stacks of Flexible Thin-Film Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2431–2440. <https://doi.org/10.1145/2207676.2208406>
- [24] Alix Goguy, Cameron Steer, Andrés Lucero, Laurence Nigay, Deepak Ranjan Sahoo, Céline Coutrix, Anne Roudaut, Sriram Subramanian, Yutaka Tokuda, Timothy Neate, Jennifer Pearson, Simon Robinson, and Matt Jones. 2019. Pick-Cells: A Physically Reconfigurable Cell-Composed Touchscreen. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3290605.3300503>
- [25] Saul Greenberg and Hideaki Kuzuoka. 2001. Using Digital but Physical Surrogates to Mediate Awareness, Communication and Privacy in Media Spaces. *Personal Technologies* 3, 4 (2001), 182–198.
- [26] Saul Greenberg, Nicolai Marquardt, Till Ballendat, Rob Diaz-Marino, and Miaosen Wang. 2011. Proxemic Interactions: The New UbiComp? *Interactions* 18, 1 (Jan. 2011), 42–50. <https://doi.org/10.1145/1897239.1897250>
- [27] Jens Emil Grønbaek, Henrik Korsgaard, Marianne Graves Petersen, Morten Henriksen Birk, and Peter Gall Krogh. 2017. Proxemic Transitions: Designing Shape-Changing Furniture for Informal Meetings. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 7029–7041.
- [28] Jens Emil Grønbaek, Majken Kirkegaard Rasmussen, Kim Halskov, and Marianne Graves Petersen. 2020. KirigamiTable: Designing for Proxemic Transitions with a Shape-Changing Tabletop. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–15. <https://doi.org/10.1145/3313831.3376834>
- [29] Jens Grubert and Matthias Kranz. 2017. HeadPhones: Ad Hoc Mobile Multi-Display Environments Through Head Tracking. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3966–3971. <https://doi.org/10.1145/3025453.3025533>
- [30] Jonathan Grudin. 2001. Partitioning Digital Worlds: Focal and Peripheral Awareness in Multiple Monitor Use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, New York, NY, USA, 458–465. <https://doi.org/10.1145/365024.365312>
- [31] Edward Titchell Hall. 1966. *The Hidden Dimension* (1st ed.). Doubleday, Garden City, NY.
- [32] Peter Hamilton and Daniel J. Wigdor. 2014. Conductor: Enabling and Understanding Cross-Device Interaction. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2773–2782. <https://doi.org/10.1145/2556288.2557170>
- [33] Ken Hinckley. 2003. Synchronous Gestures for Multiple Persons and Computers. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST '03)*. ACM, New York, NY, USA, 149–158. <https://doi.org/10.1145/964696.964713>
- [34] Ken Hinckley, Xiaojun Bi, Michel Pahud, and Bill Buxton. 2012. Informal Information Gathering Techniques for Active Reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 1893–1896. <https://doi.org/10.1145/2207676.2208327>
- [35] Ken Hinckley, Morgan Dixon, Raman Sarin, Francois Guimbretiere, and Ravin Balakrishnan. 2009. Codex: A Dual Screen Tablet Computer. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1933–1942. <https://doi.org/10.1145/1518701.1518996>
- [36] Ken Hinckley, Gonzalo Ramos, Francois Guimbretiere, Patrick Baudisch, and Marc Smith. 2004. Stitching: Pen Gestures That Span Multiple Displays. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI '04)*. ACM, New York, NY, USA, 23–31. <https://doi.org/10.1145/989863.989866>
- [37] Steven Houben, Paolo Tell, and Jakob E. Bardram. 2014. ActivitySpace: Managing Device Ecologies in an Activity-Centric Configuration Space. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces - ITS '14*. ACM Press, Dresden, Germany, 119–128. <https://doi.org/10.1145/2669485.2669493>
- [38] Hiroshi Ishii and Minoru Kobayashi. 1992. ClearBoard: A Seamless Medium for Shared Drawing and Conversation with Eye Contact. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. Association for Computing Machinery, New York, NY, USA, 525–532. <https://doi.org/10.1145/142750.142977>
- [39] Shahram Izadi, Harry Brignull, Tom Rodden, Yvonne Rogers, and Mia Underwood. 2003. Dynamo: A Public Interactive Surface Supporting the Cooperative Sharing and Exchange of Media. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST '03)*. ACM, New York, NY, USA, 159–168. <https://doi.org/10.1145/964696.964714>
- [40] Hans-Christian Jetter, Harald Reiterer, and Florian Geyer. 2014. Blended Interaction: Understanding Natural Human–Computer Interaction in Post-WIMP Interactive Spaces. *Personal and Ubiquitous Computing* 18, 5 (June 2014), 1139–1158. <https://doi.org/10.1007/s00779-013-0725-4>
- [41] Haojian Jin, Christian Holz, and Kasper Hornbaek. 2015. Tracko: Ad-Hoc Mobile 3D Tracking Using Bluetooth Low Energy and Inaudible Signals for Cross-Device Interaction. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 147–156. <https://doi.org/10.1145/2807442.2807475>
- [42] Haojian Jin, Cheng Xu, and Kent Lyons. 2015. Corona: Positioning Adjacent Device with Asymmetric Bluetooth Low Energy RSSI Distributions. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 175–179. <https://doi.org/10.1145/2807442.2807485>
- [43] Adam Kendon. 1990. *Conducting Interaction: Patterns of Behavior in Focused Encounters*. Cambridge University Press, UK.
- [44] Mohammadreza Khalilbeigi, Roman Lissermann, Wolfgang Kleine, and Jürgen Steimle. 2012. FoldMe: Interacting with Double-Sided Foldable Displays. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, New York, NY, USA, 33–40. <https://doi.org/10.1145/2148131.2148142>
- [45] H. Kim, S. Jang, D. Suh, and S. Chang. 2011. UbiWall as a Reconfigurable Ecosystem of Smart Blocks. In *5th IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST 2011)*. IEEE, USA, 95–100. <https://doi.org/10.1109/DEST.2011.5936605>
- [46] David Kirsh. 1995. The Intelligent Use of Space. *Artificial Intelligence* 73, 1 (Feb. 1995), 31–68. [https://doi.org/10.1016/0004-3702\(94\)00017-U](https://doi.org/10.1016/0004-3702(94)00017-U)
- [47] Clemens N. Klokmoose, James R. Eagan, Siemen Baader, Wendy Mackay, and Michel Beaudouin-Lafon. 2015. Webstrates: Shareable Dynamic Media. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM Press, New York, NY, USA, 280–290. <https://doi.org/10.1145/2807442.2807446>
- [48] Naohiko Kohtake, Ryo Ohsawa, Takuro Yonezawa, Yuki Matsukura, Masayuki Iwai, Kazunori Takashio, and Hideyuki Tokuda. 2005. U-Texture: Self-Organizable Universal Panels for Creating Smart Surroundings. In *UbiComp 2005: Ubiquitous Computing (Lecture Notes in Computer Science)*, Michael Beigl, Stephen Intille, Jun Rekimoto, and Hideyuki Tokuda (Eds.). Springer, Berlin, Heidelberg, 19–36. [https://doi.org/10.1007/11551201\\_2](https://doi.org/10.1007/11551201_2)
- [49] Peter Gall Krogh, Marianne Graves Petersen, Kenton O'Hara, and Jens Emil Groenbaek. 2017. Sensitizing Concepts for Socio-Spatial Literacy in HCI. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, Denver, Colorado, USA, 6449–6460. <https://doi.org/10.1145/3025453.3025756>
- [50] Yoshiki Kudo, Kazuki Takashima, Morten Fjeld, and Yoshifumi Kitamura. 2018. AdapTable: Extending Reach over Large Tabletops through Flexible Multi-Display Configuration. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces (ISS '18)*. Association for Computing Machinery, New York, NY, USA, 213–225. <https://doi.org/10.1145/3279778.3279779>
- [51] Hideaki Kuzuoka, Jun Yamashita, Keiichi Yamazaki, and Akiko Yamazaki. 1999. Agora: A Remote Collaboration System That Enables Mutual Monitoring. In *CHI '99 Extended Abstracts on Human Factors in Computing Systems (CHI EA '99)*. Association for Computing Machinery, New York, NY, USA, 190–191. <https://doi.org/10.1145/632716.632836>
- [52] R. Langner, T. Horak, and R. Dachselt. 2018. VisTiles: Coordinating and Combining Co-Located Mobile Devices for Visual Data Exploration. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (Jan. 2018), 626–636. <https://doi.org/10.1109/TVCG.2017.2744019>
- [53] P. Lapedes, E. Sharlin, M. C. Sousa, and L. Streit. 2006. The 3D Tractus: A Three-Dimensional Drawing Board. In *First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)*. IEEE, USA, 8 pp.–. <https://doi.org/10.1109/TABLETOP.2006.33>
- [54] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–17.
- [55] David Ledo, Jo Vermeulen, Sheelagh Carpendale, Saul Greenberg, Lora Oehlberg, and Sebastian Boring. 2019. Astral: Prototyping Mobile and Smart Object Interactive Behaviours Using Familiar Applications. In *Proceedings of the 2019 on Designing Interactive Systems Conference (DIS '19)*. Association for Computing Machinery, New York, NY, USA, 711–724. <https://doi.org/10.1145/3322276.3322329>
- [56] Bokyoung Lee, Michael Lee, Pan Zhang, Alexander Tessier, Daniel Saakes, and Azam Khan. 2021. Socio-Spatial Comfort: Using Vision-Based Analysis to Inform User-Centred Human-Building Interactions. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW3 (Jan. 2021), 238:1–238:33. <https://doi.org/10.1145/3432937>
- [57] Hyun-Jean Lee, Madhur Khandelwal, and Ali Mazalek. 2007. Tilting Table: A Movable Screen. In *Proceedings of the 1st International Conference on Tangible*

- and Embedded Interaction (TEI '07). Association for Computing Machinery, New York, NY, USA, 93–96. <https://doi.org/10.1145/1226969.1226988>
- [58] Jakob Leitner, James Powell, Peter Brandl, Thomas Seifried, Michael Haller, Bernard Dorray, and Paul To. 2009. Flux: A Tilting Multi-Touch and Pen Based Surface. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09)*. ACM, New York, NY, USA, 3211–3216. <https://doi.org/10.1145/1520340.1520459>
- [59] Ming Li and Leif Kobbelt. 2012. Dynamic Tiling Display: Building an Interactive Display Surface Using Multiple Mobile Devices. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (MUM '12)*. ACM, New York, NY, USA, 24:1–24:4. <https://doi.org/10.1145/2406367.2406397>
- [60] Andrés Lucero, Jussi Holopainen, and Tero Jokela. 2011. Pass-Them-around: Collaborative Use of Mobile Phones for Photo Sharing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 1787–1796. <https://doi.org/10.1145/1978942.1979201>
- [61] Andrés Lucero, Matt Jones, Tero Jokela, and Simon Robinson. 2013. Mobile Collocated Interactions: Taking an Offline Break Together. *interactions* 20, 2 (March 2013), 26–32. <https://doi.org/10.1145/2427076.2427083>
- [62] Paul Luff and Christian Heath. 1998. Mobility in Collaboration. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW '98)*. Association for Computing Machinery, New York, NY, USA, 305–314. <https://doi.org/10.1145/289444.289505>
- [63] Nicolai Marquardt, Till Ballendat, Sebastian Boring, Saul Greenberg, and Ken Hinckley. 2012. Gradual Engagement: Facilitating Information Exchange between Digital Devices as a Function of Proximity. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. Association for Computing Machinery, Cambridge, Massachusetts, USA, 31–40. <https://doi.org/10.1145/2396636.2396642>
- [64] Nicolai Marquardt, Frederik Brudy, Can Liu, Benedikt Bengler, and Christian Holz. 2018. SurfaceConstellations: A Modular Hardware Platform for Ad-Hoc Reconfigurable Cross-Device Workspaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3173574.3173928>
- [65] Nicolai Marquardt, Robert Diaz-Marino, Sebastian Boring, and Saul Greenberg. 2011. The Proximity Toolkit: Prototyping Proxemic Interactions in Ubiquitous Computing Ecologies. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. Association for Computing Machinery, New York, NY, USA, 315–326. <https://doi.org/10.1145/2047196.2047238>
- [66] Nicolai Marquardt, Ken Hinckley, and Saul Greenberg. 2012. Cross-Device Interaction via Micro-Mobility and F-Formations. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 13–22. <https://doi.org/10.1145/2380116.2380121>
- [67] Paul Marshall, Yvonne Rogers, and Nadia Pantidi. 2011. Using F-Formations to Analyse Spatial Patterns of Interaction in Physical Environments. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work (CSCW '11)*. Association for Computing Machinery, New York, NY, USA, 445–454. <https://doi.org/10.1145/1958824.1958893>
- [68] David Merrill, Jeevan Kalanithi, and Pattie Maes. 2007. Siftables: Towards Sensor Network User Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, New York, NY, USA, 75–78. <https://doi.org/10.1145/1226969.1226984>
- [69] C. Muller-Tomfelde, A. Wessels, and C. Schremmer. 2008. Tilted Tabletops: In between Horizontal and Vertical Workspaces. In *2008 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*. IEEE, USA, 49–56. <https://doi.org/10.1109/TABLETOP.2008.4660183>
- [70] Michael Nebeling. 2017. XDBrowser 2.0: Semi-Automatic Generation of Cross-Device Interfaces. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4574–4584. <https://doi.org/10.1145/3025453.3025547>
- [71] Michael Nebeling, Maria Husmann, Christoph Zimmerli, Giulio Valente, and Moira C. Norrie. 2015. XDSession: Integrated Development and Testing of Cross-Device Applications. In *Proceedings of the 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '15)*. ACM, New York, NY, USA, 22–27. <https://doi.org/10.1145/2774225.2775075>
- [72] Michael Ortega and Alix Goguy. 2019. BEXHI: A Mechanical Structure for Prototyping Bendable and EXpandable Handheld Interfaces. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces (ISS '19)*. Association for Computing Machinery, New York, NY, USA, 269–273. <https://doi.org/10.1145/3343055.3359703>
- [73] Florian Perteneder, Kathrin Probst, Joanne Leong, Sebastian Gassler, Christian Rendl, Patrick Parzer, Katharina Fluch, Sophie Gahleitner, Sean Follmer, Hideki Koike, and Michael Haller. 2020. Foxels: Build Your Own Smart Furniture. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '20)*. Association for Computing Machinery, New York, NY, USA, 111–122. <https://doi.org/10.1145/3374920.3374935>
- [74] Roman Rädle, Hans-Christian Jetter, Jonathan Fischer, Inti Gabriel, Clemens N. Klokose, Harald Reiterer, and Christian Holz. 2018. PolarTrack: Optical Outside-In Device Tracking That Exploits Display Polarization. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 497:1–497:9. <https://doi.org/10.1145/3173574.3174071>
- [75] Roman Rädle, Hans-Christian Jetter, Nicolai Marquardt, Harald Reiterer, and Yvonne Rogers. 2014. HuddleLamp: Spatially-Aware Mobile Displays for Ad-Hoc Around-the-Table Collaboration. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14)*. ACM, New York, NY, USA, 45–54. <https://doi.org/10.1145/2669485.2669500>
- [76] 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, New York, NY, USA, 31–39. <https://doi.org/10.1145/263407.263505>
- [77] Jun Rekimoto, Yuji Ayatsuka, Michimune Kohno, and Haruo Oba. 2003. Proximal Interactions: A Direct Manipulation Technique for Wireless Networking. In *Proceedings of the IFIP International Conference on Human-Computer Interaction (INTERACT '03)*. IFIP, Zürich, Switzerland, 511–518.
- [78] Tom Rodden, Yvonne Rogers, John Halloran, and Ian Taylor. 2003. Designing Novel Interactional Workspaces to Support Face to Face Consultations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. Association for Computing Machinery, New York, NY, USA, 57–64. <https://doi.org/10.1145/642611.642623>
- [79] Hugo Romat, Christopher Collins, Nathalie Henry Riche, Michel Pahud, Christian Holz, Adam Riddle, Bill Buxton, and Ken Hinckley. 2020. Tilt-Responsive Techniques for Digital Drawing Boards. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 500–515. <https://doi.org/10.1145/3379337.3415861>
- [80] Gazelle Saniee-Monfared, Kevin Fan, Qiang Xu, Sachi Mizobuchi, Lewis Zhou, Pourang Polad Irani, and Wei Li. 2020. Tent Mode Interactions: Exploring Collocated Multi-User Interaction on a Foldable Device. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3379503.3403566>
- [81] Stephanie Santosa and Daniel Wigdor. 2013. A Field Study of Multi-Device Workflows in Distributed Workspaces. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '13)*. ACM, New York, NY, USA, 63–72. <https://doi.org/10.1145/2493432.2493476>
- [82] Julia Schwarz, David Klionsky, Chris Harrison, Paul Dietz, and Andrew Wilson. 2012. Phone As a Pixel: Enabling Ad-Hoc, Large-Scale Displays Using Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2235–2238. <https://doi.org/10.1145/2207676.2208378>
- [83] Julian Seifert, Sebastian Boring, Christian Winkler, Florian Schaub, Fabian Schwab, Steffen Herrdum, Fabian Maier, Daniel Mayer, and Enrico Rukzio. 2014. Hover Pad: Interacting with Autonomous and Self-Actuated Displays in Space. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. Association for Computing Machinery, New York, NY, USA, 139–147. <https://doi.org/10.1145/2642918.2647385>
- [84] Abigail Sellen, Bill Buxton, and John Arnott. 1992. Using Spatial Cues to Improve Videoconferencing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. Association for Computing Machinery, New York, NY, USA, 651–652. <https://doi.org/10.1145/142750.143070>
- [85] Teddy Seyed, Mario Costa Sousa, Frank Maurer, and Anthony Tang. 2013. SkyHunter: A Multi-Surface Environment for Supporting Oil and Gas Exploration. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces (ITS '13)*. ACM, New York, NY, USA, 15–22. <https://doi.org/10.1145/2512349.2512798>
- [86] Mike Sinclair, Michel Pahud, and Hrvoje Benko. 2013. TouchMover: Actuated 3D Touchscreen with Haptic Feedback. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces (ITS '13)*. Association for Computing Machinery, New York, NY, USA, 287–296. <https://doi.org/10.1145/2512349.2512805>
- [87] Norbert A. Streitz, Jörg Geißler, Torsten Holmer, Shin'ichi Konomi, Christian Müller-Tomfelde, Wolfgang Reischl, Petra Rexroth, Peter Seitz, and Ralf Steinmetz. 1999. I-LAND: An Interactive Landscape for Creativity and Innovation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, New York, NY, USA, 120–127. <https://doi.org/10.1145/302979.303010>
- [88] Peter Tandler, Thorsten Prante, Christian Müller-Tomfelde, Norbert Streitz, and Ralf Steinmetz. 2001. Connectables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*. ACM, New York, NY, USA, 11–20. <https://doi.org/10.1145/502348.502351>
- [89] Anthony Tang, Michel Pahud, Kori Inkpen, Hrvoje Benko, John C. Tang, and Bill Buxton. 2010. Three's Company: Understanding Communication Channels in Three-Way Distributed Collaboration. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*. Association for Computing Machinery, New York, NY, USA, 271–280. <https://doi.org/10.1145/1718918>



- 1718969
- [90] Jaime Teevan, Brent Hecht, Sonia Jaffe, Nancy Baym, Rachel Bergmann, Matt Brodsky, Bill Buxton, Jenna Butler, Adam Coleman, Mary Czerwinski, Brian Houck, Ginger Hudson, Shamsi Iqbal, Chandra Maddila, Kate Nowak, Emily Peloquin, Ricardo Reyna Fernandez, Sean Rintel, Abigail Sellen, Tiffany Smith, Margaret-Anne Storey, Siddharth Suri, Hana Wolf, and Longqi Yang. 2021. *The New Future of Work: Research from Microsoft into the Pandemic's Impact on Work Practices*. Technical Report MSR-TR-2021-1. Microsoft.
- [91] Maurice Ten Koppel, Gilles Bailly, Jörg Müller, and Robert Walter. 2012. Chained Displays: Configurations of Public Displays Can Be Used to Influence Actor-, Audience-, and Passer-by Behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 317–326. <https://doi.org/10.1145/2207676.2207720>
- [92] John Tiab, Sebastian Boring, Paul Strohmeier, Anders Markussen, Jason Alexander, and Kasper Hornbæk. 2018. Tiltstacks: Composing Shape-Changing Interfaces Using Tilting and Stacking of Modules. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces (AVI '18)*. Association for Computing Machinery, New York, NY, USA, 1–5. <https://doi.org/10.1145/3206505.3206530>
- [93] Michael Tsang, George W. Fitzmaurice, Gordon Kurtenbach, Azam Khan, and Bill Buxton. 2002. Boom Chameleon: Simultaneous Capture of 3D Viewpoint, Voice and Gesture Annotations on a Spatially-Aware Display. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST '02)*. Association for Computing Machinery, New York, NY, USA, 111–120. <https://doi.org/10.1145/571985.572001>
- [94] Michael Tsang, George W. Fitzmaurice, Gordon Kurtenbach, Azam Khan, and Bill Buxton. 2003. Boom Chameleon: Simultaneous Capture of 3D Viewpoint, Voice and Gesture Annotations on a Spatially-Aware Display. *ACM Transactions on Graphics* 22, 3 (July 2003), 698. <https://doi.org/10.1145/882262.882329>
- [95] UGreen. 2021. Tablet Holder, Adjustable Flexible Desk Mount (Manufactured by UGreen). <https://www.ugreen.com/products/lazy-arm-tablet-holder>.
- [96] Brygg Ullmer and Hiroshi Ishii. 1997. The metaDESK: Models and Prototypes for Tangible User Interfaces. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology (UIST '97)*. Association for Computing Machinery, New York, NY, USA, 223–232. <https://doi.org/10.1145/263407.263551>
- [97] Jo Vermeulen, Kris Luyten, Karin Coninx, and Nicolai Marquardt. 2014. The Design of Slow-Motion Feedback. In *Proceedings of the 2014 Conference on Designing Interactive Systems (DIS '14)*. Association for Computing Machinery, Vancouver, BC, Canada, 267–270. <https://doi.org/10.1145/2598510.2598604>
- [98] Jo Vermeulen, Kris Luyten, Elise van den Hoven, and Karin Coninx. 2013. Crossing the Bridge over Norman's Gulf of Execution: Revealing Feedforward's True Identity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1931–1940.
- [99] Simon Voelker, Sebastian Hueber, Christian Holz, Christian Remy, and Nicolai Marquardt. 2020. GazeConduits: Calibration-Free Cross-Device Collaboration through Gaze and Touch. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–10. <https://doi.org/10.1145/3313831.3376578>
- [100] Simon Voelker, Malte Weiss, Chat Wacharamanotham, and Jan Borchers. 2011. Dynamic Portals: A Lightweight Metaphor for Fast Object Transfer on Interactive Surfaces. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. Association for Computing Machinery, New York, NY, USA, 158–161. <https://doi.org/10.1145/2076354.2076384>
- [101] Malte Weiss, Simon Voelker, Christine Sutter, and Jan Borchers. 2010. BendDesk: Dragging Across the Curve. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, New York, NY, USA, 1–10. <https://doi.org/10.1145/1936652.1936654>
- [102] Pierre Wellner. 1993. Interacting with Paper on the DigitalDesk. *Commun. ACM* 36, 7 (July 1993), 87–96. <https://doi.org/10.1145/159544.159630>
- [103] Daniel Wigdor, Hao Jiang, Clifton Forlines, Michelle Borkin, and Chia Shen. 2009. WeSpace: The Design Development and Deployment of a Walk-up and Share Multi-Surface Visual Collaboration System. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1237–1246. <https://doi.org/10.1145/1518701.1518886>
- [104] 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 (UIST '10)*. Association for Computing Machinery, New York, NY, USA, 273–282. <https://doi.org/10.1145/1866029.1866073>
- [105] Raphael Wimmer, Fabian Hennecke, Florian Schulz, Sebastian Boring, Andreas Butz, and Heinrich Hußmann. 2010. Curve: Revisiting the Digital Desk. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordCHI '10)*. ACM, New York, NY, USA, 561–570. <https://doi.org/10.1145/1868914.1868977>
- [106] Pawel Wozniak, Nitesh Goyal, Przemyslaw Kucharski, Lars Lischke, Sven Mayer, and Morten Fjeld. 2016. RAMPARTS: Supporting Sensemaking with Spatially-Aware Mobile Interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 2447–2460. <https://doi.org/10.1145/2858036.2858491>
- [107] Pawel Woźniak, Lars Lischke, Benjamin Schmidt, Shengdong Zhao, and Morten Fjeld. 2014. Thaddeus: A Dual Device Interaction Space for Exploring Information Visualisation. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordCHI '14)*. ACM, New York, NY, USA, 41–50. <https://doi.org/10.1145/2639189.2639237>
- [108] Jishuo Yang and Daniel Wigdor. 2014. Panelrama: Enabling Easy Specification of Cross-Device Web Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2783–2792. <https://doi.org/10.1145/2556288.2557199>