TrackCap: Enabling Smartphones for 3D Interaction on Mobile Head-Mounted Displays

Peter Mohr[1,2], **Markus Tatzgern**[3], **Tobias Langlotz**[4], **Andreas Lang**[1,3], **Dieter Schmalstieg**[1], **Denis Kalkofen**[1]
[1]Graz University of Technology [2]VRVis GmbH [3]Salzburg University of Applied Sciences [4]University of Otago
mohr|lang|schmalstieg|kalkofen@icg.tugraz.at,markus.tatzgern@fh-salzburg.ac.at,tobias.langlotz@otago.ac.nz

Figure 1: Our system enables using smartphones for 3D interactions in applications on head mounted displays (HMD). It consists of a planar tracking target attached to the HMD, a tracking software on the smartphone that estimates the pose of the phone relative to the HMD, and a network interface that transmits the pose to the HMD. This system enables interactions of the phone with virtual objects that are placed in world coordinates. In this example, we interact with virtual objects by attaching a smartphone to a plastic toy gun. While previous approaches require holding the device in the field of view of the HMD's camera (black), our system supports a large interaction space by using the camera on the phone for tracking (blue).

ABSTRACT

The latest generation of consumer market Head-mounted displays (HMD) now include self-contained inside-out tracking of head motions, which makes them suitable for mobile applications. However, 3D tracking of input devices is either not included at all or requires to keep the device in sight, so that it can be observed from a sensor mounted on the HMD. Both approaches make natural interactions cumbersome in mobile applications. TrackCap, a novel approach for 3D tracking of input devices, turns a conventional smartphone into a precise 6DOF input device for an HMD user. The device can be conveniently operated both inside and outside the HMD's field of view, while it provides additional 2D input and output capabilities.

CCS CONCEPTS

• **Human-centered computing** → **Interaction devices**.

KEYWORDS

3D Pointing; HMD; Wearable Computing; Augmented Reality; Mixed Reality; Mobile Devices; Input Devices

ACM Reference Format:
Peter Mohr, Markus Tatzgern, Tobias Langlotz, Andreas Lang, Dieter Schmalstieg, Denis Kalkofen. 2019. TrackCap: Enabling Smartphones for 3D Interaction on Mobile Head-Mounted Displays. In CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4-9, 2019, Glasgow, Scotland UK. ACM, New York, NY, USA, 11 pages.
https://doi.org/10.1145/3290605.3300815

1 INTRODUCTION

Virtual Reality (VR) and Augmented Reality (AR) are entering everyday life due to new, inexpensive head-mounted display (HMD) products. The most recent entry-level HMD generation (e.g., Oculus Go and Google Daydream) repurpose smartphone hardware for untethered operation, but do not support tracking with six degrees of freedom (6DOF), as high-end HMD models do. Even with support for 6DOF head

CHI 2019, May 4–9, 2019, Glasgow, Scotland UK

ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300815

tracking on an untethered HMD, the problem of interaction with the environment remains.

Hand tracking has the potential to support natural interaction and it is conceptually straightforward to implement on an HMD using inside-out looking sensors. However, any practical implementation of this concept faces several challenges. First, the integration of hand tracking adds to the technical complexity of the HMD and increases its hardware cost, computational load, and power consumption. Second, the user would have to keep the hands in the field of view of the HMD sensors (see the black outline on the HMD in Figure 1). This may be natural for pointing gestures, but keeping one's hands permanently raised to chest level quickly leads to fatigue. Third, fine-grained manipulation in free space is difficult to perform and lacks the passive force feedback provided by a touchscreen or other surface.

In this work, we propose a different approach by re-purposing a conventional smartphone as a 6DOF tracked handheld companion device for a mobile HMD. We call our approach *TrackCap*, as the phone "tracks a cap" that is mounted on an HMD. The inside-out tracking is turned around by using the smartphone's camera (rather than the HMD camera) to determine the relative pose from HMD to the device. This approach avoids all the pitfalls listed above. First, the technical requirements of the HMD remain unchanged, while an existing smartphone can be re-purposed as an inexpensive 6DOF input device. Second, the tracking only depends on a line of sight from the smartphone to the HMD and not vice versa. In other words, the smartphone can be operated outside of the forward-looking area of the HMD, for example, at hip level, as long as its camera is facing towards the HMD which we will show is a feasible assumption in most cases. Third, the smartphone can be translated and rotated with high precision, and the passive haptic feedback of the touchscreen allows for even more precise input when required. Finally, TrackCap benefits from the independence of HMD and smartphone, which enables retrofitting an existing HMD with an affordable 6DOF input device and allows to freely mix and match devices and interaction styles.

To verify our design, we present the implementation of a TrackCap prototype on current HMD types, including Google Cardboard and Microsoft HoloLens. The tracking software detects the "cap" on the HMD using the smartphone's front camera. It runs as a standalone service on the smartphone and communicates with the HMD by WiFi or Bluetooth. We provide performance tests and user evaluations, comparing our approach to state-of-the-art techniques for 3D selection and manipulation tasks with an HMD. Our evaluation focuses on basic tasks which are essential in most spatial interactions. We conclude this paper with a discussion on design spaces and application scenarios which become possible with our approach.

2 RELATED WORK

Support for spontaneous object selection and manipulation are fundamental requirements for most 3D interactions. Grasping virtual objects with bare hands or instrumented gloves is arguably the most natural selection technique [18]. Glove devices may include tracking of finger motion, while a 6DOF wrist tracker provides the position and orientation of the wrist in the scene [11]. However, simple gloves suffer from real-virtual interpenetration issues. The user can penetrate a virtual object, thus, breaking the immersive experience. Haptic feedback devices [2] can increase the realism and detail of virtual grasping techniques [12, 25] but are unlikely to be available soon for mobile systems. Recent commercial depth sensors provide an opportunity for inexpensive hand and finger tracking [27]. By using projectors, several systems turn real surfaces or even human skin into interactive displays for mobile interactions [6, 14, 21, 31, 32]. However, interacting with real surfaces requires appropriating such surfaces first which is not always applicable.

The most widely used alternative to virtual hand input is raycasting. Here, a selection is determined by intersecting the user-controlled ray with the closest object in the scene [29]. Raycasting enables operation over longer distances and it is flexible in how the ray origin and direction are specified. For example, some works have explored the utilization of smartwatches and their integrated sensors to enable raycasting in Mobile Virtual Reality [10, 15]. As the sensors do not provide 6DOF tracking, the ray origin was usually fixed and only the ray direction was controlled via the smartwatch. Besides the lack of full 6DOF tracking, implementations of raycasting with orientation-only sensors [8] based on inertial measurement units [22] suffer from drift and require frequent re-calibration. To provide 6DOF input, gaze-based techniques and handheld controllers have been considered, which, until recently, required stationary tracking systems. With robust outside-in tracking of mobile HMD types, gaze-based techniques can be applied to selection [13]. However, studies indicate that gaze-based selection techniques are slower than traditional hand-based methods [4] and seem to limit the user's ability to recall the environment [26]. Finally, Microsoft's Mixed Reality headsets provide special controllers that can be tracked using the camera that is mounted on the HMD[1]. While these systems provide precise 6DOF tracking, they require a tethered connection to an external desktop or notebook computer. An even more severe restriction is that field of view of the cameras mounted on the HMD restricts the range in which interaction can be performed. Very recently, also magnetic tracking solutions have been applied to mobile systems as

[1] https://docs.microsoft.com/en-us/windows/mixed-reality

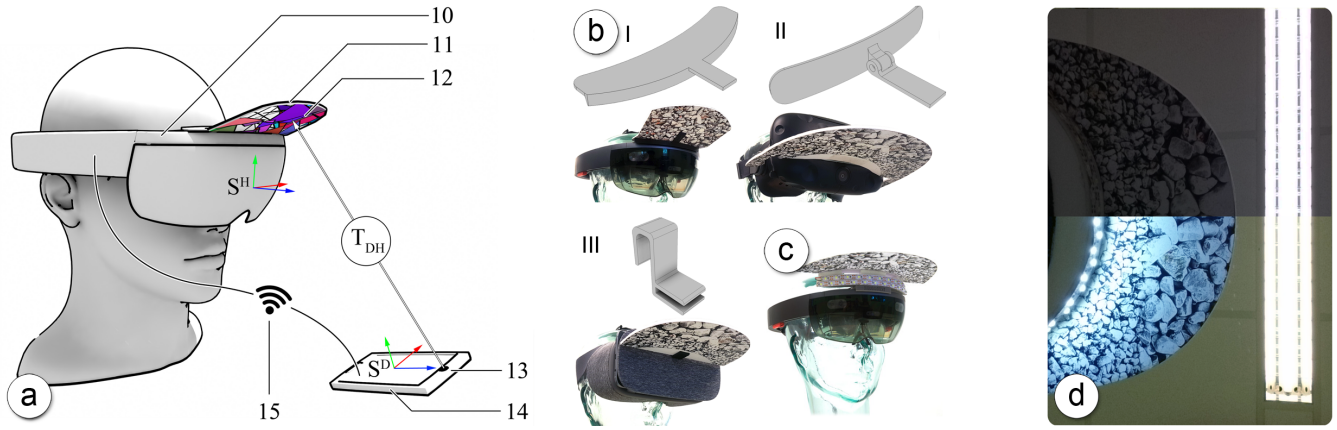

Figure 2: System Overview. (a) Our system consists of 10: HMD, 11: detachable marker mount, 12: planar fiducial marker, 13: camera, 14: smartphone. In addition, the system uses a 6DOF pose estimator on the smartphone, a network module to transmit the estimated pose to the HMD (15), and a pose combiner running on the HMD, which maps the pose of the smartphone into the world coordinate frame. (b) We designed a set of 3D printable parts to mount TrackCap to a variety of HMDs (STL files will be made publicly available). I) Microsoft HoloLens. Note that the cap was designed not to obstruct the view of the scene understanding sensors, so the marker was bent upwards. II) HTC Vive and III) Google Daydream designs use a flat marker instead. Note that we use the HTC Vive only for measuring the precision of TrackCap since the HTC Vive already comes with a precise hand tracking system. (C) We provide additional illumination for the marker to compensate for strong back-lighting from the ceiling. (D) Marker illuminated with and without an additional light source in case of back-lighting.

well[2]. However, magnetic systems commonly require a calibration that is specific to a single environment [16] and needed to reduce electromagnetic interference to acceptable levels. This makes magnetic systems difficult to use in mobile applications that are required to function in unknown environments. Even worse, in some environments in particular industrial environments, the electromagnetic interference will be hard to correct to acceptable levels.

Vision-based tracking has been explored before to enhance the interaction capabilities of controllers such as the VideoMouse by extending the degrees of freedom of a standard mouse [9] or to interact with public displays using a phone [23]. However, these approaches still relied on stationary hardware demonstrated in a constrained environment and were often constrained such as to only measure tilt in certain ranges. Urban Pointer by Langlotz et al. turns a mobile phone into a pointing device for precise pointing in outdoor environments [17]. While not requiring stationary hardware, this approach utilizes pre-captured 3D information to support precise tracking.

We conclude that no previous solution for interaction with an HMD is simultaneously untethered, inexpensive, supports interaction with 6DOF and works anywhere around the body. TrackCap was developed to fill this important spot in the design space of HMD interaction.

3 SYSTEM OVERVIEW

TrackCap provides relative 6DOF tracking from the smartphone to the "cap" on the HMD using inside-out-tracking. Contrary to existing solutions that use inside-out tracking to track a device within the world [30], for world-scale localization, our system uses the high precision 6DOF tracker of the HMD and does not need to perform any non-relative world positioning by itself. The world-space pose of the smartphone is determined by concatenating the relative pose smartphone-HMD and the HMD-to-world measurement. We use a carefully designed fiducial, which is placed on the HMD. The fiducial's size is optimized to be as small as possible, while still being well observable from the handheld input device. Furthermore, we use a standard 6DOF pose estimation based on point correspondences between the current camera frame and the fiducial [19]. Our system is illustrated in Figure 2.

Pose estimation. We implement the pose estimation of the interaction device directly on the smartphone. Our prototype uses planar marker tracking provided by the Vuforia SDK[3] and a fiducial target ("cap") mounted on the HMD. If marker tracking is lost, the system falls back to an IMU based rotation tracking. Figure 1 illustrates the interaction as seen from the device camera of the HMD. Note that the HMD camera would generally not keep the smartphone in sight.

[2]https://www.magicleap.com

[3]www.vuforia.com

Network communication. Our approach requires transmitting the estimated 6DOF pose as well as user input on the touchscreen from the phone to the HMD. Keeping latency as low as possible is crucial, especially when using an optical see-through HMD. Therefore, we send the pose in a single UDP packet over WiFi, using a payload of 28 bytes to represent the position and orientation data and 1 byte to transmit the command header. Additional data from button or touch input is sent only on demand. The UDP data payload for such data consists of a 1-byte command header and 0-4 bytes for optional parameters. We also implemented communication via Bluetooth (BT), which performed similar to the WiFi setup.

Pose combination. After receiving the pose data of the smartphone, we concatenate it with the current world pose matrix of the HMD. Since the pose of the smartphone is calculated relative to the camera center, we add an offset to the center of the physical device. Additionally, we take the offset between the origin of the HMD tracking system and the center of the fiducial marker into account. Both are static transformations and need to be defined only once.

Technical analysis

Our system aims mainly at see-through displays, like the Microsoft HoloLens, since one can see one's hands using the device but also supports immersive VR displays including entry-level systems such as Google Cardboard (see Figure 2(b)). Since our system runs the computationally expensive tracking of the input device on the smartphone itself, performance on the HMD only depends on the application. For our lightweight test scenes, we were able to achieve high frame rates on all test devices – 90 frames per second (fps) on the HoloLens and 60 fps on the Daydream. Tracking performance on the smartphone was 30 fps, limited only by the camera frame rate.

In addition to framerate, we compared the precision and latency of our tracking solution to the outside-in tracking provided by the HTC Lighthouse system. Therefore, we added a setup using TrackCap with the HTC Vive (Figure 2(c)). To compare the Lighthouse tracking performance to TrackCap, we mounted an additional Lighthouse tracker on the smartphone. This rig was calibrated so that the HTC tracker's virtual center point coincided with the center of the smartphone.

To sample the interaction space around the user, we placed a 3D grid of $4 \times 4 \times 3$ reference points within the world coordinate system. Position and orientation data of both tracking systems were collected for a duration of three seconds at each reference point. During the procedure, we also recorded the network latency. The results are shown in Table 1. TrackCap delivered a positional error below 10mm, an orientation error of less than 2°, an average WiFi latency of 10 ms and an average latency of 15 ms using a Bluetooth connection.

Error/Latency	Position	Rotation	WiFi	BT
Mean	9.812 mm	1.849°	10 ms	15 ms
Std. Dev.	3.903 mm	0.291°	14 ms	6 ms

Table 1: Tracking precision and latency of the TrackCap system.

In theory, the smartphone's front camera FOV could be a limiting factor, since it determines the possible tracking range. In practice, a typical horizontal FOV of a recent phone range from 56.3°(HTC One M8) to 71.4°(ZenPhone AR), and newer hardware tends to have an even larger FOV. In our studies, we did not notice any differences across our test devices in terms of coverage. The tracking range of our solution, as tested using a Samsung Galaxy S8 phone (68.0°FOV), covers a $1 \times 1 \times 0.6m$ volume in front of the HMD. Since the phone will necessarily be held no further than arm's length, the tracking volume proved to be sufficient.

During our tests, we noticed that pointing the smartphone camera upwards introduces occasional tracking failures caused by strong backlighting, e.g., from overhead lights or the sun, as the dynamic range of the camera is limited and automatic exposure correction makes the marker appear very dim. Figure 2(d-top) illustrates the problem. To mitigate the effect of strong backlighting, we installed a USB-powered LED array to illuminate the maker, as shown in Figure 2(c). The effect on the camera image with and without the additional lighting can be seen in Figure 2(d-bottom). We visually evaluated its improvement in different but challenging environments. However, as battery consumption is important for mobile systems, in our future work we will systematically measure its improvement in order to adjust additional lighting based on the current environment lighting.

4 EVALUATION

We performed a series of evaluations on the performance of TrackCap versus other mobile, untethered input options. The evaluations focus on object selection and manipulation tasks as fundamental elements of 3D interactions. TrackCap is compared to standard methods for 3D interaction, as available for commercially available untethered systems, such as the Google Daydream.

Since we are interested in how well TrackCap can support natural interactions, we disabled any supporting visualizations, such as a crosshair or a thin ray, during our experiments. Note that future applications of TrackCap would most likely make use of such supportive visualizations. However, supportive visualizations may require additional calibration effort and may be a confounding factor in experiments. In the interest of brevity, we also decided to steer clear from comparing TrackCap to gaze-based interaction techniques,

since previous research indicates that gaze-based techniques are slower than hand-based methods and restrict the user's ability to recall the environment [26].

For all evaluations, the data was evaluated using a significance level of 0.05. The data did not fulfill the normality requirements and, therefore, was analyzed using Wilcoxon signed-rank tests, effect sizes are calculated as $r = \frac{Z}{\sqrt{N}}$ as proposed by Fritz et al. [5]. If not indicated otherwise, values in the text are reported in the format "mean (sd)". The analysis was performed using the statistics software *R*.

Experiment 1: Selecting distant objects

We tested the capability of TrackCap for distant object selection by comparing the interaction supported by TrackCap to a hand-held IMU [8] in a picking-by-raycasting task. We expected TrackCap to perform better than IMU due to the inherent drift of the latter. Since we wanted to test TC for natural pointing, we did not show any visual correction aids.

Task. We designed a pointing task similar to the standard Fitts's law test [1](Figure 3(a)). To maximize the usage of motor and interaction space, we force users to move in 3D by distributing target spheres in the 3D space around the user (Figure 3(b)). 21 blue spheres were arranged in a circle (radius $2m$) around participants. The task started by pointing the mobile device at the first highlighted sphere and clicking. The next sphere was located on the opposite side and required the user to turn. Spheres hit by the invisible pointing ray were highlighted by changing the color to gray to provide visual feedback. To vary task difficulty spheres were of different sizes (5, 10, 15cm). Circle centers were set to the height of the users' head with a random offset (±0.175m). Due to the small FOV of the HMD, the application guided the participant to the targets by showing green arrows at the border of the view area pointing into the direction of the target.

Design. We designed a repeated-measures within-subject study. We defined an independent variable "system" with two conditions: TrackCap and mobile device only (MBO). In MBO, the ray orientation relied only on the internal 3DOF sensor. We measured task completion time (TCT), i.e., time between successive clicks on targets, error rate, i.e., percentage of spheres missed, subjective workload with the raw NASA TLX [7], usability with the Single Ease Question (SEQ) [24], and overall preference. Eight participants (1 female, $\overline{X}$ =30.3 (4.2) years) volunteered. On a scale from one to five (best), the mean of self-rated AR experience was 3.3 (1.2).

Apparatus. We used an HMD (Microsoft HoloLens) and a mobile phone (Samsung Galaxy S7). The spheres were visible in the HMD view, the mobile device controlled the ray orientation. Input was confirmed by touch on the phone. 6DOF head tracking was achieved via the HoloLens. For the TrackCap condition, the smartphone was registered in the same coordinate system as the HoloLens, and drift was compensated using TrackCap. For MBO, the ray orientation depended only on the hardware sensors of the mobile phone.

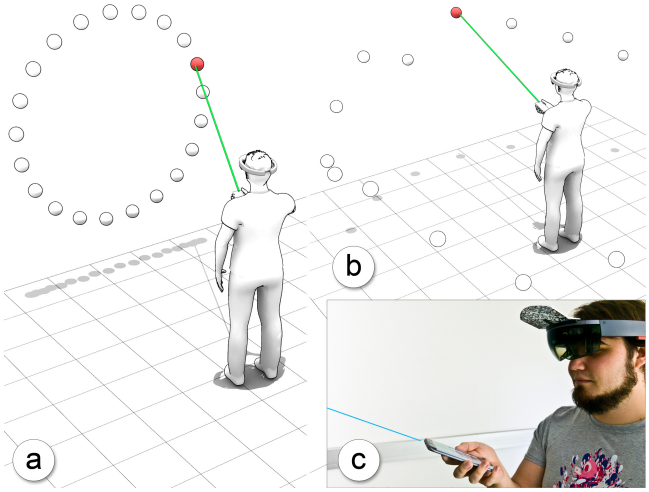

Figure 3: Study setup to measure performance during the selection of distant objects. (a) Fitts's law test on a virtual plane in front of the user. (b) Variation of the Fitts's law test in 3D. The targets are located in a circle around the user, with varying heights. (c) User with HMD during the task. The user sees a highlighted picked target sphere. The virtual picking ray is shown for demonstration.

Procedure. The starting order of systems was counterbalanced using a Latin Square setup. The systems were tested by using trial blocks, consisting of 21 trials each. Sphere sizes varied between trial blocks and were presented in random order. The height of spheres was randomized within a trial block. Participants were standing throughout the task and used their dominant hand for pointing. Participants familiarized themselves with the system by performing two test blocks, then performed the task by repeating one block for each size condition. Participants were instructed to be fast and accurate. Between blocks, participants were forced to rest for 10-20 seconds to recover from fatigue due to the mid-air interaction [34]. Upon completion of the condition, users filled in the SEQ and NASA TLX questionnaire and continued with the remaining system. After completing the final task, the user filled out the preference questionnaire.

Hypotheses. We expected that TrackCap would successfully enable mobile interaction and compensate for IMU drift. Therefore, TrackCap will perform better than MBO with respect to TCT (**H1**) and error rate (**H2**).

Results. Wilcoxon signed-rank tests revealed significant differences between TrackCap and MBO for error rate (TrackCap 35.1 (24.4); MBO 75.6 (16.2); Z=2.20, p<0.05, r=0.78), TLX (TrackCap 45.3 (20.9); MBO 79.7 (12.5); Z=-2.52, p<0.01, r=0.89) and SEQ (TrackCap 4.8 (1.3); MBO 1.5 (0.5); Z=2.58, p<0.01, r=0.91).

Discussion. We investigated the ability of TrackCap to provide intuitive and precise 3D pointing interactions when compared to a standard technique using only an IMU. Our results show that TrackCap enables more precise interactions with lower perceived task load. Therefore, we accept H2. We believe that the significantly higher error for MBO is caused by a large amount of drift introduced during interaction using the orientation sensors on the smartphone. TrackCap is able to reliably and automatically compensate for this drift and make pointing natural and intuitive.

The data did not reveal any significant difference in TCT. Therefore, we reject H1. To ensure natural and intuitive pointing, we asked the user's to not only be accurate, but also fast. We believe this instruction may have influenced their behavior, as they did not take much time to manually compensate for the drift in MBO. This behavior was intended, since we are aiming at a system for intuitive pointing in 3D. Participants stated that the drift and the many failures in MBO were frustrating, which influenced the time users spent on trying to hit the targets as the task progressed. These observations are reflected in the significantly higher task load and lower perceived ease of use of MBO. In addition, when asked for preferences, all participants (100%) preferred TrackCap over MBO.

Experiment 2: Selecting close proximity objects

TrackCap supports 6DOF input and allows to implement direct 3D object selection techniques [20]. Therefore, we compare direct 6DOF selection (using TrackCap) to 3DOF raycasting that would otherwise be used in such a scenario, in this case the approach of Hincapié-Ramos et al. [8].

Task. In contrast to the first experiment, this experiment investigated the direct selection of targets within a user's reach. The task uses the same setup as in the first experiment with the spheres placed within arm's length so that participants could reach them comfortably (see Figure 4(a)).

Design. We designed a repeated-measures within-subject study with the independent variable "system" having two conditions: TrackCap and MBO. We measured TCT, error rate, i.e., percentage of spheres missed, subjective workload using raw NASA TLX, usability using the SEQ, and overall preference. Eight participants (all male, $\overline{X}$ =31.1 (3.4) years) volunteered. On a scale from one to five (best), the mean of self-rated AR experience was 3.5 (1.2).

Apparatus. The same as in the first experiment.

Procedure. The height and distance of the spheres were set up so that each participant could reach them comfortably. The procedure was the same as in the first experiment.

Hypotheses. We expected that direct selection using TrackCap will be more successful than MBO, due to the lack of sensor drift in TrackCap and its ability to track the position

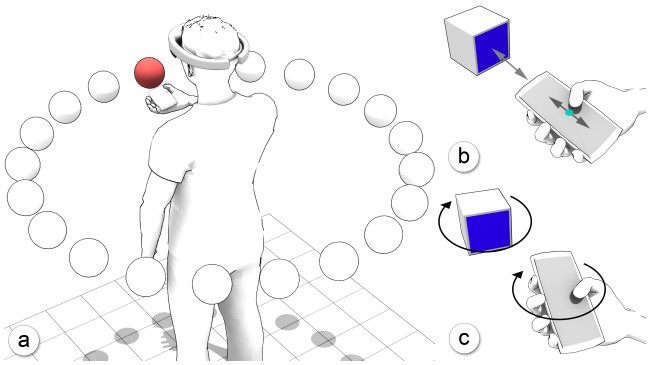

Figure 4: Selection Close Proximity Object and Object Manipulation. (a) Using the 6DOF pose generated by TrackCap, the user can select virtual objects by simply touching them with the physical device. (b) Swiping on the touchscreen moves the object closer or further away. (c) Device rotation is mapped to the objects local coordinate system.

of the input device in 3D space. Therefore, TrackCap will perform better than MBO with respect to TCT (**H3**) and error rate (**H4**) in the direct selection task.

Results. Wilcoxon signed-rank tests revealed significant differences between TrackCap and MBO for error rate (TrackCap 10.1 (8.6); MBO 27.4 (24.1); Z=2.2, $p<0.05$, r=0.78), TLX (TrackCap 27.1 (13.7); MBO 55.1 (20.4); Z=-2.52, $p<0.01$, r=0.89) and SEQ (TrackCap 6.4 (0.7); MBO 4.0 (1.4); Z=2.58, $p<0.01$, r=0.91).

Discussion. The results appear similar to the first experiment, showing that TrackCap also performed better than MBO for selecting objects in close proximity regarding the error rate, but not TCT. Hence, we accept H4 but reject H3. Similar to the first experiment, the users mostly commented on the high frustration with the MBO interface. This is also reflected in the significantly higher task load and lower perceived ease of use of MBO. When asked for preferences, seven participants (87.5%) preferred TrackCap over MBO.

Experiment 3: Object manipulation

The second experiment investigated the direct *selection* of objects in the vicinity of the user, the third experiment investigates direct *manipulation* of such objects. We show the practical value of our approach by comparing direct 6DOF manipulation via TrackCap to an established 3DOF manipulation technique utilizing raycasting for selection and hand-centered manipulation in combination a fishing-reel technique4(b)), as proposed by Bowman et al. [3].

Task. The task uses a similar setup as in the second experiment. However, instead of spheres, this task uses cubes with colored sides. Participants had to alternate selection between opposing cubes in their surroundings. However, in this experiment, participants had to drag and drop the selected cube

to the opposing side and align it with a corresponding platform in the target location. This required to translate and rotate the cubes. The orientation of the cube was defined uniquely by the differently colored sides (Figure 4(c)). For this experiment, the rotation of the target was limited, so that the smartphone camera was able to observe the cap during the entire task. In a follow-up experiment (reported below), we extended the task to include full 360°rotational changes.

Design. We designed a repeated-measures within-subject study to compare the performance and user experience of drift-compensated 6DOF TrackCap for direct manipulation and common 3DOF mobile device interaction. Holding a button on the touch screen allowed participants to grab objects; releasing the button also released the object. We compared TrackCap to a raycasting manipulation with a fishing reel: After selecting objects by raycasting, objects stick to the ray. Thus, the 3DOF of the interaction device manipulated the orientation of the object. The distance of the object along the ray could be manipulated using a sliding motion on the touch screen of the mobile phone. We defined an independent variable "system" with two conditions: TrackCap and MBO. We measured TCT, error, i.e., precision of the alignment, the subjective workload using the raw NASA TLX, usability using the SEQ, and overall preference. The participants of the second experiment took part in this experiment.

Apparatus. The same as in the first experiment.

Procedure. The height and distance of the cubes were set up for each participant so that they could be reached comfortably. The orientation required to align the cube was randomized. The rest of the procedure was the same as in the first experiment.

Hypotheses. We expected that direct manipulation using TrackCap will be more successful than MBO due to the more natural interaction. Therefore, TrackCap will perform better than MBO with respect to TCT (**H5**) and error rate (**H6**) in the direct manipulation task.

Results. Wilcoxon signed-rank tests revealed significant differences between TrackCap and MBO for TCT (TrackCap 13.9 (4.5); MBO 20.1 (7.1); $Z=2.2$, $p<0.05$, $r=0.78$) and TLX (TrackCap 33 (9.2); MBO 63.8 (16.9); $Z=-2.52$, $p<0.01$, $r=0.89$).

Discussion. Participants could interact significantly faster when using direct manipulation supported by TrackCap than when using the fishing reel metaphor of MBO. Therefore, we accept H5. However, we did not find a significant difference in error rate, why we reject H6. There was also no significant difference in perceived ease between TrackCap and MBO. However, the significantly lower TLX for TrackCap indicates that directly interacting with virtual objects using TrackCap is less demanding. Consequently, all participants (100%) preferred TrackCap over MBO. In contrast to the previous experiments, there was no significant difference in error rate between TrackCap and MBO. We believe that participants could efficiently align the virtual cubes despite the drift of MBO due to the visual feedback provided by the cube itself. Hence, participants could compensate for the drift using the virtual cube as a visual reference.

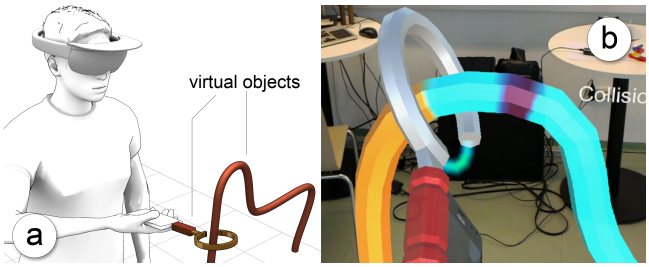

Figure 5: Complex object manipulation - AR wire game. (a) Illustration of the AR game used to measure the performance of TrackCap in complex object manipulation. (b) Screenshot through the HoloLens as seen by a user.

Apart from the more natural manipulation using TrackCap, a part of the difference in TCT could also be explained by the need to successfully select the virtual object before being able to continue the alignment task. During the selection phase of the task, participants had to select the virtual cube using the invisible raycasting. Participants could not skip this selection step as easily as in the previous experiment, but had to take their time to align the drifting raycasting with the virtual object before continuing. Only after the virtual cube was stuck to the invisible ray, participants could visually compensate for the drift. The focus of these experiments was to evaluate the ability of TrackCap to automatically compensate for drift. Future work will additionally investigate the ability of users to be able to compensate for drift by providing visual cues. Visual aids have been shown to have an impact on pointing tasks such as these ones [28].

Experiment 4: 6DOF interaction by camera switching

TrackCap performs well in manipulation tasks consisting of 3D positional changes and moderate rotational changes. However, we are interested in its performance when supporting full 6DOF interactions. Therefore, we extended TrackCap to use the smartphone's IMU, allowing us to switch between the front and back camera of the mobile device depending on its current orientation.

Task. We set up a "don't touch the wire" game requiring full 6DOF interactions (Figure 5). Users move a virtual wire loop along a winding pipe in 3D without colliding. Collisions with the pipe are indicated via sound and a particle spray.

Design. We designed a repeated-measures within-subject study to compare the performance and user experience of our dual-camera TrackCap with a system using the Google Tango API providing 6DOF SLAM tracking. We define an

independent variable "system" with two conditions: the modified TrackCap (CamSwitch) and Project Tango (Tango). We measured TCT, error rate of the task, i.e., the number of hits between the wire loop and the pipe, subjective workload using the raw NASA TLX, usability using the SEQ, and overall preference. Eight participants (1 female, $\overline{X}$ =32.5 (3.9) years) volunteered. On a scale from one to five (best), the mean of self-rated AR experience was 3.9 (1.1).

Apparatus. The apparatus consisted of an HMD (Microsoft HoloLens) and a phone supporting Project Tango (Asus ZenfoneAR). Wire and pipe were visible in the HMD view only. The mobile device controlled the wire.

Procedure. The height of the pipe was roughly set to the height of the participant. The starting order of systems was counterbalanced using a Latin Square setup. Participants were standing throughout the task and used their dominant hand to move the wire. Participants played two test games before starting the task and were instructed to be fast and accurate. After each condition, they filled in SEQ and NASA TLX. Finally, the preference questionnaire was filled out.

Hypotheses. Due to stable 6DOF tracking of the device, we expected to see no difference in TCT (**H7**) and Error (**H8**), when compared to Tango.

Results. The planned equivalence tests require a large sample size. However, we aborted the experiment after eight participants, because the feedback and our observations indicated that CamSwitch suffered from the time-consuming switching between cameras that influenced interaction performance. We checked for significant differences to explore the differences between CamSwitch and Tango. A Wilcoxon signed-rank test revealed a significant difference for SEQ (CamSwitch 2.1 (0.8); Tango 6.3 (0.7); $Z=2.56$, $p<0.01$, $r=0.91$).

Discussion. CamSwitch suffered from technical issues. The time required for switching between back and front cameras made an uninterrupted motion in CamSwitch infeasible. Participants were forced to wait for the camera switch, leading to frustration as indicated by the 100% preference and the higher perceived ease of use of the Tango device. The waiting time is also reflected in the higher TCT of CamSwitch, when compared to Tango (CamSwitch 28.4 (14.9); Tango 16.3 (4.4)). However, the small error rate of CamSwitch indicates that TrackCap is suitable for precise 6DOF motion in 3D space (CamSwitch 1.5 (2.1); Tango 1.5 (1.7)). Therefore, TrackCap would likely benefit from better support for dual camera solutions on smartphones, which can quickly search for the "cap" in both camera streams simultaneously.

Experiment 5: Fusing TrackCap with SLAM

As demonstrated in the previous experiment, a self-contained model-free 6DOF tracking system such as Tango makes a smartphone even more valuable as an input device companion to an HMD. Even though Tango hardware will likely not become available to a mass audience, self-contained 6DOF tracking with somewhat lower performance is becoming available as part of Google's ARCore or Apple's ARKit. Even though these solutions rely on the opportunistic mapping of the environment and can easily become confused under fast motion, they can support an enhanced version of interaction in the style of TrackCap. We were interested in a longer-term technical trajectory, where technologies such as ARCore are widely available, and users would like to use them for fast motion, rather than the slow interaction of our previous experiments. Therefore, we designed a final experiment to assess the benefit of TrackCap to a self-contained model-free 6DOF tracking as well.

Task. The task was inspired by tennis ball serving machines. Virtual balls (diameter=10cm) were thrown at a constant speed (1m/s) at the participant, who had to hit the ball using the mobile device as tennis racket to make the ball disappear. Once per second, red or blue balls started at the same location, moving in a random direction within a cone of 30 degrees. Red balls had to be hit with the front of the racket, blue balls with the back (see Figure 6).

Design. We designed a repeated-measures within-subject study to compare the performance and user experience of the combination of TrackCap and Project Tango, and a system using the 6DOF tracking of Project Tango only. Therefore, we define an independent variable "system" with two conditions: Project Tango and TrackCap (TangoCap) and Tango Only (Tango). We measured task completion time (TCT) and success rate of the task, i.e., the number of spheres hit, the subjective workload using the raw NASA TLX, usability using the SEQ and overall preference. Eight participants (1 female, $\overline{X}$ =30.5 (3.3) years) volunteered. On a scale from one to five, five meaning best, the mean of self-rated AR experience was 3.5 (1.4).

Apparatus. The apparatus consisted of an HMD (Microsoft HoloLens) and a mobile phone with Project Tango support (ASUS Zenfone AR). The spheres were visible in the HMD view only. The mobile device was used to control the tennis racket.

Procedure. The height of the ball emitter was set to the height of the participants. Then users were introduced to the first condition. The starting order of systems was counterbalanced using a Latin Square. Participants performed runs of 100 task repetitions, i.e., they had to hit 100 balls in a row. Participants were standing throughout the task and used their dominant hand for holding the virtual racket. After the participants familiarized themselves with the interaction method by performing two test runs, the task was performed by repeating one run for the system for each size condition. The participants were instructed to be fast and accurate. Upon completion of the condition, users filled in

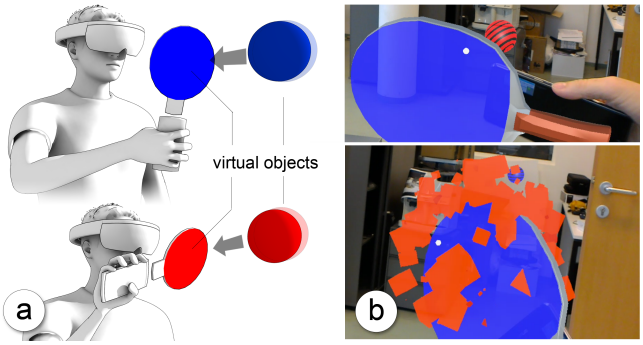

Figure 6: AR Squash. We implemented a squash game for evaluating the performance of TrackCapin the complementary operation with a model-free tracking solution. (a) Illustration of the interaction. Blue and red balls are thrown towards the user, who has to hit them with the matching side of the virtual paddle (indicated by blue and red colors). (b) Screenshots captured through the HoloLens while playing the game. Balls explode when they are hit.

the SEQ and NASA TLX questionnaire. The procedure was repeated for the remaining system. After completing the task with the last system, the user filled out the preference questionnaire.

Hypotheses. We expected that TrackCap could successfully support the relocalization of Tango, if tracking was lost. Due to the speed and appearance of balls at fixed time intervals, we did not expect to see differences in TCT (**H8**). However, we expected that TrackCap would supplement the capabilities of Tango and lead to better error rates in this task than when only using Tango tracking only (**H9**).

Results. Wilcoxon signed-rank tests revealed significant differences between TangoCap and Tango for success rate (TangoCap 81.4 (12.5); Tango 65.4 (20.9); $Z=2.52$, $p<0.01$, $r=0.89$) TLX (TangoCap 31.6 (14.5); Tango 45 (16.4); $Z=2.52$, $p<0.01$, $r=0.89$) and SEQ (TangoCap 5.4 (1.1); Tango 3.1 (0.8); $Z=2.4$, $p<0.05$, $r=0.85$).

Discussion. As expected, relocalization failed after the device lost tracking due to fast motion. This required the user to scan the room for a position known to the Tango device, thereby slowing down the user interaction. However, TrackCap ensured fast and accurate relocalization after tracking failure due to fast motion. This is reflected in the significantly higher rate of balls that users hit successfully. Note that measuring timing difference was not possible due to the balls spawning at constant time intervals. Due to the nature of TrackCap, this method works reliably also in unknown environments. Participants preferred TangoCap (87.5%) and found it more intuitive and easier to use, which is reflected in the lower workload and higher perceived ease of use. Overall TrackCap could successfully expand the usability of the existing 6DOF Tango tracking.

5 CONCLUSION

We have presented TrackCap, a novel system aimed for mobile VR and AR that allows for spontaneous, precise, and natural interactions with 6DOF using consumer-grade smartphones. TrackCap successfully extends the interaction space of existing HMD interaction with commodity smartphones, making it immediately affordable and widely deployable. This has implications in virtually all application areas of AR/VR, including industry, communication or gaming. The method also blends nicely with existing interaction methods and can increase their practical usability.

Evaluation summary. We have presented evaluation results that indicate that TrackCap improves over current HMD input devices in standard 3D selection and manipulation tasks. Our results show that TrackCap allows for precise interaction at a distance and in close proximity. In these scenarios, TrackCap outperforms traditional techniques that rely only on the IMU of the smartphone or input device.

Our fourth experiment showed a limitation of our system. When designing a task that required rotating the phone's camera out of view of the "cap", the 6DOF tracking was lost. To compensate for this issue, we expanded the capability of TrackCap to switch between front and back camera of the smartphone, depending on its orientation to the user. However, we found that most current smartphones cannot switch between front and back camera sufficiently fast. Despite the systematic interruption, TrackCap also allowed for precise 6DOF interaction in this situation. It is also worth mentioning that only a few phones (e.g., HTC M8) support simultaneous use of both cameras and do not require a camera switch. However, this is not a fundamental technical limitation; adding simultaneous support for both cameras should be cheaper than adding additional sensors or other hardware components, as in Project Tango.

Under fast motion, where the tracking of Project Tango was thrown off, TrackCap successfully supported the Tango's relocalization and enabled more fluid interaction than when using Tango alone. This demonstrates that TrackCap is not only able to make use of older smartphones for interaction, but also extended the usability of current solutions, such as those based on ARCore and ARKit.

Limitations. There are several limitations that are worth mentioning. First, we intentionally decided to not include supportive visualizations in the user evaluation. Our system, like most practical applications, supports visual aids such as cross-hairs or virtual laser pointers that might affect the performance in the studies. However, we focused on showing the ability of our system to provide reliable and intuitive interaction using only natural hand-eye coordination and proprioceptive cues.

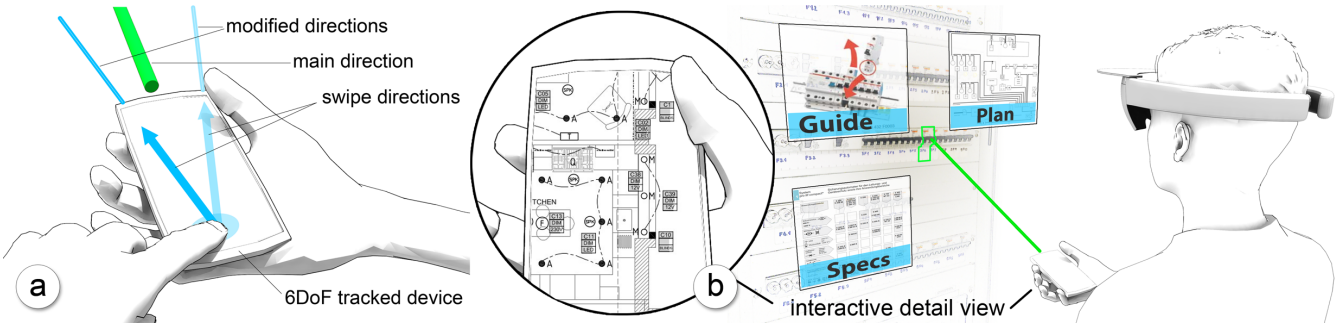

Figure 7: Application scope. TrackCap enables new means of interaction. For example, rays for selecting objects can be quickly defined by swiping over the touchscreen (also see the accompanying video). (b) The high input and output fidelity of the smartphone can also be used to display detail of the selected objects and to enable high precision interactions with them.

Similarly, we did not focus on utilizing the screen for complex interaction apart from confirming a selection. However, as outlined later there are opportunities there to further improve the results but we intentionally focused on the spatial interaction with the controller (smartphone) for the studies.

Another point for discussion is the limited number of participants for our studies. This limitation is owed to the exploratory nature of our experiments. We studied several aspects of our system using five different experiments that took two hours per participant and included a large number of trials and measured samples.

6 DISCUSSION

We argue that the work has relevance beyond the scope of this paper. Foremost, we show the lack of input devices that specifically aim for spontaneous and natural interaction with a mobile HMD. This is particularly important for commercial AR/VR solutions that leave the boundaries of scientific environments and find application in classrooms, workplaces, but also for personal entertainment and recreation.

Our work shows that inside-out tracking can be a feasible option for hand controllers, but has been largely ignored. We believe this finding is significant, as smartphones are ubiquitous and the alternatives require additional hardware or put additional constraints on the user or the environment.

TrackCap enables at least two new research directions: 1) Multi-touch gesture interfaces for AR (e.g. the swipe gesture) and 2) Mobile visual analysis in AR, incorporating the high-resolution touch-screen of the controller (Figure 7(a) illustrates a swipe gesture interaction while 7(b) demonstrates the integration of a high-resolution touchscreen in an AR analysis scenario. The data analysis task that is demonstrated in this example requires tools for quickly selecting, exploring and annotating the data. All of these tools can all highly benefit from the high input and output fidelity of a modern smartphone display.

Furthermore, note that recent mobile HMDs have been equipped with electromagnetic 6DOF controllers. However, TrackCap offers a number of advantages over such controllers. 1) TrackCap offers spontaneous, ad-hoc interaction using a pervasive device: a mobile phone. 2) TrackCap is not sensitive to electromagnetic interference, which is of relevance in many environments. 3) TrackCap offers high resolution, haptic, and multi-touch input as well as high-resolution output. 4) TrackCap can be retrofitted to almost all HMDs, and as the cap mount can be 3D printed, TrackCap comes with almost no additional cost. 5) Given 4), TrackCap is an enabler for further research on spontaneous interaction and can be extended. Finally, 6) TrackCap can support multiple controllers, which 7) can be easily shared among various users. Note that the computational effort is mainly concentrated on the phone, so that TrackCap easily scales with additional controllers (e.g. for both hands).

Finally, we believe that SLAM systems and TrackCap should be combined. SLAM system are able to support tracking when the cap is not visible while TrackCap enables SLAM re-initialization and tracking in dynamic environments, which is still challenging for all recent SLAM systems.

7 FUTURE WORK

Adopting phones as controllers is not a compromise, but an opportunity to exploit capabilities so far not offered by most available controllers. While we show the performance of TrackCap in a variety of experiments, we did not yet utilize the full potential offered by the touchscreen or haptic feedback [33]. Figure 7 shows an illustration of this concept where TrackCap is used to interact with the digital environment shown in an HMD, while offering additional controls and providing haptic feedback. We also did not further explore how personal phones can fuel personalization of the experience or support collaborative activities.

We wanted to support unmodified phones, why we use the IMU as fallback when tracking is lost. However, we will also explore the use of additional wide angle lens adapters to improve performance. Furthermore, future solutions can avoid tracking the cap and track the HMD directly. While one

advantage of the existing approach is that it can be retrofitted to existing HMD designs, a streamlined version can include the "cap" into the HMD design itself.

8 ACKNOWLEDGMENTS

This work was enabled by the Competence Center VRVis, the FFG (grant 859208 - Matahari) and the EU FP7 project MAGELLAN (ICT-FP7-611526). VRVis is funded by BMVIT, BMWFW, Styria, SFG and Vienna Business Agency in the scope of COMET - Competence Centers for Excellent Technologies (854174) which is managed by FFG. Furthermore, Tobias is partially supported by Callaghan Innovation, host of the Science for Technological Innovation National Science Challenge, Seed Project 52421, and by the Marsden Fund Council from Government funding, administered by the Royal Society of NZ.

REFERENCES

[1] 2011. ISO 9241-420:2011: Ergonomics of human-system interaction – Part 420: Selection of physical input devices, International Standard, International Organization for Standardization.

[2] Christoph W. Borst and Arun P. Indugula. 2005. Realistic Virtual Grasping. In *Proc. of IEEE VR*. 91–98.

[3] Doug A. Bowman and Larry F. Hodges. 1997. An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments. In *Proc. of I3D*. 35–ff.

[4] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. 2003. Gaze- vs. Hand-based Pointing in Virtual Environments. In *Proceedings of CHI Extended Abstracts*. 772–773.

[5] Catherine O. Fritz, Peter E. Morris, and Jennifer J. Richler. 2012. Effect size estimates: Current use, calculations, and interpretation. *Journal of Experimental Psychology: General* 141, 1 (2012), 2–18.

[6] Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: Appropriating the Body As an Input Surface. In *Proc. of CHI*. 453–462.

[7] Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology* 52 (1988), 139–183.

[8] Juan David Hincapié-Ramos, Kasim Ozacar, Pourang P. Irani, and Yoshifumi Kitamura. 2015. GyroWand: IMU-based Raycasting for Augmented Reality Head-Mounted Displays. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. 89–98.

[9] Ken Hinckley, Mike Sinclair, Erik Hanson, Richard Szeliski, and Matt Conway. 1999. The VideoMouse: A Camera-based Multi-degree-of-freedom Input Device. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology (UIST '99)*. 103–112.

[10] Teresa Hirzle, Jan Rixen, Jan Gugenheimer, and Enrico Rukzio. 2018. WatchVR: Exploring the Usage of a Smartwatch for Interaction in Mobile Virtual Reality. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18)*. ACM, Article LBW634, LBW634:1–LBW634:6 pages.

[11] Daniel Holz, Sebastian Ullrich, Marc Wolter, and Torsten Kuhlen. 2008. Multi-Contact Grasp Interaction for Virtual Environments. *JVRB - Journal of Virtual Reality and Broadcasting* 5(2008), 7 (2008).

[12] Jan Jacobs and Bernd Froehlich. 2011. A soft hand model for physically-based manipulation of virtual objects. In *IEEE VR*. 11–18.

[13] Jorge Jimenez, Diego Gutierrez, and Pedro Latorre. 2008. Gaze-based Interaction for Virtual Environments. *j-jucs* 14, 19 (2008), 3085–3098.

[14] T. Karitsuka and K. Sato. 2003. A wearable mixed reality with an on-board projector. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.* 321–322.

[15] Daniel Kharlamov, Brandon Woodard, Liudmila Tahai, and Krzysztof Pietroszek. 2016. TickTockRay: Smartwatch-based 3D Pointing for Smartphone-based Virtual Reality. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology*. ACM, 365–366.

[16] Volodymyr V. Kindratenko. 2000. A survey of electromagnetic position tracker calibration techniques. *Virtual Reality* 5, 3 (Sep 2000), 169–182.

[17] Tobias Langlotz, Elias Tappeiner, Stefanie Zollmann, Jonathan Ventura, and Holger Regenbrecht. 2018. Urban Pointing: Browsing Situated Media Using Accurate Pointing Interfaces. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, LBW604:1–LBW604:6.

[18] J.J. LaViola, E. Kruijff, D.A. Bowman, I.P. Poupyrev, and R.P. McMahan. 2017. *3D User Interfaces: Theory and Practice*.

[19] V. Lepetit, F.Moreno-Noguer, and P.Fua. 2009. EPnP: An Accurate O(n) Solution to the PnP Problem. *IJCV* 81, 2 (2009).

[20] Paul Lubos, Gerd Bruder, and Frank Steinicke. 2014. Analysis of Direct Selection in HMD Environments. In *IEEE 3DUI*. 11–18.

[21] Pranav Mistry and Pattie Maes. 2009. SixthSense: A Wearable Gestural Interface. In *ACM SIGGRAPH ASIA Sketches*. Article 11, 1 pages.

[22] Gerhard Reitmayr, Chris Chiu, Alexander Kusternig, Michael Kusternig, and Hannes Witzmann. 2005. iOrb - Unifying Command and 3D Input for Mobile Augmented Reality. In *Proc. IEEE Virtual Reality Workshop on New Diretions in 3D User Interfaces*.

[23] Michael Rohs. 2005. Real-World Interaction with Camera Phones. In *Ubiquitous Computing Systems*, Hitomi Murakami, Hideyuki Nakashima, Hideyuki Tokuda, and Michiaki Yasumura (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 74–89.

[24] Jeff Sauro and Joseph S. Dumas. 2009. Comparison of Three One-question, Post-task Usability Questionnaires. In *ACM CHI*. 1599–1608.

[25] Anthony Talvas, Maud Marchal, and Anatole Lécuyer. 2013. The god finger method for improving 3D interaction with virtual objects through simulation of contact area.. In *3DUI*. IEEE, 111–114.

[26] Vildan Tanriverdi and Robert J. K. Jacob. 2000. Interacting with Eye Movements in Virtual Environments. In *Proc. of CHI*. 265–272.

[27] Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. 2016. Efficient and Precise Interactive Hand Tracking Through Joint, Continuous Optimization of Pose and Correspondences. *ACM Trans. Graph.* 35, 4, Article 143 (July 2016), 12 pages.

[28] Robert J Teather and Wolfgang Stuerzlinger. 2014. Visual aids in 3D point selection experiments. In *Proc. of SUI*. 127–136.

[29] Lode Vanacken, Tovi Grossman, and Karin Coninx. 2007. Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments.. In *3DUI*. 27.

[30] Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller, and D'nardo Colucci. 1999. The HiBall Tracker: High-performance Wide-area Tracking for Virtual and Augmented Environments. In *Proceedings of ACM VRST*. 1–ff.

[31] Andrew D Wilson and Hrvoje Benko. 2010. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. In *Proc. of UIST'10*. 273–282.

[32] G. Yamamoto. 2007. A PALM Interface with Projector-Camera System. *UbiComp 2007 Adjunct Proceedings* (2007), 276–279.

[33] Xing-Dong Yang, Edward Mak, David McCallum, Pourang Irani, Xiang Cao, and Shahram Izadi. 2010. LensMouse: Augmenting the Mouse with an Interactive Touch Display. In *Proceedings of CHI'10*. 2431–2440.

[34] Thomas S. Young, Robert J. Teather, and I. Scott Mackenzie. 2017. An arm-mounted inertial controller for 6DOF input: Design and evaluation. *3DUI'17* (2017).