Gluey: Developing a Head-Worn Display Interface to Unify the Interaction Experience in Distributed Display Environments

Marcos Serrano[1], Barrett Ens[2], Xing-Dong Yang[3], Pourang Irani[2]

[1] IRIT-University of Toulouse
Toulouse, France, 31062
marcos.serrano@irit.fr

[2] University of Manitoba
Winnipeg, MB, Canada, R3T 2N2
{bens, irani}@cs.umanitoba.ca

[3] Dartmouth College
Hanover, NH, USA
xing-dong.yang@dartmouth.edu

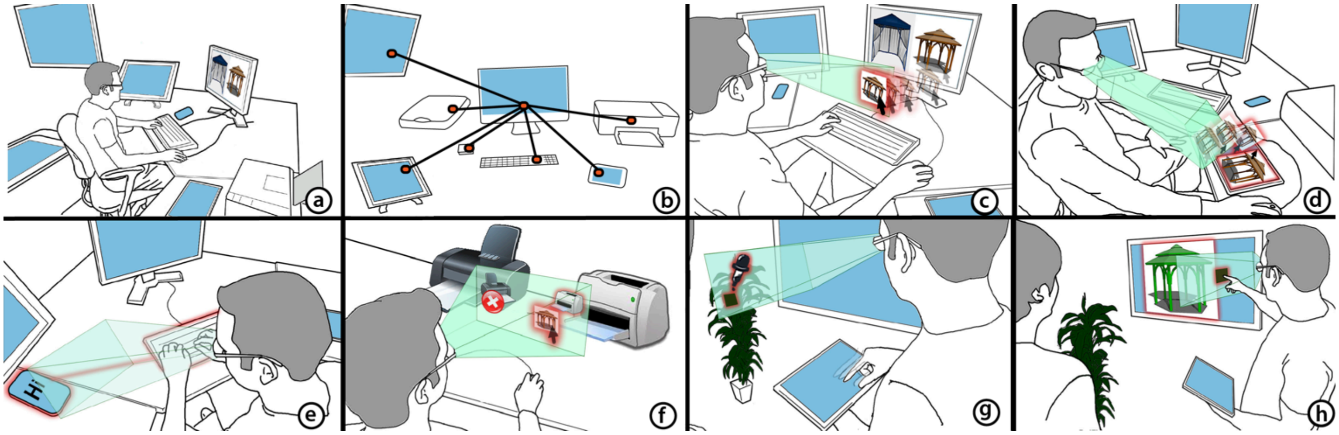

Figure 1. In this scenario, an architect uses Gluey in his daily work routine (a). He first registers devices and creates a spatial map using Gluey (b). He copies a sketch from his desktop screen to Gluey's always-available display (c), and pastes it to a tablet (d). With Gluey's field-of-view tracking feature, he can redirect keyboard input to his phone (e) or select a printer (f). Gluey's unified input capability allows him to pick a real-world color with a touchscreen (g) and apply it to his sketch using a touchscreen (e).

ABSTRACT

Distributed display environments (DDEs) allow use of various specialized devices but challenge designers to provide a clean flow of data across multiple displays. Upcoming consumer-ready head-worn displays (HWDs) can play a central role in unifying the interaction experience in such ecosystems. In this paper, we report on the design and development of Gluey, a user interface that acts as a 'glue' to facilitate seamless input transitions and data movement across displays. Based on requirements we refine for such an interface, Gluey leverages inherent head-worn display attributes such as field-of-view tracking and an always-available canvas to redirect input and migrate content across multiple displays, while minimizing device switching costs. We implemented a functional prototype integrating Gluey's numerous interaction possibilities. From our experience in this integration and from user evaluation results, we identify the open challenges in using HWDs to unify the interaction experience in DDEs.

The definitive version will be published in *MobileHCI '15,* August 25 - 28, 2015, Copenhagen, Denmark

Author Keywords

Head-Worn Display; Distributed displays; Multi-display environments; Input redirection; Content migration.

ACM Classification Keywords

H.5.2. Information interfaces and presentation: Interaction

INTRODUCTION

A new generation of light-weight, see-through head-worn displays (hereafter referred to as 'HWDs') is emerging for general purpose use (e.g. Google Glass, Meta, Microsoft HoloLens). These wearable devices will soon co-exist within a larger ecosystem of displays (i.e. desktop monitors, tablets, smartphones) we depend on for daily tasks. Researchers have sought ways to unite multiple displays and input devices into distributed display environments (DDEs) [13,24,30,37]. However, DDEs are commonly afflicted by device switching [11,47] and data transfer costs [37], which can be mitigated by inventive techniques for content migration (e.g. [32,38,43]) and input redirection (e.g. [25,34]). In such display-saturated ecosystems, what unique role can HWDs play beyond simply being an additional display medium?

To this end we implement Gluey, a user interface that acts as a 'glue' to facilitate seamless information flow and input redirection across multiple devices (Figure 1). Gluey derives its bonding force from several features unique to HWDs. For example, the embedded cameras and spatial sensors of next generation HWDs allow Gluey to maintain a spatial model of the relative positions of various devices, which can be used to facilitate cross-device interactions [48]. Gluey's spatial model can track which devices the user is looking at, and use this information to move content across devices. The handless operability of HWDs, allows users to freely interact with other displays using any available input device. Finally, HWDs' transparent displays can 'disappear' and reappear only when needed with implicit sensing of the user's actions.

Our work builds on previous research that envisioned HWDs taking on a pivotal role in DDEs [10,21]. While each of these early systems made valuable contributions, we contribute a holistic set of design requirements to drive forward the use of HWDs in DDEs with a complete system implementation. In this paper we report on the lessons learned during our design and development of Gluey on an Epson Moverio HWD. Using our proof-of-concept prototype, we implement and evaluate a set of interaction techniques for content migration and input redirection.

This work has 3 main contributions: 1) we propose a set of design requirements to leverage inherent HWD capabilities to facilitate seamless content migration and input redirection in DDEs; 2) we introduce and evaluate Gluey, an implementation of this vision; and 3) we articulate a set of open challenges in using HWDs to unify DDEs.

LEVERAGING HWD CAPABILITIES IN DDES

Gluey is inspired by Weiser's vision of 'invisible' devices [45]. In his seminal paper, he proposed to that in a highly usable, distributed display environment, the details regarding device usage should become unnoticeable and 'fade' into the background, allowing users to fully attend to their tasks. However, several workplace studies [13,30,37] reveal that, although tasks often span multiple devices [13,30], these devices are far from being 'invisible'; the work required to facilitate inter-operability occupies a significant portion of users' awareness. For instance, from taking a photo and presenting it in a finished document, data is often passed along a long chain of devices using ad-hoc methods [37]. In contrast to these reflections on the current reality, we present a scenario depicting our vision of Gluey, which addresses these issues by using HWDs' unique affordances for 'transparency'.

Gluey Scenario

John, an architect, relies on numerous digital devices while juggling between ad-hoc tasks in his daily work (Figure 1a). Most of his drawing takes place on a desktop computer attached to two monitors and a printer. John uses a tablet for sketching and for showing his drawings to clients. He uses a smartphone for his communication needs.

John completes a plan for a gazebo on his desktop computer and prepares to use his tablet for presenting it to a client. After saving his presentation, John moves it with his mouse to a clipboard on his HWD (1a), so that he can later copy it onto any other device. He then glances at his tablet and uses his mouse to grab the presentation from the HWD clipboard and transfers it to the device that is in the HWD's direct view, i.e. the tablet screen (1b). Meanwhile, his business partner James sends a text message to his smartphone, noting the urgency of their soon to begin meeting. He is able to quickly reply to James, without opening his phone's soft keyboard. He instead uses the desktop keyboard, already near his hands, to type his reply. He does this by turning to look at his phone, which is detected by Gluey's head-mounted camera. As he types, the text gets redirected to the smartphone (Figure 1c). Before leaving to meet his client, John prints a copy of the same presentation he had on the HWD clipboard by dragging its document icon to a printer icon, which comes in view when he looks in the direction of the printer (1d). At his client's office, he is led to a boardroom equipped with a large wall display for presentations. Since John has previously registered this display within Gluey's spatial configuration, he can immediately use his tablet as a trackpad to drag the presentation of his gazebo drawing from his HWD onto the large display. The client discusses different options, and suggests a different color for the gazebo, which matches the leaves of a nearby plant. John glances over to the plant, and selects its deep green color with Gluey (1e), then fills in the gazebo using a finger gesture (1f).

This scenario captures how John seamlessly interleaves many tasks in an ad hoc manner. He minimizes his need to switch input devices and can seemingly migrate content to where it is needed by glancing at devices. While the actual data transfer is happening in the 'cloud', John need only concern himself with simple, intuitive gestures for manipulating abstract representations of objects that suit his immediate needs. Furthermore, he can extract content from his physical surroundings (such as color or captured images) which he copies onto his digital display.

Design Criteria

To realize the above scenario a DDE interface would need to fulfill the following design requirements (or DRs), which we ground in both existing literature and our experience with HWDs. A fundamental feature in any DDE system is **Input Redirection** (DR1) [34,44], which allows available input devices to be used across multiple displays. Such input redirection features should facilitate **Content Migration** (DR2) across devices [8,12]. Moreover, these features should be designed to support cross-compatibility between **All Devices** (DR3) in the environment [2,37], including touch gestures and other available input devices, as well as different display types. However, implausible pairings (e.g. a mouse cursor on a printer) should be omitted to avoid confusion. Initiating input redirection and content migration quickly and painlessly requires a robust **Device Registration** (DR4) mechanism [7,36] (i.e. identifying the presence and location of a device). Implicit device registration requires maintenance of a **Spatial Model** (DR5) of the environment [34,36] (i.e. updating the spatial coordinates of registered devices captured via head-tracking, depth sensor and RGB data). However, such implicit methods are prone to false activations and solutions are needed to overcome these while minimizing the need for explicit input. All of the above features will benefit from **Always Available** (DR6) feedback [10,13] as the user's attention often shifts among the available displays. Furthermore, to enable such operations beyond a single instrumented environment, **Mobility** (DR7) is crucial [2,30]. Our Gluey design is largely dedicated toward fulfilling these design requirements for a 'transparent' and mobile, next-generation computing platform.

STATE OF THE ART

A long history of research on HWD interfaces has led to the development of advanced platforms for superimposing digital information on the real world [16,17,33]. Whereas such Augmented Reality applications are aimed toward providing the user with context-dependent information, Gluey is designed to enhance the user's interaction experience. Gluey is related to work on Device Composition [2] and draws inspiration from several areas of research on multi-display systems [5,12,20].

Input Redirection and Content Migration in DDEs

With input redirection (DR1), one input device is used to control data on multiple displays [44]. An early ubicomp concept, Mouse Anywhere [9], uses spatial tracking to direct mouse input to the display panel nearest to the user. In later techniques, multiple displays are 'stitched' together to form a large distributed workspace [25,34]. In this space, users can migrate content between separate displays using variations of the traditional dragging technique [25,34]. A later comparison of stitching techniques [29] notes that techniques traversing interstitial space face a disadvantage due to lack of visual feedback in this space. Gluey's always available display can be used to mitigate this disadvantage.

In contrast to input redirection, clipboard techniques (DR2) store a temporary instance of an object while transferring it to a new location. This metaphor has been extended for use in DDEs [27]. To make such interaction intuitive, designers can introduce a physical or virtual object to represent the intermediary storage [32]. Other techniques provide visual representation of the copied object [8], support storage of multiple objects on a tabletop [38] or allow migration of data captured from real-world objects [7].

More recently, WatchConnect [23] uses a smartwatch as a central device in DDEs. However a watch is not always available, such as when typing on a keyboard, and lacks the unique benefits of HWDs. For example, HWDs can overlay virtual information on the real world and allow for drag & drop across displays as demonstrated in an early prototype called EMMIE [10]. HWDs are by definition always available (DR6) and have been combined with head, hand and object trackers to manipulate virtual information in DDEs [21]. Gluey's always available, hands-free display is ideal for visual storage of multiple migratory objects. Using Gluey, users can easily copy and paste items in multiple interleaved operations using head motion.

Head-orientation and Gaze Based Techniques in DDEs

Head or gaze orientation can facilitate input redirection or content migration in DDEs. Zhai et al.'s MAGIC technique [48] was one of the earliest systems to exploit users' gaze to coarsely reposition a cursor. Gaze direction has also proven useful for redirecting keyboard input to a specific device at which a user is looking [14]. Initially, head-orientation based content migration techniques were designed to move objects in 3D graphical workspaces [31,34]. Turner et al. [43] also explored several gaze-based techniques for transferring content between displays. In user studies, they found a trade-off between performance and ease of use, with users preferring techniques that demand lower hand-eye coordination. Unlike gaze-tracking systems, which pinpoint the precise location of the user's view [42], Gluey uses head orientation as a proxy for gaze as gaze-modelling studies show that target switching motions can be primarily accounted for by head motion [18,26]. With head orientation Gluey can determine the display currently in use and detect switches between displays.

Device Registration and Spatial Models in DDEs

Most DDEs employ external techniques to register and locate devices, such as GPS [21], external cameras [35], or ultrasonic and magnetic sensors [10]. Smartphone cameras have been used to identify displays without help from external devices by recognizing the screen content [8,12]. For example, TouchProjector [8] allows users to drag and drop objects from remote screens through live video image and DeepShot [12] captures the work state on a device and allows that state to be resumed on a different device. Unlike Gluey however, these techniques require the user to hold the device throughout the entire process.

	DR1	**DR2**	**DR3**	**DR4**	**DR5**	**DR6**	**DR7**
	Input Redirection	Content Migration	All Devices	Device Registration	Spatial Model	Always Available	Mobility
Fix & Float [35]	×	✓	×	×	×	×	✓
Sync. Clip. [27]	×	✓	×	×	×	×	✓
Toolspace [31]	×	✓	×	×	×	×	✓
EMMIE [10]	×	✓	×	×	×	✓	✓
MARS [21]	×	×	×	×	×	✓	✓
Aug. Surface[34]	✓	✓	×	✓	✓	×	×
Hybrid UI [36]	✓	×	✓	✓	✓	✓	✓
Touch Proj. [8]	✓	✓	×	×	✓	×	✓
DeepShot [12]	×	✓	×	✓	✓	×	✓
WatchConnect [23]	✓	✓	×	×	×	×	✓
MultiFi [19]	✓	×	×	×	✓	✓	✓
Gluey	✓	✓	✓	✓	✓	✓	✓

Table 1. Prior work summary in relation to the Design Requirements (DRs) from the literature.

Head direction has been previously used to help determine the best topology for facilitating cursor movement between screen edges [28] and to automatically redirect input across displays based on gaze estimation [1,4]. HWDs have also been used to display the configuration of input devices and objects in DDEs [36] or to extend wearable device interaction capabilities [19]. Gluey builds on these concepts by building a complete spatial model of displays in the environment, but with only the use of the HWD-mounted camera. Gluey also differentiates from previous work by leveraging field-of-view (FoV) tracking from the same camera to implicitly support input redirection and content migration. Table 1 situates this prior work in relation to our holistic set of design requirements.

GLUEY HARDWARE

We implemented our prototype on an Epson Moverio BT-100 head-worn display. The display has a resolution of 960×540 pixels, a diagonal FoV of 23°, 24-bit color depth and a perceived image size of 80 inches at 5 meters. The display's tethered controller runs Android v2.2. In this initial implementation, we did not explore design issues related to binocular depth and used it in monocular mode.

We equipped the HWD with a Logitech C270 HD webcam (Figure 2 top left), chosen for its high resolution and frame rate (1280×720 at 30 fps) with a small footprint (17/45 g without/with cable, 70×18×30mm). The camera's FoV (52° diagonal) is wider than the limited field of the Moverio, allowing Gluey to track objects beyond the visualization range of the Glueboard (Figure 2 top right).

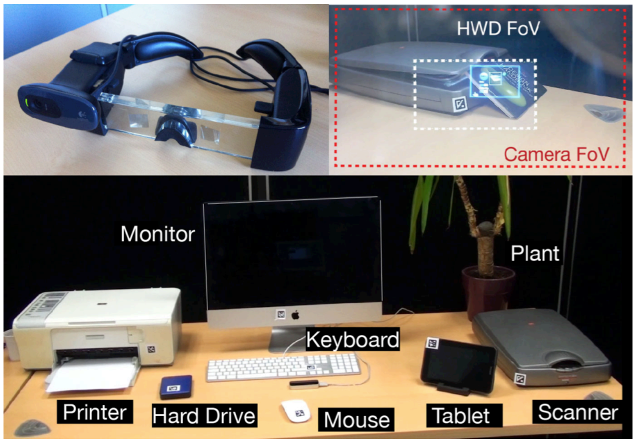

Figure 2. HWD equipped with a webcam (top left). First person view of the HWD content and comparison of HWD and

The camera is connected to the desktop computer running Mac OSX v10.7, thus all computer vision tasks are conducted on a single computer, ensuring a high frame rate. The communication between the desktop computer and the HWD was performed using TCP sockets over USB. The HWD runs the TCP server, while the computer and tablet connect as clients at an estimated frequency of 60Hz. Our implementation tested Gluey's capability in a DDE that includes several devices (Figure 2 bottom).

GLUEY COMPONENTS

Gluey incorporates several key components that support simple and intuitive techniques for unifying the interaction experience of distributed display environments. To provide cohesion between otherwise disjoint displays, we exploit the unique features of HWDs such as *view-fixed displays*, *cameras* and inclusion of *spatial sensors* (Figure 3).

We describe the key components of our system using the terms Glueboard, Glueview and Gluon as these define concepts in our implemented system but have slightly different meanings than what might traditionally be considered a 'clipboard', 'head-gaze input' and 'input redirection'. The Glueboard allows the user to have an-always visible clipboard, which allows content to move across devices, even those that might not be in the user's vicinity. The term Gluon denotes input from any source, but also goes beyond its primary purpose of facilitating input direction. For example, the Gluon, when aligned with the Glueview, can allow picking colors from physical objects in the environment. This function deviates from the traditional definition of input redirection. We thus felt new terms would define our functionality more accurately.

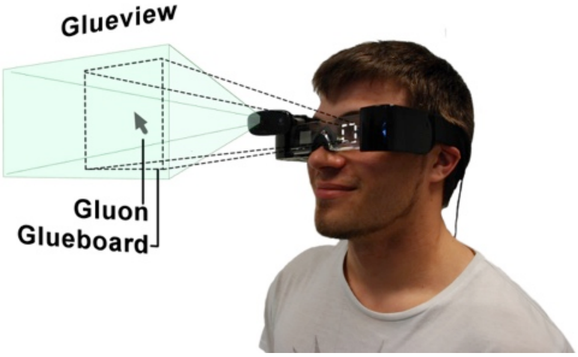

Figure 3. Gluey exploits the unique features of HWDs with three components: Glueview, Glueboard and Gluon. Together, these components enable several novel interaction techniques.

Glueview

Many HWDs now include built-in *cameras* and *spatial sensors* (e.g. Meta), from which we can determine a user's head position in relation to the environment and parse details about what they are viewing. Because head motion dominates over eye movement when switching targets (accounting for 55% of vertical and 85% of lateral motion) [26], we can gain information based on a user's general *head orientation*, for instance which device they are looking at. With head-direction, we can approximate the functionality of gaze-based systems while avoiding many of their drawbacks, such as the need for additional hardware and difficulties aligning HWD visual content with the real world [16]. We use the term *Glueview* to denote the use of FoV tracking (i.e. head orientation as a proxy for gaze tracking) in combination with Gluey's spatial model (**DR5**) for enabling implicit device registration (**DR4**): for example by simply having the smartphone in the FoV, the user can link his desktop keyboard and the smartphone.

Implementation

We register and track the position of surrounding devices with the marker-based ARToolkitPlus. Each tracked object has a unique 2×2 cm marker, allowing identification and tracking in 3D space. During an initiation stage, the Gluey user views each marker and adds it to a 3D spatial model. The tracking application runs at 60 fps; in each frame, the tracker detects all visible markers, maps them to known devices and sends both device id and position to the HWD. Finally webcam coordinates are transformed into HWD coordinates for aligning visual output. Despite the drawbacks of adding a marker to each tracked object and dependency on light conditions, we found this approach sufficient for building our proof-of-concept application in a controlled environment.

Glueboard

Rather than treat the HWD as just another information display, we can use the device's *view-fixed display* as an always-available 'canvas' for displaying *visual feedback* about interactions in the DDE. The HWD is advantageous for this purpose as the feedback is always-available (**DR6**) even when the user is mobile (**DR7**). For example, this display can show text next to a keyboard, augment devices with useful status information or provide a *visible clipboard* space to store *multiple data objects* in transit between copy/paste operations (**DR2**) to multiple interleaved destinations (**DR3**). We call this combined, always-available feedback and multi-object-clipboard space the *Glueboard*.

Implementation

During use, the Glueboard is dynamically updated with information on surrounding objects, including status of registered devices (e.g. power on, ready to print) and incoming input data. To display the Glueboard, we set the background color to black to maintain display transparency (black is not visible against the real-world background).

Gluon

A main limitation of multi-device use is the need to change input modes when switching between devices. Since HWDs do not need to be held, the Gluey user can use any input device at hand, whether it be a keyboard, mouse, a mobile touchscreen or mid-air finger gestures to control multiple displays (**DR1**, **DR2**). The *Gluon* pairs input devices with displays in the DDE to provide a unified interaction experience, independent of input mode or display type. For instance a user can use a desktop keyboard to enter text on a smartphone SMS. Other input modes can also be used to interact with any display or even the real world; for instance a user can select a physical object's color using a tablet as a trackpad. The *Gluon* can be thought of as the visible pointer for various cross-device input modes (e.g. mouse-Gluon, finger-Gluon – see Figure 7), but more so, the Gluon represents the concept of input-agnostic pairing between all available input and display types (**DR3**). This concept is primarily possible as the display on the HWD can be operated in a hands-free manner, a characteristic that is not easily possible with other types of mobile devices.

Implementation

The Gluon is implemented according to the following principles to promote usability. First, each input device is associated with only one display at any given time. Second, the Gluon is a singleton, meaning that it can only appear on one display at a time, be it the desktop monitor, the HWD or any another device, such as a tablet or smartphone. Third, the graphical representation of the Gluon pointer is different on the HWD than on other devices to give its current location and delineate transitions between overlapped displays. In our implementation, the computer sends mouse positions, keyboard input and Leap motion gestures at a frequency of 60 Hz.

GLUEY INTERACTION CAPABILITES

We illustrate Gluey's three main components through an array of interaction techniques. All features presented hereafter were implemented in our prototype.

Assorted Content Migration Techniques across Multiple Devices

Gluey facilitates at least three forms of content migration.

1) Migrating content across multiple displays: As with prior systems [10,27,32], Gluey facilitates fluid *digital content migration*, for example moving a text document or an image from one display to another. To copy multiple items in Gluey from a variety of devices, we implemented a link between the desktop computer's clipboard and our Glueboard. When the user issues a copy command on the desktop system, the object is added to a list of objects maintained by the HWD and a thumbnail showing its presence appears on the Glueboard (Figure 4). This item can be any object of the file system, such as an image, a video, a pdf, a web link or simply text. For efficiency, we also implemented a feature to allow quick text-entry onto the Glueboard, for replying to interleaving SMS bursts during a task, for example. As the user looks at the keyboard and types, Gluey stores the text in a buffer. When the user looks away, the text appears on the Glueboard for subsequent placement on any device via the Glueview.

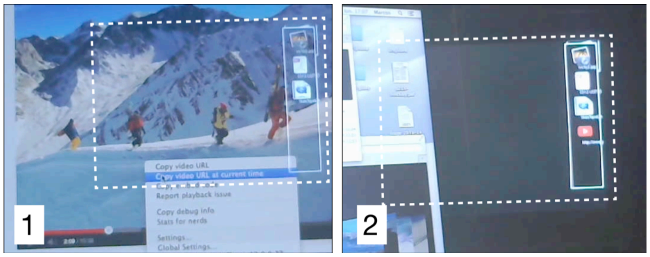

Figure 4. Copying a Youtube link (left) onto the multi-object Glueboard alongside an image, a video and a pdf file (right).

We implemented two methods for moving items from the Glueboard onto other devices. With the mouse-Gluon, the user can simply select and drag the Glueboard item onto a device. The drop region is defined by a rectangle, centered at the marker (Figure 5). The item is removed from the Glueboard once it is moved onto another device. An alternative solution takes advantage of the mid-air Gluon whereby the user can select an item on the Glueboard via finger movement (detected by our Leap device) and flick it in the direction of the device as per Gluey's spatial model. For instance, the user can look at the desktop display and make a swiping gesture to paste the item to the tablet on the right. In this implementation we cannot differentiate among multiple devices in a given direction, but this could be mapped to flick speed or rely on contextual or recency data.

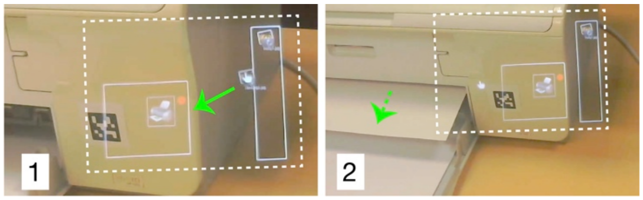

Figure 5. Printing an image via drag-and-drop from the Glueboard to an idle printer can mitigate wait queues.

2) Migrating content across remote devices: Through its mobility feature, Gluey enables *remote content migration*, for example by grabbing text from a display and 'dropping' it on an available printer or scanner in an adjacent room. This can mitigate delays incurred during interaction with remote devices, such as printers. For example, a user can walk with their HWD to a printer room and determine which devices are operational and jobless before pasting the item onto a printer of choice (Figure 5). Such remote migrations are as fluid as those used for migrating content across devices.

3) Migrating physical world features onto a display: Gluey allows capturing features of the user's immediate environment for use on any display, such as in our implemented *image capture* and *color picker* techniques. As the HWD is equipped with a camera, the user can capture an image of content visible through the Glueview. We implemented this feature using the Leap device. A clockwise circular mid-air gesture with the finger grabs the image in the Glueview (Figure 6). The image then appears on the Glueboard and can be transferred to any display as with other digital content. Similar operations are also possible with other input methods; for example, with a mouse-Gluon, the user can lasso a region under the Glueview to select an object in the environment.

Inspired by the Toolglass technique [6], we also implemented a color picker widget that allows the user to choose colors from the user's surroundings. For example, the user can look at a book or a cup and pick its color to use it in a drawing. The color is copied into an available swatch on the Glueboard where it can be later selected with the Gluon. While this approach depends on correct calibration of the HWD camera, it shows Gluey's capability to migrate content from outside the boundaries of the digital ecosystem. Other forms of capture are also possible, such as selecting text from a book, or capturing faces at a meeting.

Input Redirection across Multiple Contexts

Gluey's framework also facilitates diverse input redirection methods. To demonstrate this, we implemented four instantiations of the Gluon, each for very different contexts and applications.

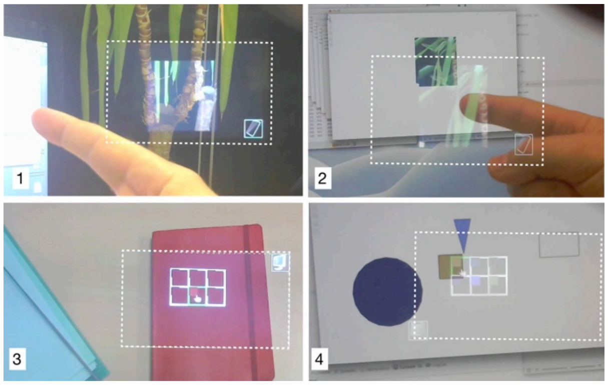

Figure 6. Top: 1) Taking a snapshot with a finger gesture; 2) pasting the snapshot on a display with a swipe forward gesture; 3) Picking colors from real-world objects. 4) Pasting the colors into graphical items on the computer display.

The *mouse-Gluon* allows the user to control any other device with the desktop mouse cursor. This is particularly effective for interleaving tasks between the desktop and other devices or for precise control or selection of objects among all devices. To transfer the mouse-gluon onto the Glueboard, the user aligns the Glueboard to be adjacent to the display on which the mouse cursor exists, such that the two appear 'stitched' together. When the mouse cursor is moved beyond the display's boundary it lands on the Glueboard (Figure 7-top). The mouse-Gluon changes appearance to indicate its current location. The *touch-Gluon* replaces the mouse input when the latter is unavailable as in mobile contexts. Using an active touch device, such as a smartphone, the user can invoke the touch-Gluon, in a similar manner as with the mouse-Gluon, to interact with content on other devices, including with content on the Glueboard (Figure 7-bottom). It is also possible to integrate an always-available shortcut, such as a bezel-tap gesture [40] to transfer control from a touchscreen to the HWD.

The *keyboard-Gluon* redirects input from an active keyboard, to any other device. This redirection is triggered by the user's glance at the target display. For instance, while editing a text document on a desktop, the user can turn her head to a tablet to automatically redirect text there, without needing to open its native text-entry mechanism.

The *midair-Gluon* is useful for creating quick ad-hoc links between devices. HWDs equipped with depth cameras can track finger motions in mid-air. We implemented two types of finger gestures, pointing and swiping, using the Leap Motion affixed to a desk. A pointing gesture selects an object that is intersected by an imaginary ray projected from the pointing finger. For instance, to print a photo, the user points to select it on the desktop and then points at the printer. Swiping is used for moving objects between devices in a given direction, corresponding to relative device locations in Gluey's spatial model.

We implemented two complementary methods to handle false positives, i.e. accidental input redirections. The first method verifies whether the Gluon is compatible with the selected device. For instance the user cannot redirect a mouse to a printer. The second method adds a brief dwell time before starting input redirection. Visual feedback on the HWD informs the user of the time left and allows her to cancel the redirection by shaking her head.

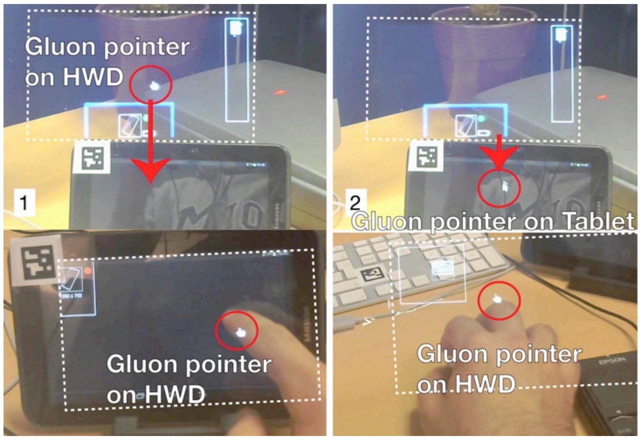

Figure 7. Mouse-Gluon transfer between stitched Glueboard and tablet display (top); touch-Gluon (bottom left); and mouse-Gluon (bottom right).

Always-Available Visual Feedback

An always-available display, such as the Glueboard can provide at least two forms of visual feedback. The Glueboard can show multiple items that have been copied for transfer to other devices. It can also show feedback about the user's Gluon usage and text-entry. In addition, Gluey can reveal hidden properties and states of devices in the DDE. For example, when the user looks at a device, she can request feedback about the desktop mouse battery life, display resolution of a device or the list of jobs on a printer.

Environment's spatial model

To assist with useful operations such as finger pointing or display stitching, Gluey uses the spatial model created during an initial device registration phase. This spatial model captures the position of each device around the user. When the user needs to recall the locations of registered devices, she can view a 2D representation of the spatial layout. The spatial model also allows interaction with out-of-view devices: the user can point at the printer on the left side of the desk without turning her head to send a document. Small icons on the border of the Glueboard indicate the position of nearby devices.

PRELIMINARY USER FEEDBACK

We gathered preliminary user feedback with Gluey to examine the potential of its interaction capabilities, and to seek ideas for improving its features and implementations.

Participants

We recruited 12 students (1 female, between 21 and 36 years of age) to give feedback on Gluey. Our participants were "naïve" to our implementation and are not actively doing research on HWDs. Only 3 participants had previous experience with the commercial Oculus Rift.

Procedure

Participants viewed a video presenting the Gluey prototype, its main components, and interaction capabilities. After showing all interactions participants used the Gluey prototype for roughly 5 minutes for each of the following techniques: device registration; input redirection; content migration (across displays, printing and scanning, color picking and image picking); and out-of-view interactions with the spatial map. For example, for input redirection they were asked to redirect the mouse-Gluon to the HWD and then to a table. After each technique, the experimenter interviewed the participant for roughly 3-5 minutes and recorded his rankings for the technique on a 7-point Likert scale (1: strongly dislike and 7: strongly like). We recorded additional information for each technique: 1) what participants liked and disliked most about a feature; 2) how they would do the same action without Gluey; 3) suggestions on how the current feature could be improved; and 4) the level of distraction resulting from Gluey's information overlay. We collected the answers to the questionnaire and videotaped the entire experiment.

Results

On average the experiment took 55 minutes (15 minutes for presenting Gluey and showing the videos, and 40 minutes for testing the prototype). We report the results in terms of general usability and then discuss the impact of our implemented design requirements in Gluey.

General acceptability of a HWD interface for unifying the interaction experience

Overall, participants' rankings were highly positive. They gave mostly favorable comments on content redirection features (copy and paste on the Glueboard) and input redirection. Seamlessly taking a snapshot and pasting it later on a device was another highly appreciated interaction.

Input Redirection (DR1 & DR4)

Forms of input redirection: All forms of input redirection were ranked high (83% rated redirection favorably, with a score of 5 or higher). Keyboard and the touch redirection were the most highly favored. The keyboard was perceived as "useful to interact with a tablet or smartphone" (p4) and the touch "for using Gluey in a mobile context" (p3). The mouse was liked for its "performance" (p5) and "familiarity" (p11). Some pointed that "transferring the mouse from the PC to Gluey can be tedious" (p4) and that "it is not adapted to all devices" (e.g. tablet) (p2). Some users preferred using gestures over the mouse as they are "more natural" (p2) and there is "no need to handle an input device" (p6). A suggested improvement for gestures is to "visualize the area where gestures can be performed" and "provide continuous feedback on the gesture itself" (p8).

Handling false positives: Users appreciated our two mechanisms to handle false positives. Participants liked the visual feedback, which is "informative" (p6) and "easy to understand" (p9). A possible improvement would be to "set the timer value according to the device or context" (p8).

Content Migration (DR2 & DR4)

Copy and paste: was the most preferred interaction with Gluey. Participants rated it on average a 6.4. The benefits cited by participants include "not needing USB cables or keys" (p9), being able to "replace PCs dedicated towards data transfer" (p5) and its "immediacy" (p4). A possible perceived limitation of this approach is its scalability, i.e. how many items a user can copy (p2). Suggestions for improvement include "using gestures or voice" (p1), "being able to copy several items" and grouping them at once (p8) and "having an undo command" (p6).

Print: Most participants (75%) appreciated being able to print using Gluey. Some users pointed out that Wi-Fi printers are gaining popularity and "allow printing from any device" (p5). However, with Gluey, the user can walk up to a row of printers and identify the one that is not busy before issuing the print command. Participants who did not like this feature cited problems such as "turning the head towards the printer" (p8) or that "this task is already easy on PCs" (p6). Interestingly, one user greatly appreciated the feature, explaining that she owns a Wi-Fi printer that does not allow printing from her mobile device. This requires her to manually transfer data to a PC before printing, a problem Gluey was specifically designed to alleviate.

Physical content migration: Taking colors from the real world was liked by 83% of the participants. While some of them said color picking was "more interesting for graphic designers" (p10), others saw it as a "tool for replacing the PC color picker" (p6) or as being useful "to find a similarly colored cloth on the Internet" (p5). Furthermore, 83% of participants liked taking snapshots and scanning.

Always available visual feedback (DR6)

Visual feedback, i.e. seeing the state, position and input interaction of devices was perceived to be useful by 83% of the participants. Participants liked "seeing the state of devices, such as battery status" (p3), "memory left" (p4), "connection" (p4) or "shortcuts while typing" (p5). Most participants also pointed out that such feedback should be disabled at will. One participant suggested showing only a small icon that can be expanded when needed (p2).

Gluey visual distraction: from Glueview is considered minimal. Most users (75%) considered the visual distraction acceptable. Some indicated that they ignored distractions after getting used to Gluey (p5). Participants that rated Gluey as distracting cited mobile contexts (p3), driving (p7) or looking at another display (p6) as situations where they would not want to have a visual overlay. As a solution, one participant suggested providing a simple input to turn off the GlueView at will, such as a tap on the HWD temple.

Mobility (DR7)

Gluey's inherent mobility was perceived to be a useful feature. Interestingly, while some participants cited mobility to be useful for work, others favoured its benefits at home. At work Gluey allows "sharing code with a colleague" (p2) or "taking a PowerPoint presentation to a meeting" (p5). At home you can "share content among the different devices in different rooms" (p9), "copy a post-it from my fridge and paste it on my computer" (p10) or "replace my Google Chromecast" (p6).

Spatial Model (DR5)

Being able to use the spatial model to select an out-of-view device via pointing was identified as a very useful feature. It allows users to "keep the current visual context" and is "more effective" than repeatedly turning the head (p10). A possible improvement is "to add a selection sound to each device" (p8) for improved feedback.

Summary

Overall, participants were enthused at the concept of Gluey. These results are encouraging taking into account the current limitations of the prototype, such as its weight or the limited display FoV. This exploratory study allowed participants to envision Gluey's utility and to generate significant feedback for future improvements.

CHALLENGES IN UNIFYING DDES WITH HWDS

We further discuss our observations on the challenges that need to be addressed for future improvements on Gluey.

*Transition from Prior Multi-Display Use to Gluey***.** We probed our participants to describe their routine methods for redirecting information across devices and for minimizing device switching costs. Most users indicated they have adopted a mix of USB memory sticks, email or a cloud service to move data across devices, even if the migration is for a temporary file, i.e. moving a document to the system that is connected to the printer. These ad-hoc solutions are common among multi-device users [37]. Users viewed Gluey as addressing many of their data migration challenges and potentially making it more efficient. An effective HWD-based system should take into account these prior uses to facilitate the transition.

Unwanted Activations. Users commented favorably on having the head direction trigger the active display. To avoid unwanted activations, we implemented a false positive mechanism that only redirects to a device after it has been fixed in for a certain period of time, giving time for cancelation. However this solution adds a latency, which can affect the overall performance of the system.

Visual Overload. Users highly favored the Glueboard for storing multiple items during migration. However, the Glueboard's visual feedback mechanism needs to also take into account the possibility for visual overload, although users did not find the Glueboard overly distracting. Gluey should provide users with explicit control over what gets placed on the always-available display, such as preview devices states, or the list of jobs on a printer.

Data Fragmentation. Gluey does not fully solve the issue of data fragmentation, although it does provide an intuitive experience for migrating content. Gluey allows transferring files to others' devices and for asynchronous, opportunistic and unplanned copy and paste. Study participants welcomed the idea of seeing a specific document at work get copied on the HWD and taken home for further use. We plan to link Gluey with existing cloud services and study how to integrate it with such approaches.

Connectivity and Deployment. For our proof-of-concept system we relied on a simple yet effective TCP/IP stack over Wi-Fi to interconnect devices with the HWD. One value of this star-formation is that any device can communicate via another through the HWD. This can provide a highly modular method of adding new devices to a user's personal display ecosystem. While such an approach sufficed in our exploration, we intend on exploring the effectiveness of new communications and interconnectivity protocols designed for DDEs [**22**].

Future Gluey versions can be deployed in at least two ways. First, Gluey can behave as a distributed application, similar to current cloud applications such as Dropbox. In this manner it will allow for content migration and simple input redirection (keyboard, gestures-based commands). Alternatively, it can exist as a module of the OS. In this manner it may facilitate a tighter integration of the various Gluon inputs. For instance, Gluey as a module of Android OS could convert input events from non-tactile devices into touchscreen events at the system level.

Other Technical Challenges. Other technical considerations that need to be addressed for a final implementation of Gluey are adapting data to the device capabilities (or informing the user if data cannot be used on a specific device) and handling network hiccups (which can affect Gluey's fluid experience).

LIMITATIONS AND FUTURE WORK

Our current prototype is limited by the marker-based tracking mechanism, the weight of the device and the resolution of the head-worn display. Our prototype is not fully mobile as the head-mounted camera is powered by our computer. We plan to replace the current webcam with a depth camera to enable tracking gestures [39,46]. System latency also needs to be substantially reduced to provide a smooth and pleasant interaction experience. However, these limitations did not impede our ability to implement and receive user feedback of our proof-of-concept prototype. Also, the limitations of HWD hardware, such as FoV, will improve with newer commercial products. We plan to further improve the mobility of our system and identify novel interaction scenarios for mobile DDEs [11] as we integrate Gluey to operate with other devices, such as smart watches and tabletops. Another concern for transparent displays is the color blending effects for which we plan to implement in Gluey recently developed color correction methods [41].

CONCLUSION

We introduced Gluey, a head-worn software interface that acts as a 'glue' for bonding information across the various displays in a distributed environment. Gluey exploits three key components: Glueview (i.e. using users' *head-direction as a proxy* to detect which device they are looking at), GlueBoard (i.e. using the device's *view-fixed display* as an always-available 'canvas' for displaying visual feedback about interactions) and Gluon (i.e. using the input device at hand seamlessly between displays). We created a proof-of-concept prototype using a HWD equipped with a webcam, guided by a number of design requirements. This allowed us to implement a large set of dedicated interaction techniques that illustrate Gluey's potential, which users commented on favorably through preliminary usage feedback. As the appearance of improved HWDs continues to accelerate, Gluey can be fully realized to resolve many of the device switching and data transfer costs in DDEs.

ACKNOWLEDGEMENTS

We thank all our participants for the many valuable suggestions and feedback on this project as well as our lab members for early thoughts and many discussions on iterations of this work. We also thank our reviewers who provided many insightful comments that helped improve the quality of the paper. Finally, we acknowledge an NSERC Discovery Grant awarded to the last author.

REFERENCES

1. Ashdown, M., Oka, K., and Sato, Y. Combining head tracking and mouse input for a GUI on multiple monitors. *CHI '05 EA*, 1188-1191.
2. Bardram, J.E. Activity-based computing: Support for mobility and collaboration in ubiquitous computing. *Journal of Personal and Ubiquitous Computing* 9(5), 312-322, 2005.
3. Baur, D., Boring, S., and Feiner, S. Virtual projection: exploring optical projection as a metaphor for multi-device interaction. *CHI '12*, 1693-1702.
4. Benko, H. & Feiner, S. Multi-Monitor Mouse. *CHI '05*, 1208-1211.
5. Bi, X. and Balakrishnan, R. Comparing usage of a large high-resolution display to single or dual desktop displays for daily work. *CHI '09*, 1005-1014.
6. Bier, E., Stone, Maureen., Pier, K., Buxton, W., and DeRose, T. Toolglass and magic lenses: the see-through interface. *SIGGRAPH '93*, 73-80.
7. Boring, S., Altendorfer, M., Broll, G., Hilliges, O. and Butz, A. Shoot & Copy: Phonecam-based information transfer from public displays onto mobile phones. *Mobility '07*, 24-31.

8. Boring, S., Baur, D., Butz, A., Gustafson, S., and Baudisch, P. 2010. Touch projector: mobile interaction through video. *CHI '10*, 2287-2296.

9. Brumitt, B., Meyers, M., Krumm, J., Kern, A. and Shafer, S. EasyLiving: Teechnologies for intelligent environments. *HUC '00*, 12-29.

10. Butz, A., Höllerer, T., Feiner, S., MacIntyre, B., and Beshers, C. 1999. Enveloping Users and Computers in a Collaborative 3D Augmented Reality. IWAR. IEEE, 35.

11. Cauchard, J., Löchtefeld, M., Irani, P., Schoening, J. Krüger, A., Fraser, M. and Subramanian, S. Visual separation in mobile multi-display environments. *UIST '11*, 451-460.

12. Chang, T.-H. and Li, Y. Deep shot: a framework for migrating tasks across devices using mobile phone cameras. *CHI '11*, 2163-2172.

13. Dearman, D. and Pierce, J. It's on my other computer!: computing with multiple devices. *CHI '08*, 767-776.

14. Dickie, C., Hart, J., Vertegaal, R., and Eiser, A. LookPoint: an evaluation of eye input for hands-free switching of input devices between multiple computers. *OZCHI '06*, 119-126.

15. Ens, B., Finnegan, R. and Irani, P. The Personal Cockpit: A spatial interface for effective task switching on head-worn displays. *CHI '14*, 3171-3180.

16. Feiner, S., Macintyre, B. and Seligmann, D. Knowledge-based augmented reality. *Communications of the ACM*, 53-62, 1993.

17. Fitzmaurice, G.W. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 39-49, 1993.

18. Freedman, E. G., and Sparks, D. L. Coordination of the eyes and head: movement kinematics. *Exp. Brain Res.* 131(1), 22–32, 2000.

19. Grubert,J., Heinisch, M., Quigley, A. and Schmalstieg, D. MultiFi: Multi-Fidelity Interaction with Displays On and Around the Body. CHI'15. ACM, to appear.

20. Grudin, J. Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. *CHI '01*, 458-465.

21. Höllerer, T., Feiner, S., Terauchi, T., Rashid, G. Hallaway, D. 1999. Exploring MARS: Developing Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System. In Computers and Graphics. Vol 23, 779-785.

22. Houben, S., Nielsen, S., Esbensen, M, and Bardram, J.E. NooSphere: An activity-centric infrastructure for distributed interaction. *MUM '13.*

23. Houben, S. and Marquardt, N. WatchConnect: A Toolkit for Prototyping Smartwatch-Centric Cross-Device Applications. CHI'15. ACM, to appear.

24. Hutchings, D.R., Stasko, J. and Czerwinski, M. Distributed display environments. *Interactions,* 12(6), 50-53, 2005.

25. Johanson, B., Hutchins, G., Winograd, T., and Stone, M. PointRight: experience with flexible input redirection in interactive workspaces. *UIST '02*, 227-234.

26. Kim, K.-H., Reed, M.P. and Martin, B.J. A model of head movement contribution for gaze transitions, *Ergonomics,* 53(4), 447-457.

27. Miller, R.C. and Myers, B.A. Synchronizing clipboards of multiple computers. *UIST '99*, 65-66.

28. Nacenta, M., Sallam, S., Champoux, B., Subramanian, S., and Gutwin, C. Perspective cursor: Perspective-based interaction for multi-display environments. *CHI '06*, 289-298.

29. Nacenta, M, Mandryk, R., and Gutwin, C. Targeting across displayless space. *CHI '08*, 777-786.

30. Oulasvirta, A. and Sumari, L. Mobile kits and laptop trays: Managing multiple devices in mobile information work. *CHI '07*, 1127-1136.

31. Pierce, J., Conway, M., Dantzich, M., and Robertson, G. 1999. Toolspaces and glances: storing, accessing, and retrieving objects in 3D desktop applications. I3D '99. ACM, 163-168.

32. Rekimoto, J. Pick-and-drop: a direct manipulation technique for multiple computer environments. *UIST '97*, 31-39.

33. Rekimoto, J. and Nagao, K. The world through the computer: Computer augmented interaction with real-world environments. *UIST '95*, 29-36.

34. Rekimoto, J. and Saitoh, M. Augmented surfaces: A spatially continuous workspace for hybrid computing environments. *CHI '99*, 378-385.

35. Robertson, G. and Card, S. Fix and float: object movement by egocentric navigation. In Proc. of UIST '97, 149-150.

36. Sandor, C., Olwal, A., Bell, B., and Feiner, S. 2005. Immersive Mixed-Reality Configuration of Hybrid User Interfaces. In Proc. of ISMAR '05. IEEE, 110-113.

37. Santosa, S. and Wigdor, D. A field study of multi-device workflows in distributed workspaces. *UbiComp'13*, 63-72.

38. Schmidt, D., Sas, C. and Gellersen, H. Personal Clipboards for individual copy-and-paste on shared multi-user surfaces. *CHI '13*, 3335-3344.

39. Serrano, M., Ens, B. and Irani, P. Exploring the use of hand-to-face input for interacting with head-worn displays. *CHI '14*, 3181-3190.

40. Serrano, M., Lecolinet, E. and Guiard, Y. Bezel-Tap gestures: Quick activation of commands from sleep mode on tablets. *CHI '13*, 3027-3036.

41. Sridharan, S.K., Hincapié-Ramos, J.D., Flatla, D.R. and Irani, P. Color correction for optical see-through displays using display color profiles. *VRST'13*, 231-240.

42. Stellmach, S. and Dachselt, R. Still looking: Investigating seamless gaze-supported selection, positioning, and manipulation of distant targets. *CHI '13*, 285-294.

43. Turner, J., Alexander, J., Bulling, A., Schmidt, D. and Gellersen, H. Eye pull, eye push: Moving objects between large screens and personal devices with gaze and touch. *INTERACT '13*, 170-186.

44. Wallace, J.R., Mandryk, R.L, and Inkpen, K. Comparing content and input redirection in MDEs. *CSCW '08*,157-166.

45. Weiser, M. The computer for the 21st century. *Scientific American*, 94-104, 1991.

46. Wilson, A. and Benko, H. 2010. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. UIST '10. ACM, 273-282.

47. Yang, X-D., Mak, E., McCallum, D., Irani, P., Cao, X. and Izadi, S. 2010. LensMouse: augmenting the mouse with an interactive touch display. *CHI '10*, 2431-2440.

48. Zhai, S., Morimoto, C., and Ihde, S. Manual and gaze input cascaded (MAGIC) pointing. *CHI '99*, 246-253.