Extend, Push, Pull: Smartphone Mediated Interaction in Spatial Augmented Reality via Intuitive Mode Switching

Jeremy Hartmann
University of Waterloo
Waterloo, ON, Canada
j3hartma@uwaterloo.ca

Aakar Gupta
University of Waterloo
Waterloo, ON, Canada
aakarg@acm.org

Daniel Vogel
University of Waterloo
Waterloo, ON, Canada
dvogel@uwaterloo.ca

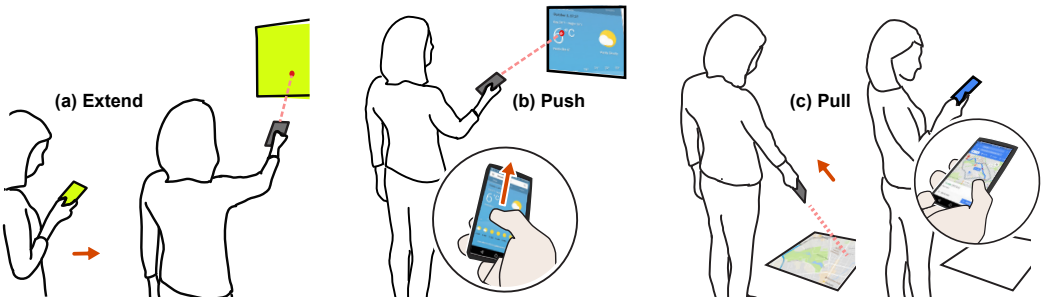

Figure 1: Illustration of the basic technique: (a) arm extension switches from mobile interaction mode to spatial interaction mode; (b) extending arm while holding finger on an application screen, then flicking up, pushes content to environment; (c) pointing at content in environment, then flicking down, pulls content to phone application for detailed manipulation.

ABSTRACT

We investigate how smartphones can be used to mediate the manipulation of smartphone-based content in spatial augmented reality (SAR). A major challenge here is in seamlessly transitioning a phone between its use as a smartphone to its use as a controller for SAR. Most users are familiar with hand extension as a way for using a remote control for SAR. We therefore propose to use hand extension as an intuitive mode switching mechanism for switching back and forth between the mobile interaction mode and the spatial interaction mode. Based on this intuitive mode switch, our technique enables the user to push smartphone content to an external SAR environment, interact with the external content, rotate-scale-translate it, and pull the content back into the smartphone, all the while ensuring no conflict between mobile interaction and spatial interaction. To ensure feasibility of hand extension as mode switch, we evaluate the classification of extended and retracted states of the smartphone based on the phone's relative 3D position with respect to the user's head while varying user postures, surface distances, and target locations. Our results show that a random forest classifier can classify the extended and retracted states with a 96% accuracy on average.

SUI '20, October 31-November 1, 2020, Virtual Event, Canada

ACM ISBN 978-1-4503-7943-4/20/10…$15.00
https://doi.org/10.1145/3385959.3418456

ACM Reference Format:
Jeremy Hartmann, Aakar Gupta, and Daniel Vogel. 2020. Extend, Push, Pull: Smartphone Mediated Interaction in Spatial Augmented Reality via Intuitive Mode Switching. In *Symposium on Spatial User Interaction (SUI '20), October 31-November 1, 2020, Virtual Event, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3385959.3418456

1 INTRODUCTION

Smartphone-based content and services are now central to many logistical and social aspects of life. However, a small phone screen still constrains how content can be viewed, manipulated, and shared in our immediate physical environment. One solution to constrained screen sizes is the use of external screens in the form of large displays or augmented reality to view and manipulate smartphone content. This work explores how smartphones can mediate this interaction in spatial augmented reality.

Current phones support television "screencasting" and its subsequent use as a 'remote control'. Researchers have also proposed methods to send phone content to large displays (e.g. [3, 14]). Several other works have proposed using the phone as a pointer for varying forms of external content around a user, including for large displays [20, 23], for head-mounted augmented reality [4, 15], or for projected spatial augmented reality (SAR) [11, 21]. However, these works do not consider the problem of how to seamlessly transition a phone between its use as a smartphone to its use as a remote control for external spatial content. When using regular smartphone operations, such as swipes, taps, rotations, to push or manipulate external spatial content, some mode switch is needed to avoid conflict between these two use cases. Although this could be accomplished with a dedicated remote control app, this kind of explicit mode switch introduces high friction when switching back

and forth between mobile interaction and spatial interaction modes multiple times within a short period.

Users routinely extend their hand out towards the display when using a remote control. In this paper, we propose to use this hand extension as a more implicit mechanism for switching back and forth between the smartphone's default usage ('mobile interaction mode') and its usage as a push and point device for external spatial content ('spatial interaction mode'): when the user extends out their hand, the smartphone switches to spatial interaction mode, and when the user retracts their hand, it switches back to mobile interaction mode. Based on this intuitive mode switch, we describe the design of our interaction technique that enables the user to push smartphone content to an external SAR environment, interact with the spatial content, rotate-scale-translate it, and pull the content back into the smartphone, all the while ensuring no conflict between the mobile interaction mode and spatial interaction use. While similar gestures have been proposed as design techniques [3], there have been limited sensing investigations that demonstrate that such an intuitive mode switch is feasible. We evaluate the classification of extended and retracted states of the smartphone based on the phone's relative 3D position with respect to the user's head while varying user postures, surface distances, and target locations. Our results show that a random forest classifier can classify the extended and retracted states with a 96% accuracy on average.

2 RELATED WORK

We divide our investigation of related work into two parts. Firstly, we look at works that use the smartphone as a pointing device for external content. Secondly, we investigate around-body interaction especially pertaining to our scenario.

2.1 Smartphone as a Pointing Device

Multiple works have explored the use of smartphones as pointing devices for controlling content on large displays, augmented reality, and spatial augmented reality. Myers et al. [20] investigated large display pointing with a laser equipped Personal Digital Assistant, which has a similar form factor to a smartphone. Their *Semantic Snarfing* technique is used for remote laser pointing, and features a method to capture remote content into the phone for detailed manipulation. *PointerPhone* [23] studied how a laser equipped smartphone could be used with a large display across six tasks, included similar capture techniques that can transfer external content to the phone's display. Beaudouin-Lafon et al. [1] investigated the use of a smartphone for interaction in a multi-display environment using unimanual and bimanual gestures. Langner et al. [14] developed a flick-transfer gesture for content sharing to a large display which is combined with a hybrid raycast and orthogonal pointing technique. However, none of these techniques address mode switch as a problem and assume that the user is using an application dedicated to interacting with the large displays. Similar to our work, *Code Space* [3] proposes arm extension as a form of an implicit mode switch for a multi-display environment to enhance the code review process.

Techniques have been proposed that combine a mobile device with an AR HMD for spatial selection [15] or for visualization of high-dimensional datasets [24]. Büschel et al. [4] used a smartphone with an AR HMD to evaluate pan and zoom techniques for 3D data spaces. They found that device movement and touch-based drag operations were most effective for unimanual interaction. On a larger scale, raycasting from a smartphone [11] or hand-held pointer [21] has been shown to be an effective and versatile approach for SAR. We use raycasting as our primary pointing method. All these techniques motivate the use of the smartphone as a pointing device for spatial content and demonstrate further the significance of enabling easy and intuitive mode switching between mobile interaction and spatial interaction modes.

2.2 Around-Body Interaction

Conceptually, the area around the body has pericutaneous, peripersonal, and extrapersonal layers [9]. Each layer describes how we view ourselves in relation to the objects situated around us, and prior work has investigated aspects of these layers to expand the set of affordances the smartphone can provide. For example, the space in front of the user has been imagined as containing hidden digital information that is viewed through the smartphones screen [29], or as a means to explore multi-layered panorama images [26].

Most relevant to our work, is using the space around the body for input. *Virtual Shelves* [17] used spatial locations positioned around the user to trigger smartphone shortcuts, and Chen et al. proposed a set of techniques that map in-air spatial locations (as well as body parts) to a set of gestures for information retrieval, storage, and actions [6]. Chen et al. conduct a preliminary study where they use the smartphone's 3D position relative to the location of the face to classify the phone's position along different distance and orientation categories [7]. The study is a preliminary study consisting of only a single user. Our work classifies the extended vs retracted state which depends on the distance and orientation of the phone relative to the user's head, while considering other influencing factors including the target location and the user posture.

In the next section, we describe the design overview of our technique that ensures conflict-free interaction for mobile and spatial modes, while ensuring other design principles including user comfort and eyes-free operation during spatial manipulation. We then describe our prototype implementation, followed by the classification analysis and usability study.

3 DESIGN OVERVIEW

The primary goal of our interaction technique is to use an arm extension as an intuitive mode switch to support both a default mobile interaction mode as well as a rich spatial interaction mode when interacting within a SAR environment. The interactions supported for the spatial mode are: push content from smartphone to SAR, delete content from SAR, RST (rotate-scale-translate) manipulation of app windows in SAR, and capture content from SAR to perform synchronized content-specific manipulation between SAR and the smartphone.

Our design is aimed at achieving the following five design goals:

Intuitive: Transitioning between a native smartphone application to spatial content should be easy to understand and discover.

Conflict-free: The method should avoid actions that conflict with existing system wide smartphone input. For example, the smartphone supports different types of touch gestures, bezel swipes, force

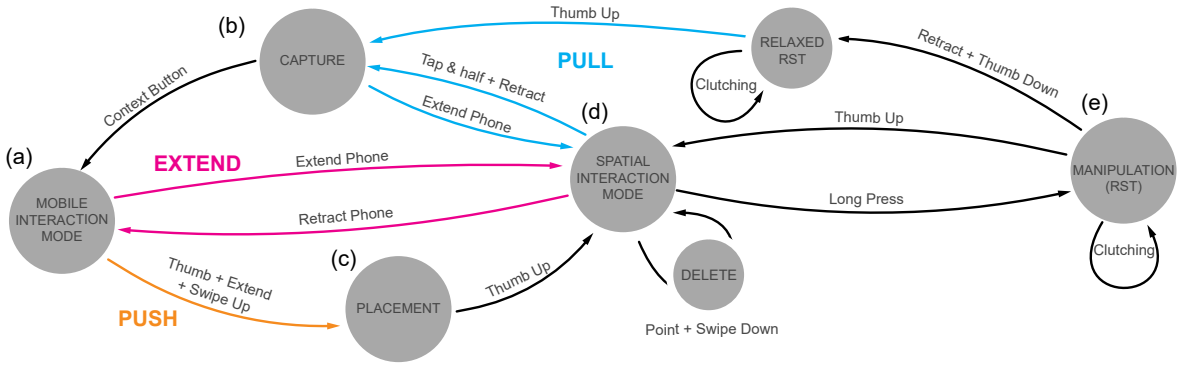

Figure 2: Interaction state diagram: (a) the native smartphone application when arm is retracted; (b) a tap-and-a-half while retracting the smartphone captures the spatial content that is in focus; (c) content is placed by holding the thumb down on an item (e.g. photo, app, etc), extending the arm, and flicking up; (d) extend arm to activate spatial interaction mode, removal of spatial content is achieved by pointing and flicking down on screen; and (e) holding the thumb on the screen while pointing enters manipulation mode where RST can be performed, retracting the arm in this state allows a relaxed posture.

presses, and overloaded physical buttons, but all of them have designated default system-level or application-level functions and cannot be used to enable fast, low-friction mode switch to another spatial mode. The arm extension and retraction enables a conflict-free mode switch while being *intuitive*.

Comfortable: However, one problem with using the phone as a remote pointer when the arm is extended is that it leads to rapid arm fatigue (gorilla arm effect). We avoid extended periods of strain in our design by enabling a relaxed RST mode where the user can perform RST operations with a retracted hand while maintaining the conflict-free use.

One-Handed Extended Use: All extended hand interactions in our design work one-handed because it is difficult to perform interactions with two extended hands.

Eyes-free Extended Use: When interacting in extended mode, the interaction should not require the user to look at the phone screen because it may not be easily visible and also because the user should be able to focus on the spatial content while manipulating it. Our design ensures this by using a combination of taps, long presses, swipes, and 3D displacement and rotation of the phone in the extended mode, all of which are eyes-free.

3.1 Interaction Technique

Figure 2 illustrates the action states and transitions in our interaction technique.

3.1.1 Extended and Retracted State (Fig. 1a, Fig. 2a,d). The user extends their hand to interact with the spatial content. The system continually uses the 3D position of the phone relative to the user's head to determine if the phone is in the extended state or retracted state. As soon as the system detects that the user has transitioned from retracted to extended, the system enters the spatial interaction mode. The extend motion naturally becomes a pointing gesture to specify a spatial location to place, remove, or manipulate content. When the user brings their arm back to the retracted state, the system switches back to the smartphone interaction mode. To enable *comfort*, the exception to this rule is when the user wants to perform relaxed RST manipulation or content-specific manipulation. While the user is in the extended state, the user can perform specific gestures to continue to interact with the spatial content in the retracted state. We detail these later.

The notion of extending the hand vs. retracting is subjective and does not depend solely on the distance or orientation of the phone. Primarily, it depends on four factors: 1) *Target Location*: the targeted spatial location of interaction. For instance, the distances when the user extends the phone towards the ground vs towards the wall vs towards the roof would be very different. 2) *User Posture*: There would be variations in how the hand is extended, depending on the user's posture, whether they are standing, sitting, or lying down. 3) *Distance of the projection surface*: The arm extension will also be impacted by the distance of the projection surface. For instance, the extension may be smaller if the wall is nearer and less than arm's length. 4) *Users*: Different users may extend the arm differently, while some may perceive extension to be a complete arm-stretch, others may opt for a slightly more relaxed version. There may be different ways users respond in the above conditions. Further the distance of the phone relative to the head may also depend on users' arm lengths. Due to these factors, it is difficult to specify a simple threshold-based classification of extended vs retracted states or use heuristic based raycasting like in Langner et al. [14]. We therefore conduct a classification study as described in the next section.

3.1.2 Placement and Removal (Fig. 2c). To distribute spatial content from the smartphone into the spatial environment, the user holds their thumb on top of the application they wish to place in the environment. With the thumb held down, they extend their hand to switch into spatial interaction mode and flick their thumb up (Fig. 1b). This can be done *one-handed*. The location and orientation of the content is dependent on where the a ray from the smartphone points just before the flick. This avoids any errors due to unintentional movements during the flick. Raycasting has been shown to have good performance for these types of tasks [11].

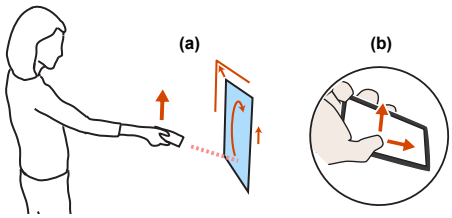

Figure 3: RST manipulation of spatial content using eyes-free touch for rotate and scale and raycasting for position.

Removal of spatial content works in a complementary way. When in spatial interaction mode and pointing at a content item, swiping down on the phone screen removes it (Fig. 1c).

3.1.3 RST Manipulation (Fig. 2e, Fig. 3). We consider four of our design principles when constructing the interactions around spatial content manipulation: *One-handed*: Any two-fingered gestures like pinching or rotating are not possible, *Intuitiveness*: RST interactions should not be hidden behind nested menus or a complicated interface, *eyes-free extended use*, and *Comfort*.

To manipulate content, the user must extend their arm and point the smartphone towards a spatial content item, then hold their thumb anywhere on the screen for dwell period of 200ms. After, the spatial content will be in selected and the user can relax their arm to a *comfortable* position.

Rotation is accomplished by moving the thumb along the x-axis of the smartphone. The rotation occurs around the contents center of mass where the rotation axis is the surface normal. Moving the thumb to the left will rotate counter-clockwise and to the right, clockwise. *Scaling* uses thumb movement along the y-axis of the smartphone. This will cause a uniform scaling of the content along all dimension, making it larger when pushing the thumb up, and smaller then pulling the thumb down. *Translation* uses raycasting, the content will automatically follow the ray and snaps to the intersecting spatial surface.

For comfort, the user can relax their arm during RST by retracting their hand while the thumb is down on the touchscreen. The system then enters the *Relaxed RST* mode and stays in it as long as the thumb does not lift from the screen for more than 1000ms (Figure 2e). This delay is needed to enable clutching for rotation and scaling. Figure 2 shows how the interaction remains *conflict-free*.

3.1.4 Capture for content-specific manipulation (Fig. 2b). Capturing spatial content into the smartphone enables more detailed manipulation, for example adjusting application-specific parameters of the content, such as a map location, or weather forecast type. This can be thought of as an extension to the content itself, an *intuitive* remote interface. To capture content, the user extends their hand, points toward the content, performs a tap-and-a-half (a tap immediately followed by a touch-down) on the screen, and then brings the phone back towards their body into a *comfortable* state. This opens a specialized application-specific interface corresponding to the spatial content. Exiting content capture uses a method *compatible* with standard operating systems: the contextual back-button or home screen gesture.

Figure 4: An example of spatial content viewed in the SAR setup. Note how content can be displayed on any surface including walls, floor, furniture, and objects.

4 IMPLEMENTATION AND APPLICATIONS

We built a proof-of-concept system to enable applications that demonstrate our interaction technique in SAR. To eliminate confounds and simplify engineering, we use a commercial motion tracking system to track the user's head and the phone. Later, we describe how this system was used first to evaluate the feasibility of the extend gesture while gathering data to build a recognizer, and second, to evaluate the usability of our interaction technique.

4.1 SAR Environment

Our environment is a corner of a large room occupying approximately 4×4 meters of floor space (Fig. 4). Placed around the environment are five digital projectors, six Microsoft Kinect cameras (each connected to an IntelNUC Intel® Core™ i7-7567U PC), and a ten-camera Vicon motion tracking system (Vera/Bonita IR cameras). An instance of the Vicon Tracker 3.6.0 software running on a dedicated server handles real time tracking of a smartphone and a person's head. The phone tracking object is a custom-printed phone case with seven 6.4mm spherical reflective markers and two 9.5mm ones. The head is tracked through a ball cap with five markers attached to the visor and crown. All tracking is filtered using the One Euro Filter [5] ($f = 9.9$ and $\beta = 0.5$ for position, $f = 20$ and $\beta = 0.5$ for orientation).

The main server (Windows 10, Intel® Core™ i7-6850K) is connected to the Vicon server and IntelNUCs using a local intranet (LinkSys WRT3200ACM 10Gb router). All data processing and software powering the environment is computed and rendered using this main server. The server sends transformed projection-mapped content to the five projectors using two GeForce® GTX 1080 WINDFORCE OC 8G graphic cards at approximately 60 FPS.

The software powering the environment uses Unity3D for the rendering back-end. Projectors and Kinect cameras are calibrated using the RoomAlive toolkit [13]. The resulting 3D reconstruction of the room imported into Unity3D. Further calibration synchronize the 3D environment with the Vicon tracking system.

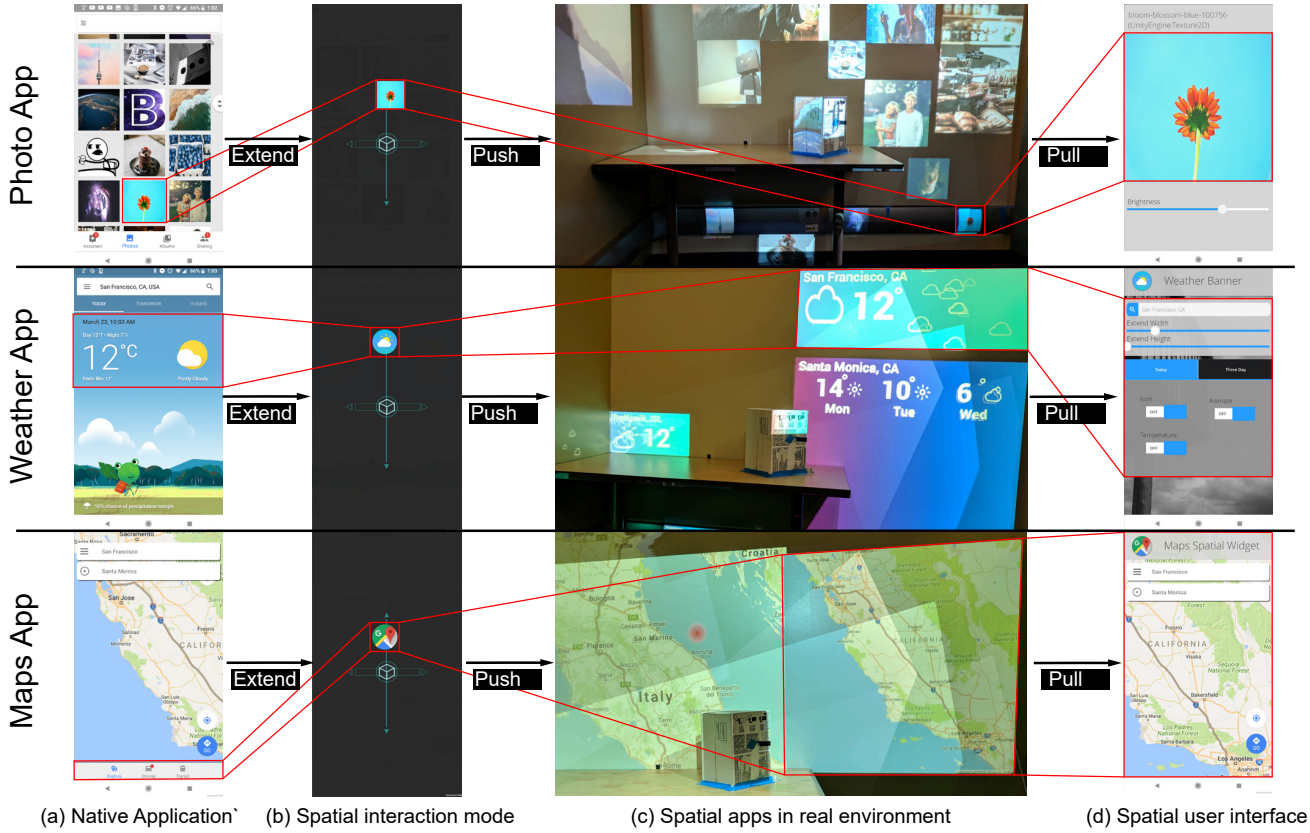

Figure 5: Three example scenarios created with our framework. (a) The native photo, weather, and maps application use the spatial APIs to enable elements of their interface for spatial use (highlighted in red). (b) Content can be pushed onto surfaces in the physical environment by extending the arm and flicking the thumb forward. (c) Spatial content existing within the physical environment and managed through the framework's spatial server. (d) Content already in the environment can be pulled into the smartphone by using a tap & a half gesture which will bring up a customized spatial UI for detailed adjustments on the smartphone.

The smartphone is a Google Pixel (5.0 inch display, 149 × 74 × 11mm with case) running Android 8.1. The complete environment allows for fast and accurate prototyping of various interaction techniques within a spatially enabled environment

4.2 System Architecture

Our framework handles connections, event processing, and room rendering. Each smartphone contains a spatial client running in the background that communicates with the native applications running on the device. The client handles phone localization, gesture recognition, and switching between personal smartphone use to spatial interaction. All communication from a native application to its spatial content is handled through the client by a set of application programming interfaces (API). These sets of APIs provide an interface for mobile applications to create, delete, control, and manipulate associated spatial content.

The spatial server receives communication events from the client, manages the spatial content, and handles projection mapping. All connected projectors are managed by the RoomAlive Toolkit [13]. All spatial content is persisted inside the server where all logic for content layout, such as snapping to planar surfaces, are handled.

4.3 Demonstration Applications

We implemented three prototype applications using Unity3D[1] for a modern smartphone (Fig. 5).

4.3.1 Photos Application (Fig. 5 top). To place a photo in the environment, the user touches a single photo in the application with their thumb. With the thumb on the photo, they extend their arm to activate spatial interaction mode, and flick their thumb forward to push the photo onto the surface the smartphone is pointing toward. This can be repeated for multiple photos. Once a series of images have been placed, the position, scale, and orientation can be determined through the manipulation interactions described above, or other attributes (e.g. brightness) can be controlled by pulling in the photo, bringing up the spatial UI.

4.3.2 Weather Application (Fig. 5 middle). To place ambient weather information in the environment, the user touches a peice of information with their thumb, extends their arm, and then flicks their thumb forward to place the ambient display on one of the rooms surfaces. The location, scale, orientation can be manipulated like with the photos. If the user needs finer control over aspects of the

[1]https://www.unity.com/

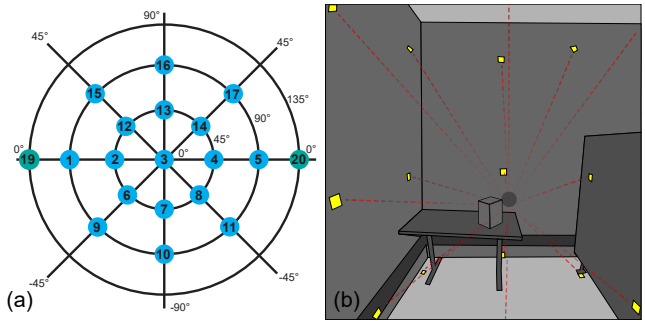

Figure 6: Target placement. (a) Depicts the mapping of targets onto a 2D projected sphere, where target 3 is the forward vector relative to the user's head. The far variant for sit and stand use all the blue targets, and the near variant uses all target up to 11. Supine-far uses a subset from 1-11 excluding 9 and 11; and supine-near uses a subset (1, 2, 4, 5, 10, 13, 16, 19, 20). (b) Illustration of the target placement for the sit-far configuration mapped onto a physical environment.

spatial content, they can capture the spatial UI through the pull gesture described previously.

4.3.3 Maps Application (Fig. 5 bottom). To place a map, the user touches the map bar on the bottom of the application, extends their arm, and flicks their thumb forward. The location, scale, and orientation can be adjusted through the methods stated previously. If the map is placed on the floor, it can create the illusion of walking long the route presented on the map.

5 STUDY 1: EXTENDED VS RETRACTED CLASSIFICATION

Our technique requires robust detection of whether the user is in the *extended* state or the non-extended *retracted* state when the user interacts with the touchscreen. Existing work [7] shows promise that the position and orientation of the phone with respect to the user's head can be obtained using inside-out tracking from the phone. One trivial approach to determining the states is to calculate the distance using ℓ_2-norm from the head to the smartphone and use a simple threshold for delineation. However, as mentioned earlier, this approach would be unable to generalize for deviations in the smartphone's target location, user posture, surface distance, and a user's specific way of extending their hand and their arm length. To demonstrate feasibility of our interaction we need to demonstrate the feasibility of accurately classifying the *extended* vs *retracted* states under the variations of these factors.

We conducted a study to collect data on multiple extend target locations (angles) across three different postures: standing, sitting, and laying down supine, two different surface distances: near and far, across 12 users. We trained a binary classifier on the collected data that consisted of smartphone's position and orientation relative to the head. We also used the user's height as an additional feature to investigate its effect on the classification. We now describe the experiment procedure and classification results.

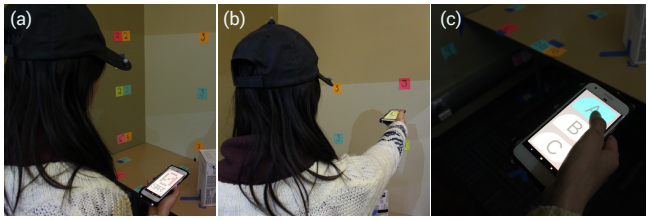

Figure 7: Study 1 trial task. (a) User is prompted to select a target upon which the user taps to confirm their retracted state. (b) User performs the extend gesture and flicks their thumb up on the screen. (c) User retracts the hand and performs a tap to confirm the retracted state and is then prompted to select three buttons ('A', 'B', and 'C') to simulate native phone usage until the next trial.

5.1 Data Collection

We recruited 12 participants, ages 20 to 29, 3 female. All participants were right-handed. Most participants actively used a mobile device an average of 4.1 hours a day. Height ranges between 158 cm to 187 cm and the length of their right shoulder to their index finger ranged from 66cm to 79cm. Participants received $15 for their time.

We collected data for six configurations consisting of posture state (sit, stand, and supine) and room state (near and far): sit-near, sit-far, stand-near, stand-far, supine-near, and supine-far. An office divider 168cm tall and 151cm long oriented perpendicular to one wall allowed us to simulate near and far surfaces.

Physical targets were placed around the user with an associated number and color (Fig. 6). Targets were positioned relative to a canonical head location with angles determined by a laser pointer attached to a smartphone with an orientation sensor. In each stand-far and sit-far configuration, targets were placed in the environment using 0°, 45°, and 90°offsets across both the x- and y- axes relative to their origin point, resulting in 17 directions (Fig. 6a *all blue targets*). In each stand-near and sit-near configuration, targets were generated with a similar approach, resulting in 11 directions (Fig. 6a: *blue targets 1-11*). Supine-far excluded targets 9 and 10, while supine-near used a subset of all 20 targets. Both resulting in 9 directions (see Fig. 6a).

The task in each trial was to extend, point towards a specified target, and retract back (Figure 7). At the beginning of a trial, the participant holds their phone in the non-extended *retracted* manner. They then receive a smartphone prompt to extend and point to a specific target. Participant taps the screen and then extends their arm towards the target and swipes up. The participant then retracts the arm and taps again followed by a series of button presses to simulate phone usage before the next trial starts. The data is recorded at the time of the two taps and the swipe up gesture. Participants were asked to extend their arm naturally without overstraining their arms.

The order of the 6 configurations were counter-balanced using a balanced Latin square. For each configuration, the participant completed a short practice block of trials, then 3 blocks of measured trials consisting of all target positions in a random order. Participants were given breaks after each block to ensure minimal effect of fatigue on the data. Each session lasted approximately 70 minutes. In total, there were 12 participants × (17 [stand-far] + 17 [sit-far] +

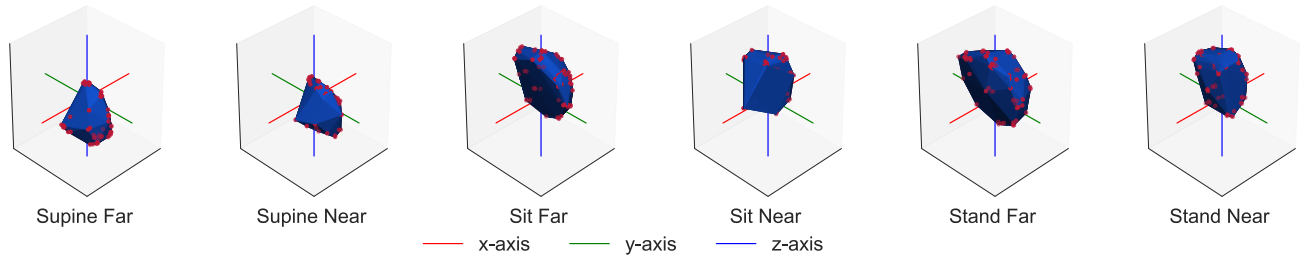

Figure 8: Three-dimensional volumes depicting the extend gesture point clouds. Origin is the head position and axes range from ±100cm .

Classifier		Stand Far	Stand Near	Sit Far	Sit Near	Supine Far	Supine Near
General	*M*	96.09	96.08	97.48	91.65	88.70	82.03
	SD	3.75	4.62	2.66	4.09	14.92	13.25
Per User: Height	*M*	**98.63**	**98.87**	**99.50**	**96.26**	**95.47**	**93.15**
	SD	2.13	1.01	0.56	2.18	4.31	4.73
Per User: No Height	*M*	98.46	98.61	99.45	96.11	93.20	90.35
	SD	2.27	1.18	0.61	1.83	5.03	5.07

Table 1: Three random forest classifiers trained on different variations of user data: *General* is trained on all data using cross-validation; *Per User: Height* is trained for each user using height as a feature; *Per User: No Height* is trained for each user without height. Overall accuracy is 96% (SD 4.5).

11 [stand-far] + 11 [sit-far] + 9 [supine-far] + 9 [supine-near]) × 3 blocks = 2,664 trials that were used for classification.

5.2 Classification

Figure 7 shows the convex hull for the *extended* smartphone's relative position with respect to the head. It illustrates the diversity in the point clouds of the six configurations. We first conduct an analysis of how much of the data can be explained by using a single radius threshold value. The spherical volume that results from the radius delineates the space. We optimized a sphere fitting algorithm that minimises its cost function to find the optimal radius (x) through least squares [18]:

$$\underset{x}{\operatorname{argmin}} \sum_{q \in Q} \left| x - \frac{q_n + q_e}{2} \right|$$

Q is the set of datapoints containing the head to smartphone distances, q_n is the distance in the *retracted* state, and q_e is the distance in the *extended* state. The resulting optimal radius come out to be 53.85cm with a classification accuracy of 82.9%. This shows that the optimal radius can delineate 82.9% of the *extended* and *retracted* data. Of course since the optimization is across the whole data without splitting out a test set, whether this radius value generalizes well is an open question. However, it does indicate that a more advanced classifier that includes the relative position and orientation features might yield a good generalizable performance. We trained a per-user random forest classifier [10] as well as a general leave-one-out cross-validation classifier for each of the six configurations.

5.2.1 Per-user Classifiers. We trained on two blocks of user data and tested on the third. We evaluated all three train-test combinations and averaged the results per user. The overall mean accuracy for all users came out to be 96%. A summary of the results can be viewed in Table 1. For the conditions stand-far, stand-near, sit-far, the classifier shows near perfect accuracies. The sit-near conditions is also high. However, the accuracy for supine-far and supine-near conditions is lower than the other conditions overall. This can be explained by how the participants held the phone while in a the supine posture, which deviated from both the sit and stand postures.

A users' height may influence the length of the hand extension gesture. We added the users' heights as a feature and redid the above analysis. Table 1 shows that while accuracy for the stand and sit conditions remain relatively unaffected, the accuracy in both supine conditions have improved. We conducted Mcnemar's test [8] to compare the performance of the two classifiers for both the supine-near and supine-far conditions; the difference came out to be statistically significant ($p < 0.05$). Thus, including height as one of the features can increase the accuracy of the supine condition by a low but significant percentage for per-user classifiers. Overall, the results show that with user-specific classifiers, the extend gesture is a practical possibility.

5.2.2 General Classifier (Leave-One-Out Cross-Validation). To evaluate the general predictive accuracy of the classifier, when there is no training data from the user, we conducted a 12-fold leave-one-out cross-validation where data from 11 users were used for training and was tested on the 12th user for all 12 combinations. The overall accuracy with a random forest classifier came out to be 92%. A summary of the results in Table 1 shows the accuracy per condition. The accuracy for stand-far, sit-far, and stand-near are good enough for practical use. However, the accuracies of sit-near and supine-far are lower. The accuracy of supine-near at 82% indicates the dependence of the user's specific way of handling a phone when laying on their back. Adding the height feature only added a marginally observable difference in this case and is therefore not reported.

5.2.3 Summary. Overall, our results show that simple heuristics are unable to account for the extend gesture's variance, and by utilizing the smartphone's position and orientation, and the head to smartphone distance of the user, a high degree of accuracy can be obtained (96%), thus demonstrating the feasibility of using arm extension as an intuitive mode switch.

6 STUDY 2: PILOT USABILITY EVALUATION

We evaluate the end-to-end usability of our interaction technique using the three applications we described above in a pilot study.

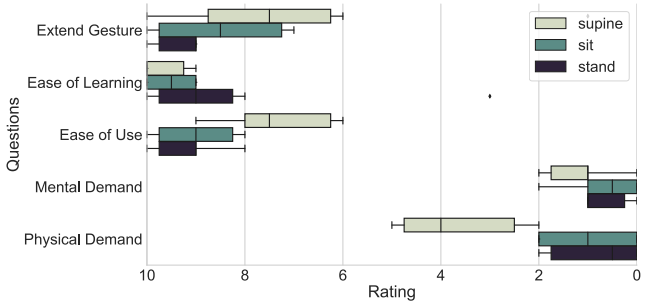

Figure 9: Boxplots showing usability study results.

We recruited 6 participants that did not participate in the previous study: ages 20 to 25, 1 male, all right handed, reported phone usage 3.6 hours per day on average. Remuneration was $10.

The protocol was as follows. First, the participant was briefly instructed on how to use the interaction technique, then they used the system for 5 minutes to familiarize themselves and practice the different actions. Next, they performed the different actions used by the interaction technique while assuming different postures: standing, sitting, and supine (laying down). After, they used the complete interaction technique in realistic usage scenarios enabled by the three prototype applications described above. Again, they completed each scenario while standing, sitting, and supine. At the end, they rated each posture condition on multiple measures, and participated in a closing interview. The posture condition order was counter-balanced.

6.1 Results

Ratings by posture are provided in Figure 9. Each uses a scale from 1 and 10, where 10 is a positive rating for *Extend Gesture*, *Ease of Learning*, and *Ease of Use*. For *Mental Demand* and *Physical Demand*, 0 indicates less demand.

Participants found the interactions easy to use (stand = 9.16, sit = 9, supine = 7.3); easy to learn (stand = 8.16, sit = 9.5, supine = 9.6), and thought they integrated well with the existing smartphone ecosystem. The mental and physical demand were rated low for all postures (lower means less demand) except for supine which was rated higher then the others for physical demand (3.6). Overall, participants found the extend gesture intuitive to use for stand (9.3) and sit (8.5), while the gesture for supine was sometime seen as cumbersome (6.8). Five participants stated that they would use spatial applications at home or office, but all were neutral on using them in a public space. All participants found laying down supine and using a smartphone with a single hand sometimes difficult.

7 DISCUSSION AND FUTURE WORK

7.1 Real World Tracking of a Smartphone

Our current system uses absolute tracking provided by a Vicon motion tracking system to accurately track the smartphone and the user's head position within an instrumented area. This was done to simplify prototyping and provide experimental control, so verifying that our techniques will work outside this kind of fixed tracking environment is currently an open question. However, recent advancements in 3D tracking, using combinations of accelerometer and "inside-out" computer vision [16, 19, 28], are quite robust in current generation mobile AR. Implementing and testing our interaction methods is this kind of ad hoc tracking context remains a topic for future work.

7.2 Extended vs Retracted Classification

Our classification results demonstrate the feasibility of using arm extension as an intuitive mode switch gesture and provide the impetus for the next set of investigations in this space. There are multiple directions of future work pertaining to this classification problem. Firstly, our results currently depend on the awareness of the configurations that the user is in. The user could set this up in the beginning depending on their most frequent use-case and switch it when their configuration changes. The implicit recognition of user posture and surface proximity is a good subject for future work. Secondly, while we demonstrate the feasibility of the extend gesture using robust 3D positions obtained from external tracking, further investigation is needed to ascertain that the 3D position obtained from inside-out tracking using a combination of 3D environment mapping, face tracking, and inertial measurement units provides a similar level of robustness. Thirdly, we observed higher accuracies for per-user classifiers and more work needs to be done to investigate quick user calibrations or on-the-go personalization of the classifier model.

7.3 Extending the Interaction Space

The interaction vocabulary currently supports a subset of the interactions possible within an augmented environment (Fig. 2). A natural extension to explore would be the group manipulation content, content snapping and layouts, and other higher level functionality whereby multiple objects can be manipulated at once.

In our technique design, we purposely created it to be usable across three common postures a user would frequently encounter. However, instead of our posture-invariant technique, it would be interesting to explicitly use these postures to control aspects of application state, changing how the technique functions based on the current posture. These posture-dependent techniques could be an interesting area for future work.

7.4 Direct Touch

Some participants found it difficult to perform the extend gesture while laying down supine (Fig 9). Comments indicate that they had trouble lifting the smartphone away from their body and that they had a hard time holding onto the phone with a single hand when targets were beside them. Other smaller issue came about when targets where generally close in proximity overall. In our interaction space and system implementation, we refrained ourselves from using direct touch for nearby targets so we could focus on at-distance interaction, but investigating manipulation through direct touch would be the logical next step.

For our prototype environment, we utilized a projection-based AR were projectors are calibrated using the RoomAlive Toolkit [27]. The result of this calibration process can sometimes introduce artifacts that may reduce visual fidelity, such as projector misalignment.

Some of these issues could be mitigated through better projector alignment techniques [22, 25] or laser projectors.

7.5 Two-Handed Interaction

We explicitly designed our technique for single-hand interaction, however there are two-handed smartphone techniques, such as viewport pointing [2] and mid-air gestures [12], that have been used for similar types of object manipulation and selection. Previous work indicates raycasting from a phone held by a single hand has some advantages in a SAR environment compared to viewport pointing with two hands [11]. A head-to-head comparison between our one-handed method and two-handed techniques would be an interesting direction for future work.

8 CONCLUSION

Pushing out and interacting with smartphone content in augmented reality is an increasingly relevant problem without any clear solutions so far. In this work, we proposed using the smartphone itself as the mediator of this interaction based on arm extension, a seamless and intuitive way for the phone to switch between the mobile interaction and spatial interaction modes, similar to how users extend and retract their arms when using a remote control. Our interaction technique enables the user to push smartphone content to an external SAR environment, interact with the external content, rotate-scale-translate it, and pull the content back into the smartphone, all the while ensuring comfort, no conflict between the mobile and spatial interactions, and single-handed and eyes-free use in the spatial mode. To ensure feasibility of hand extension as mode switch, we evaluated the classification of extended and retracted states of the smartphone while varying user postures, surface distances, and target locations. Our results show that a random forest classifier can classify the extended and retracted states with a 96% accuracy on average. A final usability study of the interaction space with three demonstrative applications found interactions to be usable and intuitive.

ACKNOWLEDGMENTS

This work made possible by the NSERC Discovery Grant 2018-05187, the Canada Foundation for Innovation Infrastructure Fund 33151 "Facility for Fully Interactive Physio-digital Spaces," and Ontario Early Researcher Award ER16-12-184.

REFERENCES

[1] Michel Beaudouin-Lafon, Stéphane Huot, Mathieu Nancel, Wendy Mackay, Emmanuel Pietriga, Romain Primet, Julie Wagner, Olivier Chapuis, Clément Pillias, James R Eagan, Tony Gjerlufsen, and Clemens Klokmose. 2012. Multisurface Interaction in the WILD Room. *Computer* 45, 4 (apr 2012), 48–56. https://doi.org/10.1109/MC.2012.110

[2] Sebastian Boring, Dominikus Baur, Andreas Butz, Sean Gustafson, and Patrick Baudisch. 2010. Touch Projector: Mobile Interaction through Video. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10 (CHI '10)*. ACM Press, New York, New York, USA, 2287. https://doi.org/10.1145/1753326.1753671

[3] Andrew Bragdon, Rob DeLine, Ken Hinckley, and Meredith Ringel Morris. 2011. Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces - ITS '11*, Vol. 16. ACM Press, New York, New York, USA, 212. https://doi.org/10.1145/2076354.2076393

[4] Wolfgang Büschel, Annett Mitschick, Thomas Meyer, and Raimund Dachselt. 2019. Investigating Smartphone-based Pan and Zoom in 3D Data Spaces in Augmented Reality. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '19*, Vol. 19. ACM Press, New York, New York, USA, 1–13. https://doi.org/10.1145/3338286.3340113

[5] 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 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*. ACM Press, New York, New York, USA, 2527. https://doi.org/10.1145/2207676.2208639

[6] Xiang 'Anthony' Chen, Nicolai Marquardt, Anthony Tang, Sebastian Boring, and Saul Greenberg. 2012. Extending a mobile device's interaction space through body-centric interaction. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services - MobileHCI '12*. ACM Press, New York, New York, USA, 151. https://doi.org/10.1145/2371574.2371599

[7] Xiang 'Anthony' Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott Hudson. 2014. Around-Body Interaction: Sensing & Interaction Techniques for Proprioception-Enhanced Input with Mobile Devices. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services - MobileHCI '14*. ACM Press, New York, New York, USA, 287–290. https://doi.org/10.1145/2628363.2628402

[8] Thomas G. Dietterich. 1998. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation* 10, 7 (oct 1998), 1895–1923. https://doi.org/10.1162/089976698300017197

[9] Lorin J. Elias and Deborah M. Saucier. 2006. Neuropsychology : clinical and experimental foundations. (2006), 531.

[10] Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely randomized trees. *Machine Learning* 63, 1 (apr 2006), 3–42. https://doi.org/10.1007/s10994-006-6226-1

[11] Jeremy Hartmann and Daniel Vogel. 2018. An Evaluation of Mobile Phone Pointing in Spatial Augmented Reality. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, Vol. 2018-April. ACM Press, New York, New York, USA, 1–6. https://doi.org/10.1145/3170427.3188535

[12] Wolfgang Hürst and Casper Van Wezel. 2013. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications* 62, 1 (2013), 233–258. https://doi.org/10.1007/s11042-011-0983-y

[13] Brett Jones, Lior Shapira, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, and Nikunj Raghuvanshi. 2014. RoomAlive: Magical Experiences Enabled by Scalable, Adaptive Projector-camera Units. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*. ACM Press, New York, New York, USA, 637–644. https://doi.org/10.1145/2642918.2647383

[14] Ricardo Langner, Ulrich von Zadow, Tom Horak, Annett Mitschick, and Raimund Dachselt. 2016. Content Sharing Between Spatially-Aware Mobile Phones and Large Vertical Displays Supporting Collaborative Work. In *Collaboration Meets Interactive Spaces*. Springer International Publishing, Cham, 75–96. https://doi.org/10.1007/978-3-319-45853-3_5

[15] Chi-Jung Lee and Hung-Kuo Chu. 2018. Dual-MR: Interaction with Mixed Reality Using Smartphones. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology - VRST '18*, Vol. 18. ACM Press, New York, New York, USA, 1–2. https://doi.org/10.1145/3281505.3281618

[16] Ce Li, Chunyu Xie, Baochang Zhang, Chen Chen, and Jungong Han. 2018. Deep Fisher discriminant learning for mobile hand gesture recognition. *Pattern Recognition* 77 (may 2018), 276–288. https://doi.org/10.1016/j.patcog.2017.12.023

[17] Frank Chun Yat Li, David Dearman, and Khai N Truong. 2009. Virtual Shelves: Interactions with Orientation Aware Devices. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology - UIST '09*. ACM Press, New York, New York, USA, 125. https://doi.org/10.1145/1622176.1622200

[18] Jorge J Moré. 1978. *The Levenberg-Marquardt algorithm: implementation and theory*. Springer. 105–116 pages. https://link.springer.com/content/pdf/10.1007/BFb0067700.pdf

[19] Raul Mur-Artal and Juan D. Tardos. 2016. ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. *IEEE Transactions on Robotics* 33, 5 (oct 2016), 1255–1262. https://doi.org/10.1109/TRO.2017.2705103 arXiv:1610.06475

[20] Brad A Myers, Choon Hong Peck, Jeffrey Nichols, Dave Kong, and Robert Miller. 2001. Interacting at a Distance Using Semantic Snarfing. In *LNCS*. Vol. 2201. Springer-Verlag, 305–314. https://doi.org/10.1007/3-540-45427-6_26

[21] Julian Petford, Miguel A Nacenta, and Carl Gutwin. 2018. Pointing All Around You: Selection Performance of Mouse and Ray-Cast Pointing in Full-Coverage Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press, New York, New York, USA, 1–14. https://doi.org/10.1145/3173574.3174107

[22] Behzad Sajadi and Aditi Majumder. 2012. Autocalibration of Multiprojector CAVE-Like Immersive Environments. *IEEE Transactions on Visualization and Computer Graphics* 18, 3 (mar 2012), 381–393. https://doi.org/10.1109/TVCG.2011.271

[23] Julian Seifert, Andreas Bayer, and Enrico Rukzio. 2013. PointerPhone: Using Mobile Phones for Direct Pointing Interactions with Remote Displays. In *14th IFIP TC 13 International Conference on Human-Computer Interaction, INTERACT 2013*,

Vol. 8119 LNCS. Ulm University, Institute of Media Informatics, James-Franck-Ring, 89081 Ulm, Germany, 18–35. https://doi.org/10.1007/978-3-642-40477-1_2

[24] Mickael Sereno, Lonni Besançon, and Tobias Isenberg. 2019. Supporting Volumetric Data Visualization and Analysis by Combining Augmented Reality Visuals with Multi-Touch Input. (2019), 16–18. https://doi.org/10.2312/eurp.20191136

[25] Ross T. Smith, Guy Webber, Maki Sugimoto, Michael Marner, and Bruce H. Thomas. 2013. Automatic Sub-pixel Projector Calibration. *ITE Transactions on Media Technology and Applications* 1, 3 (2013), 204–213. https://doi.org/10.3169/mta.1.204

[26] Shan-Yuan Teng, Mu-Hsuan Chen, and Yung-Ta Lin. 2017. Way Out: A Multi-Layer Panorama Mobile Game Using Around-Body Interactions. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 230–233. https://doi.org/10.1145/3027063.3048410

[27] Andrew D. Wilson and Hrvoje Benko. 2017. Holograms without Headsets: Projected Augmented Reality with the RoomAlive Toolkit. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '17*. ACM Press, New York, New York, USA, 425–428. https://doi.org/10.1145/3027063.3050433

[28] Jiahui Wu, Gang Pan, Daqing Zhang, Guande Qi, and Shijian Li. 2009. Gesture Recognition with a 3-D Accelerometer. In *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing*. Vol. 5585. 25–38. https://doi.org/10.1007/978-3-642-02830-4_4

[29] Ka Ping Yee. 2003. Peephole Displays: Pen interaction on spatially aware handheld computers. In *Conference on Human Factors in Computing Systems - Proceedings*. 1–8.