

3DTouch and HOMER-S: Intuitive Manipulation Techniques for One-Handed Handheld Augmented Reality

Annette Mossel¹

Benjamin Venditti¹

Hannes Kaufmann¹

¹Interactive Media Systems Group, Vienna University of Technology

Favoritenstr. 9-11/188/2, 1040 Vienna, Austria

mossel@ims.tuwien.ac.at

ABSTRACT

Existing interaction techniques for mobile AR often use the multi-touch capabilities of the device's display for object selection and manipulation. To provide full 3D manipulation by touch in an integral way, existing approaches use complex multi finger and hand gestures. However, they are difficult or impossible to use in one-handed handheld AR scenarios and their usage requires prior knowledge. Furthermore, a handheld's touch screen offers only two dimensions for interaction and limits manipulation to physical screen size. To overcome these problems, we present two novel intuitive six degree-of-freedom (6DOF) manipulation techniques, *3DTouch* and *HOMER-S*. While *3DTouch* uses only simple touch gestures and decomposes the degrees of freedom, *Homer-S* provides full 6DOF and is decoupled from screen input to overcome physical limitations. In a comprehensive user study, we explore performance, usability and accuracy of both techniques. Therefore, we compare *3DTouch* with *HOMER-S* in four different scenarios with varying transformation requirements. Our results reveal both techniques to be intuitive to translate and rotate objects. *HOMER-S* lacks accuracy compared to *3DTouch* but achieves significant performance increases in terms of speed for transformations addressing all 6DOF.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces | Interaction styles, I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism | Virtual Reality

General Terms

Design, Algorithms, Performance

Keywords

3D Interaction Techniques, 3D Manipulation, Handheld Augmented Reality

1. MOTIVATION & CONTRIBUTION

Recently emerged multi-core mobile processing units increase the 3D capabilities of mobile devices, i.e. multi-touch smartphones or tablets. State of the art mobile devices provide 3D rendering of virtual scenes in real-time, which are blended with the live image

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Laval Virtual VRIC '13, March 20-22, 2013, Laval, France.

Copyright 2013 978-1-4503-1875-4 ...\$10.00



Figure 1-1: Translate & rotate using HOMER-S

of the built-in mobile camera, so that the mobile device acts as a “window into the virtual world”. In hand-held video see-through Augmented Reality (AR), 3D interaction with a virtual scene is a key feature to explore the full potential of mobile AR. One of the primary interaction tasks is 3D object manipulation. However, the use of (multi) touch input for 3D object manipulation induces several problems. Since only one hand is available for interactions in typical one-handed handheld AR, only simple touch gestures of one hand for interaction are suitable. Furthermore, interaction space for touch interfaces is limited to the physical screen size and usability can suffer because users occlude the object with their fingers [1]. The implicit characteristics of a mobile touch screen provide only 2D data for manipulation. Hence, all three coordinate axes can never be addressed simultaneously at one moment in time. Therefore, multiple manipulation steps are required to rotate and translate the selected object in all three dimensions.

Our main motivation is to provide simple and intuitive 3D manipulation in one-handed handheld AR. To address the described problems of a multi-touch interface for 3D object manipulation, we propose *3DTouch*, which offers six-degree-of-freedom (6DOF) manipulation by combining simple 2D touch gestures with the handheld's current orientation. To overcome physical limitations of the device screen, object manipulation should be decoupled from touch screen input. Therefore, we present the novel *HOMER-S* technique, which needs only a single touch event for object selection and does not require any touch input for manipulation, but maps the handheld's position and orientation onto the selected object. Hence, our proposed techniques *3DTouch* and *HOMER-S* aim on improving intuitiveness and ease-of-use by reducing user input touch complexity and adapting real-world metaphors for virtual object transformation.

Both techniques are integrated in the Virtual Reality Framework ARTIFICe [2] that provides a mobile AR environment, which has been used for our thorough user study to evaluate performance and usability of both novel techniques. In summary, this paper presents the following two main contributions:

(1) Introducing the novel *3DTouch* manipulation technique that offers 6DOF manipulation. We propose splitting rotation, scaling and translation (RST) into single 3D manipulation entities by applying degree-of-freedom separation. By combining 2D touch information with the implicit given device orientation, manipulation of the remaining 3DOFs can be achieved, while necessary user touch gestures can be dramatically simplified. Thereby, *3DTouch* provides straightforward mapping between single and at most two-finger input to translate, rotate and scale (RST) an object along all three axes.

(2) Providing a full 6DOF manipulation technique by adapting the well-known *HOMER* [2] technique for smartphones (therefore *HOMER-S*) and tablets. It maps the physical 6DOF-pose of the smartphone onto the object to allow full 6DOF transformation (translate and rotate) as illustrated in Figure 1-1. The handheld's translation is further used for 3D scaling. Thus, manipulation is designed intuitively and interaction space for transformation control is not limited to the physical size of the mobile screen.

2. RELATED WORK

Manipulation such as translating or rotating an object is a universal interaction task in Bowman [3]. 3D manipulation using large multi-touch 2D displays in table top environments has gained interest, i.e. by Hancock [4]. However, methods for these large-scale environments are limited to the tabletop metaphor and hence not suitable for 3D manipulation using general-purpose multi-touch displays. On handhelds, recent approaches [5] [6] [7] [8] explore the capabilities of the user-facing camera for gesture-based manipulation using markerless finger and hand tracking. The position and orientation of the finger or hand are mapped to the virtual object for manipulation. However, finger tracking lacks accuracy compared to handheld pose estimation using natural feature tracking or the built-in inertial unit. Thus, we focus on related work designed for precise manipulation in VEs that can be adapted to handheld AR, as well as manipulation techniques using multi-touch input and additional built-in sensors.

2.1 Techniques for (immersive) VEs

As reported by Bowman [3], virtual hand techniques generally perform more effectively for object manipulation tasks in VEs than virtual pointing techniques, i.e. Ray-Casting [3]. Virtual Hand techniques such as VirtualHand [3], GoGo [9] or the hybrid technique HOMER [2] that uses Ray-Casting for object selection and VirtualHand for manipulation, map the physical transformation of the interaction device onto the selected object to transform it in 3D space. While Ray-Casting cannot move an object along the z-axis and only provides rotations around the cast ray, VirtualHand, GoGo and HOMER provide effective performance for translation. They can rotate the selected object around arbitrary axes by mapping the orientation of the user's interaction device onto the object. ArcBall provides precise rotation around any axis but does not offer object translation [3].

2.2 2D Multi-Touch Techniques

Touch input for 2D object transformation has become a de-facto standard on smartphones to transform objects in 2D [10]. The direct mapping between finger touches and 2D object

transformation is straightforward and hence easy to understand to users.

Various manipulation techniques have recently been designed for multi-touch displays. Hancock [4] presents one-, two- and three-touch input interaction techniques to manipulate 3D objects on any kind of multi-touch display. By using three touch interactions, simultaneous translation and rotation can be performed. This approach is limited to 5DOF and requires a large number of simultaneous touch inputs, which is not applicable to one-handed interaction on a smartphone. The Z-Technique [11] uses multi-touch input of two fingers and adjusts the depth position of the object by moving both fingers on the screen. This method requires prior knowledge of the specific two-finger gesture and does not provide rotation. To handle 3D manipulation, Reisman [12] integrates all the DOF and allows the user to directly manipulate 3D objects with three or more touch points. This approach takes perspective into account and is promising, but requires at least three points and mostly two hands for interaction input to access all 6DOF. Instead of integrating all 6DOF, Martinet [13] proposes to separate the 3D transformation into translation and rotation, resulting in a 3DOF problem using 2D touch input. The approach combines the Z-Technique [11] to control 3D position and [12] to control the orientation. Telkenaroglu and Capin [14] present several dual finger approaches for object translation along the z-axis and rotation around each axis separately. For each rotation, a separate dual finger gesture needs to be applied. Both techniques require prior knowledge of touch gestures, which reduces intuitiveness.

In addition to these multi-touch metaphors, Lee [15] and Guven [16] introduce manipulation metaphors particularly designed for handheld AR. Both approaches freeze the current real-world view for touch manipulation and aim on reducing faulty user input due to a shaky AR handheld environment. Cohé [17] uses multimodal input for 6DOF object manipulation. Translation is done via touch sliders, the handheld's inertial sensor data is mapped to the object for rotation. Scaling of the object is performed through pinch gestures with fingers. The inertial unit of the handheld is used by Hürst and Wezel in [6] to offer 3D translation in space. Rekimoto [18] investigates the use of the device's tilt as input for small screen interfaces to control menus, scroll bars and view point. This early work is promising and can be extended to work as a 3D object viewer, but does not offer full 6DOF manipulation control of an object. Henrysson [19] uses natural feature tracking to estimate the 6DOF device pose. The authors compare usage of keypad buttons with one-handed physical movement of a phone in order to move and rotate a selected object. The rotation of the selected object is chosen based on the orientation of the phone in space after the selected object has been released. Intuitive 3D rotation of an object was the main motivation of the work of Kratz and Rohs in [20]. This approach extends the virtual trackball metaphor by using a second phone as rear input device. This allows for accessing the full sphere to control 3D rotation using simple touch gestures. This work is very interesting, but does not offer translation and scale operations and requires a special hardware setup. Benzina [21] explores travel techniques for mobile virtual environments using touch input for viewpoint translation and the built-in sensors to control the viewpoint's orientation. Daiber [22] proposes an approach for sensor-based interaction with 3D data on a mobile device. It provides interaction techniques for gaming environments for translation and rotation using simultaneously touch input and the device orientation for object manipulation. The proposed rotation requires touching an object with a finger and then rotating the

device. Thereby, the object is fixed and the scene is rotate around it. This does not allow for intuitive manipulation. Since no detailed information about the rotation and translation algorithms and its limitation are given, this approach cannot be further evaluated in comparison to our proposed algorithms.

In contrast to related work, our proposed interaction techniques provide quick to familiarize 6DOF object manipulation using either straightforward one- or two finger gestures or no fingers but only the device pose for translation, rotation or scaling an object. This design allows for 3D manipulation in one-handed handheld AR.

3. MANIPULATION DESIGN

Both proposed methods are driven by the device pose, which is implicitly given in handheld AR. The handheld’s 6DOF comprises 3D position and 3D orientation, which can be estimated using a variety of tracking methods including inside-out optical tracking (i.e. of visual features) or utilization of the built-in mobile inertial measurement unit.

3.1 Design Guidelines

Since there is usually only one hand available in handheld AR, we developed *3DMultiTouch* and *Homer-S* for straightforward object manipulation. During the development of both techniques, we followed the following guidelines:

- (1) **Keep direct touch abilities:** The probably most appealing aspect of touch displays is the ability to directly “touch” an object in order to interact with. We aim on preserving this ability and do not introduce any offsets or non-direct gestures.
- (2) **Simplify input:** Since multi finger interaction requires prior knowledge for correct usage of the touch gesture and can be hard to apply with only one hand available, we aim to simplify touch gesture complexity for object manipulation. If necessary, we introduce degree-of-freedom separation to fulfill this guideline as well as mode switches to perform RST operations. Furthermore, we aim on adapting real world metaphors for intuitive object manipulation.

3.2 Object Selection

To select an object for manipulation, we use a mobile adaption of *Ray-Casting*. Therefore, Ray-Casting is triggered by a single touch event on the screen; these 2D screen coordinates are back-projected into 3D space and a virtual ray is cast from the virtual camera’s position in direction of the back-projected 3D point into the handheld AR scene. This direction can be estimated using the handheld’s device 6DOF pose that is implicitly given in handheld AR. The first object the ray hits is selected. Hence, object selection results in a simple single touch experience, which should be easily understandable for users. In the following sections, we refer to this method as *Mobile Ray-Casting*.

3.3 3DTouch Algorithm

Since direct mapping between finger touches and virtual touch points enables intuitive object transformation and is easy to understand, we preserve this direct relation in our *3DTouch* approach. Jacob [23] proposes DOF integration to match as closely as possible the structure of the manipulation task with the structure of the input device. But when applying 3D manipulation with a 2D multi-touch input device, it is hard to follow this conclusion. Recent work of Martinet [13] concludes that performance is improved if the interaction technique follows the

structure of the input device rather than the structure of the task. According to this work, we match the separated structure of the input device by proposing degree-of-freedom separation to split 6DOF transformations into single 3DOF entities. Since we are aiming on reducing touch gesture complexity to manipulate the remaining 3DOFs, we combine existing simple 2D multi-touch manipulation metaphors as in Hancock [10] with the current device pose. Inspired by Reisman [12], we transform the 2D screen coordinates of the touch input into 3D coordinates using the handheld’s pose to achieve non-uniform perspective 3D object manipulation, which relates to the distance between handheld and object upon selection. Applying this approach, the resulting transformation entities RST need either one (translate rotate) or two fingers (scale) input. We can reduce touch gesture complexity to a minimum while providing non-uniform 3D transformations.

3.3.1 Translations with 3DTouch

3D translations are performed using single touch input that is combined with the current device pose. Therefore, touch points from the current and last frame $p_{t1}, p_{t2} \in \mathbb{R}^2$ are back-projected from screen space into 3D space with respect to the distance $dist(mobile, object)$, which is calculated upon selection, resulting in 3D points $P_{t1}, P_{t2} \in \mathbb{R}^3$. Next, collinearity between $\vec{v}(P_{t1}, P_{t2})$ and the world coordinate system’s basis vectors are calculated to determine the current interaction plane used for the translation. Based on the resulting scalar s , the direction of translation is determined. Subsequently, the 3D position of the selected object is adjusted. In Figure 3-1, some examples of the resulting 3D translations using the proposed *3DTouch* algorithm are illustrated.

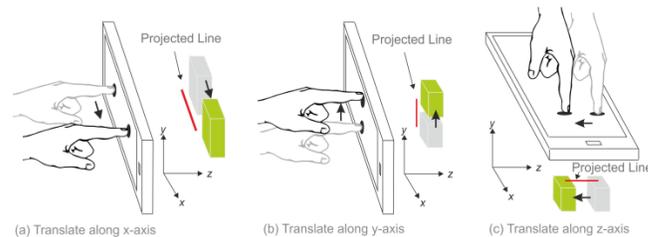


Figure 3-1: Examples of Translations using 3DTouch

Moving the finger up and down in Figure 3-1(a) would cause a translation along the y-axis. Analogously, moving the finger right or left in Figure 3-1(b), respective (c), results in translations along the x-axis.

3.3.2 Rotations with 3DTouch

Similar to translations, 3D Rotations are performed using single touch and the device pose. The algorithm is based on the proposed translation algorithm in Section 3.3.1. Before back-projecting the two touch points, a line perpendicular to $line(p_{t1}, p_{t2}) \in \mathbb{R}^2$ is calculated and subsequently the two new points are back-projected. To calculate the angle of rotation, we determine the factor

$$f_r = \frac{360 * s * |\vec{v}(P_{t1}, P_{t2})|}{U},$$

with s taken from section 3.3.1, indicating a positive or negative rotation. $|\vec{v}(P_{t1}, P_{t2})| \in \mathbb{R}^3$ regulates the angle as a fraction of the circumference U of the bounding sphere of the manipulated object. The factor f_r is then applied to the current rotation around the determined interaction direction in the object’s local coordinate system using collinearity evaluation.

In Figure 3-2, examples of the resulting 3D rotations using the proposed *3DTouch* algorithm are illustrated. Moving the finger up and down in Figure 3-2(a) would cause a rotation around the x-axis. Analogously, moving the finger right or left in Figure 3-2(c) results in a rotation around the z-axis. While translation and rotation in 3D are performed using single touch, *3DTouch* scales objects using two-finger touch input.

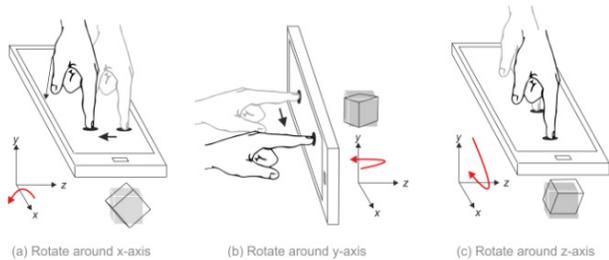


Figure 3-2: Examples of Rotations using 3DTouch

3.3.3 Scaling

The proposed scaling algorithm supports non-uniform scaling along each axis. Therefore, we apply a two finger pinch-like gesture and use an adapted version of the proposed translation algorithm from Section 3.3.1. The touch points of both fingers at two moments in time t_1, t_2 , are back-projected into 3D resulting in $P_{1,t_1}, \dots, P_{2,t_2} \in \mathbb{R}^3$ and used for collinearity estimation to determine the current interaction plane. Instead of calculating the sign, we determine whether the user moves both fingers together or apart and compute the length of movement. Then, the amount of scaling is calculated by $f_s = |P_{1,t_2} - P_{2,t_2}| - |P_{1,t_1} - P_{2,t_1}|$. For positive scaling, f_s is bigger than 0, for negative, the value is smaller than 0. f_s is then added to the current scale in the object's local coordinate system within the current interaction plane.

3.3.4 Mode Switches

Since *3DTouch* offers RST-transformations by decomposing each transformation into a single 3DOF entity, mode switches are required, which is realized through a simple button interface, as illustrated in Figure 3-3.

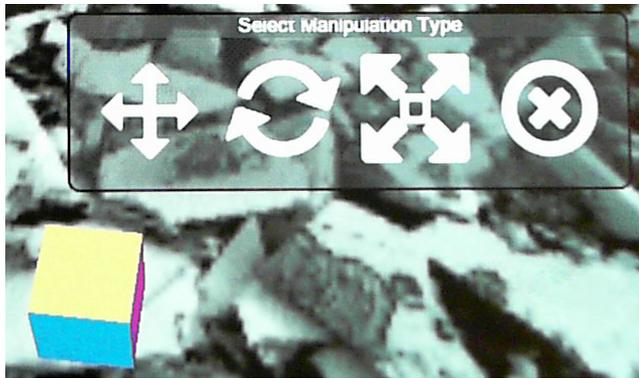


Figure 3-3: Floating GUI to select 3DTouch manipulation type

Compared to previous work from Reisman [12] and Telkenaroglu and Capin [14], this introduces an additional extra input modality. However, as reported in literature by Martinet [13] and Veit [24], separating the DOF of the task leads to better results than trying to use the separated DOF of a multi-touch display in an integral way, as demonstrated in [12].

3.4 HOMER-S Algorithm

The mapping between touch input and object manipulation of *3DTouch* is straightforward and simple. However, the touch abstraction layer still exists, and manipulation is limited to the screen size of the mobile device. Therefore, we introduce our novel *HOMER-S* technique, which integrates all 6DOF of a translation and rotation task by directly mapping the handheld's pose onto the object. Scaling is separated as a 3DOF tasks by reusing the device's position information. Thereby, we imitate real-world metaphors for translation, rotation and scaling, eliminate touch input during manipulation and extend the interaction space to the user's physical space.

Inspired by Henrysson [19] and using the immersive 3D manipulation technique HOMER as a baseline technique, we adapted HOMER to use it on smartphones and tablets (therefore *HOMER-S*) in a handheld AR environment. The original algorithm uses the 6DOF pose of the user's torso and that of the interaction device to manipulate an object. Since a handheld AR environment has different characteristics than an immersive VE, we adapted HOMER as illustrated in Figure 3-4.

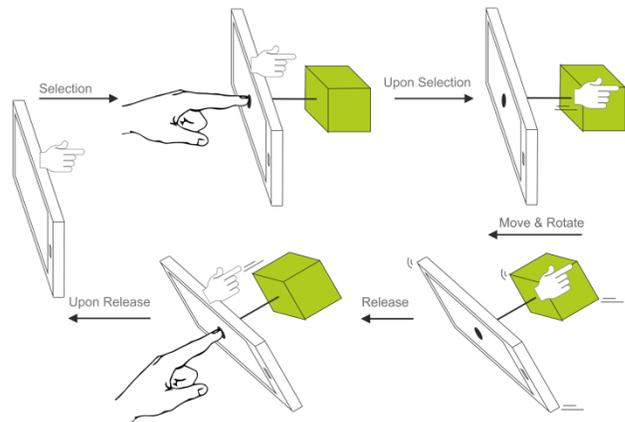


Figure 3-4: 6DOF Manipulation using HOMER-S

Rotations of the selected object around arbitrary axes are controlled independently. An isomorphic mapping between the handheld's orientation and the virtual hand is applied to rotate an object around the hit point that describes the pivot point. Thereby, the physical movement and rotation of the mobile device directly influences the transformation of the selected object.

By performing *Mobile Ray-Casting*, the object is released and the virtual hand moves back to the handheld's position. Our proposed *HOMER-S* algorithm is denoted as follows:

- (1) While no object O with position O_p and orientation O_o is selected, get handheld's position h_p and orientation h_o and map it to virtual hand's position, so that

$$vh_p = h_p, vh_o = h_o$$
- (2) Upon selection,

$$\text{Store current handheld's position } vh_{sel} = h_p$$
- (3) At each frame,
 - perform rotation, set $O_o = h_o$
 - perform translation,

$$\text{Set } vh_{curr} = h_p$$

$$\text{Calculate distance } d(vh_{curr}, vh_{sel})$$

$$\text{Normalize vector } \vec{v}(vh_{sel}, vh_{curr})$$

$$\text{Set } vh_p = vh_{sel} + d * \vec{v}_{norm}$$

To scale an object, we re-use the position information of the virtual hand. Therefore, at each frame we calculate the difference between the 3D positions of the previous and the current frame. The delta information is subsequently mapped onto the selected object. Thus, moving the virtual hand in positive direction of each axis will scale up; moving in negative will scale down the object. Thereby, we achieve non-uniform scaling along all axes.

As described, *HOMER-S* provides manipulation integrating all 6DOF for simultaneous translation and rotation and provides touch-less 3D scaling. Thereby, the touch abstraction layers that require prior knowledge is replaced with real-world metaphors. Additionally, the user's interaction space is no longer limited to physical screen size, but is expanded dramatically. It only depends on valid device tracking data and pose estimates.

3.4.1 Mode Switches

Compared to *3DTouch*, no degree-of-freedom separation needs to be performed when using *HOMER-S*, since it provides translation and rotation in an integral way. But if desired, translation and rotation can also be separated into single transformation entities. *HOMER-S* takes advantage of re-using real world metaphors to provide intuitive translation, rotation and scale. However, the metaphors for translating and scaling are akin in movement and hence are hard to distinguish when only using the device pose. Thus, we decided for conscious mode-switching between translate/rotate and scale rather than introducing a new and hence probably unknown real-world metaphor to handle scaling.

To summarize, *HOMER-S* provides the following 3D transformations: (1) translation, (2) rotation, (3) translation and rotation, and (4) scale. Through a button interface, mode switches between (1) and (4) and optional (2) and (3) are realized, as illustrated in Figure 3-5.

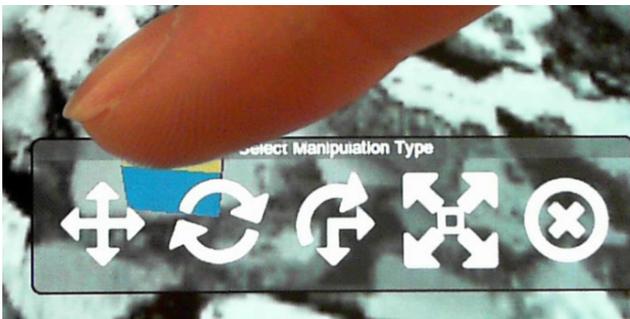


Figure 3-5: GUI of *HOMER-S* manipulation types upon selection

3.5 Assistance Design

To allow easier manipulation, spatial context as assistance information is provided to users while transforming a selected virtual model in space. Therefore, *3DTouch* and *HOMER-S* draw axes during translation and scaling as well as gimbals during rotation. To be able to distinguish between available interaction-planes, axes and gimbals are color coded as illustrated in Figure 1-1. Using *3DTouch*, those axes and gimbals are hidden, which are not accessible with respect to the current device pose.

3.6 Critical Aspects

The nature of the proposed manipulation techniques offers intuitive handling of 6DOF manipulation, but introduces some critical aspects as well. Since both methods are designed for handheld AR environments, a valid device pose is required. Thus, loss of tracking results in malfunction of object manipulation. But since there are several tracking methods for mobile devices

available, a complete loss of tracking can be eliminated by combining two or more tracking techniques. Future implementations could overcome the loss of natural feature tracking by using the built-in inertial unit for pose estimation for a short period of time. Freezing the AR view as in [15] or [16], to stabilize the device pose during interactions with *HOMER-S*, was neglected to allow for uninterrupted immersion.

Regarding rotations with *HOMER-S*, a drawback is caused by the direct mapping of the device orientation onto selected objects. Given the implicit binding of input- and output device in handheld AR, rotations around the pitch axis are limited. This is especially the case, when only using one marker for natural feature tracking, since the display is quickly no longer visible to the user. 360° rotations around the yaw axis can be applied by real world movements of the user, while rotations around the roll axis are straightforward and adapt the steering wheel metaphor. Further research will be conducted to evaluate the usability of a non 1:1 mapping to compensate rotation drawbacks of pitch and yaw rotations.

4. Performance Studies

For a comprehensive evaluation of our proposed manipulation techniques, we conducted a summative evaluation across four different scenarios with varying manipulation tasks.

4.1 Objectives

The main goal of the experiment was to evaluate the performance and usability of *3DTouch* and *HOMER-S*. Since *3DTouch* matches the separated structure of the multi-touch input device and *HOMER-S* adapts real-world metaphors, both techniques apply for straightforward manipulation. Hence, a second objective was to compare both techniques and to examine intuitive handling. In designing the experiment, we formulated the following hypothesis:

- [H1] *3DTouch* and *HOMER-S* are both designed to provide straightforward manipulation. Thus, both techniques will perform similar in terms of speed and ease-of-use for 3DOF manipulation tasks.
- [H2] Since *HOMER-S* offers full 6DOF transformation, it will perform considerably faster than *3DTouch* in compound translation and rotation manipulation tasks.
- [H3] Humans are able to control their single fingers more precisely than moving an object, which is attached to a device, freely in space. Thus, *3DTouch* performs better for manipulation tasks that require precise input.
- [H4] Regarding prior knowledge, users with experience using multi-touch devices will perform equally or better with *3DTouch* than with *HOMER-S*. Likewise, the design of *HOMER-S* enables better performance for users with no prior multi-touch knowledge.

4.2 Experimental Design and Procedure

We conducted the study using a within-subjects factorial design where the independent variables were manipulation technique and task scenario. In a second order evaluation, user experience was the third independent variable. The manipulation techniques were *3DTouch* and *HOMER-S*, while the scenarios included four different experimental tasks with varying manipulation types and combinations. The dependent variables were task completion time and number of interactions steps. Furthermore, we measured user preferences for both technique in terms of speed, accuracy, and usability.

At the beginning of the study, each participant was asked to read and sign a standard consent form as well as to complete a pre-questionnaire. We asked standard questions about age, gender and prior experience in 3D gaming and using multi-touch devices. Upon completion, the participant was given a detailed description of the practical part about “Manipulation in Handheld AR”. A tutor coached them on how to use the handheld device and how to perform manipulation in the testing environment. Afterwards, each participant had five minutes time to practice both manipulation techniques. Once they started the study, they were not interrupted or given any help. Upon completion of the practical part, they were asked to fill out a post-questionnaire (see Table 1). All 28 participants yielded successful simulation trials from which all data was used for analysis.

Table 1: Post-Questionnaire

Q1	How adequate do you feel the time allotted for practice was?
Q2	How comfortable were you with using a smartphone for task completion?
Q3	How would you rate the 3DTouch manipulation technique in usability? Speed? Accuracy?
Q4	How would you rate the HOMER-S manipulation technique in usability? Speed? Accuracy?
Q5	How would you rate intuitiveness of 3DTouch for 2D-translate, 3D-translate, rotate, move & rotate, scale an object?
Q6	How would you rate intuitiveness of HOMER-S for 2D-translate, 3D-translate, rotate, move & rotate, scale an object?
Q7	Which manipulation technique do you prefer to 2D-translate, 3D-translate, rotate, move & rotate, scale an object.
Q8	Rank the two manipulation techniques in order of desired use (with 1 being the most desired).
Q9	When determining how much you like using a manipulation technique, how important in influence on your decision was ease-of-use? Speed? Accuracy?

4.3 Subjects and Apparatus

Of the 28 participants ranging from 23 to 38 years, 12 were female and 16 male. 12 participants stated not to have any mobile 3D gaming experience at all, while 7 reported no experience with multi-touch smartphones. Table 2 given on overview of users based on their prior experience.

Table 2: Users grouped by prior experience

	inexperienced	experienced
a) Mobile 3D Gaming	12	16
b) Smartphone	7	21

The user study consisted of a pre-questionnaire followed by a practical test and a post-questionnaire. It took approximately 25 minutes for each participant to finish the user study. The practical test ran on a *Samsung Galaxy S II I9100*, featuring an Arm Cortex A9 Dual Core-Processor, a 4.27" WVGA multi-touch display and an 8 mega pixel camera. *Galaxy S II* weighs 116g and has the physical dimensions of 125.3 x 66.1 x 8.49 mm. We protected the phone with a market available hard cover to minimize the problem of cancelling the simulations by mistake by pushing the buttons on the sides.

4.4 Implementation

All computations – tracking, rendering, selection and manipulation of virtual objects – are performed on a smartphone using Android OS. For developing and testing the proposed interaction techniques, we used the Virtual and Augmented Reality Framework ARTIFICe [25]. It uses Vuforia [26] for tracking and pose estimation of the mobile device and Unity3D [27] game engine for 3D rendering and scene management. To access touch inputs on the mobile device screen, to trigger selection and release of scene objects, ARTIFICe uses Unity’s built-in Android interface to access the hardware layer. Pose data from Vuforia is processed by the specific interaction technique (IT) and handed to the ARTIFICe interaction framework for object manipulation. Using its interaction interface, we implemented *3DTouch* and *HOMER-S*.

4.5 Tested Scenarios

We built four different scenarios to cover different manipulation situations in 3D space. According to Bowman [3], we use the basic canonical manipulation tasks “position”, “rotation” and “scaling” for task design. Using these basic tasks as building blocks [28], we designed four tasks of varying complexity to allow for adequate simulation of manipulation in an AR application. To manually identify the desired object for subsequent manipulation, another canonical task “selection” is used. Since the selection task is performed by all users in the same way and is equally designed over all four tasks, the required time for selection does not influence the mean values of the performance metrics.

All scenarios are based on the same virtual working ground, a black & white textured plane that was printed to a 56x40cm sized piece of paper, which acted as a marker for the Vuforia framework. All 28 users completed all four scenarios in a random order. The participants could inspect the scenario, without being able to manipulate it, in order to understand the task according to its description before starting with the actual test.

4.5.1 Scenario 1 – Positioning on a Plane

The first task comprises the canonical task “positioning”. The user was challenged to translate a pink cube in the lower left corner to the center of a green area in the upper right corner. The distance between the targeted object and its destination was 35cm on the horizontal plane. It was sufficient to complete the task with the cube partly overlapping the designated target.

4.5.2 Scenario 2 – Positioning in 3D Space

The second task extends the first scenario by requiring positioning in all three dimensions. The user was challenged to translate a pink cube in the lower left corner on top of a small tower in the upper right corner (see Figure 4-1).



Figure 4-1: Scenario 2 "Move the pink cube on top of the green & blinking platform."

The distance between the targeted object and its destination was 35cm on the horizontal plane and 20cm vertically. The destination area was again a square. If it was partly overlapped by the target object, the task was completed.

4.5.3 Scenario 3 – Positioning & Rotation

For better simulation of manipulation requirements in AR applications, we applied an integral task design for the third scenario comprising a combination of “positioning” and “rotation”. The user was challenged to rotate a red barrel in the lower left corner by 45° around its vertical axis and translate it on top of an inclined plane (see Figure 4-2).

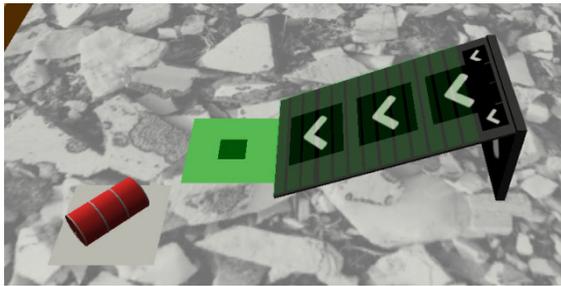


Figure 4-2: Scenario 3 "Let the barrel roll down the inclined plane onto the green & blinking platform."

From there the barrel was supposed to roll down the plane and over a square at its bottom. The test was successfully completed if the barrel was let loose on the top of the inclined plane rolling down its full length and at least partly hitting the center of the destination area.

4.5.4 Scenario 4 – Scaling & Positioning

A second integral task was designed for the fourth scenario. Here, the user was first challenged to scale a blue cube by a fifth in length and a third in width of its original size and then translate the cube into a glass positioned at the center of the scene (see Figure 4-3).



Figure 4-3: Scenario 4 "Resize and drop the blue cube into the glass."

The distance between the targeted object and its destination was 38cm horizontally and 10cm vertically. The destination was the circular shaped bottom of the glass. Users needed to let the cube fall into the glass from above and as soon as it hit the bottom, the task was completed.

4.6 Results

The quantitative data gathered from the questionnaires and automatically collected data of the test application were analyzed with Friedman’s χ^2 test and repeated measures single factor ANOVA accordingly. We focused on three different aspects during data analysis. First, data of all participants regarding manipulation techniques is evaluated. Second, we analyze the

techniques’ performance depending on tasks. Third, we look at data of selected participants - according to their experience listed in Table 2 - for each manipulation technique and task separately.

4.6.1 Performance Measures

To evaluate the quantitative data, “Task Completion Time” and “Number of Interaction Steps” were the main evaluation criteria. Task completion time represents the time it takes to successfully finish a specific scenario. Number of interaction steps comprises the amount of necessary mode switches to successfully finish an (integral) manipulation task.

4.6.2 Performance Evaluation

Analyzing the overall mean completion time, no significant difference was found between *HOMER-S* and *3DTouch* ($F_{1,27}=0.00299$, $p=0.957$) as illustrated in Figure 4-4.

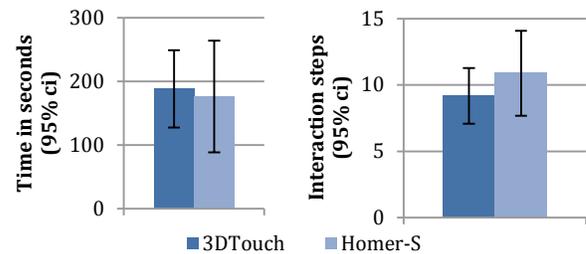


Figure 4-4: Mean completion time and number of interactions.

When inspecting the mean completion time for each task separately, again no significant differences could be found for both positioning tasks of scenario one and two at ($F_{1,27}=1.4$, $p=0.2468$) and ($F_{1,27}=0.814$, $p=0.375$) respectively. However, the Positioning & Rotation task was performed significantly faster with *HOMER-S* ($F_{1,27}=7.379$, $p<0.0114$). In contrast to that, *HOMER-S* took significantly more time to complete for the Scaling & Positioning task ($F_{1,27}=7.379$, $p<0.0114$) according to Figure 4-5.

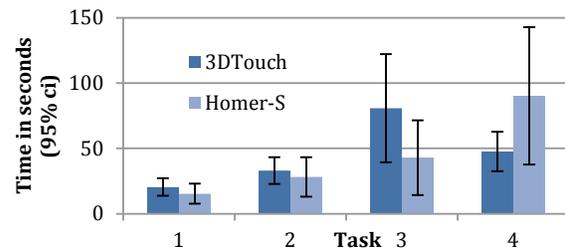


Figure 4-5: Mean completion time per task

Analyzing the task completion time, grouped by users’ knowledge according to Table 2 in section 4.2, revealed no further significant differences other than the overall ones illustrated in Figure 4-5. No significant differences could be found for both positioning tasks when analyzing the users’ experience. In the Positioning & Rotation task, the significantly better performance of *HOMER-S* was never independent of the users’ experience. The inexperienced users of the mobile gamer group (a) as well as of the smartphone group (b) performed significantly faster with *HOMER-S* than with *3DTouch*. The experienced users of both groups performed faster with *HOMER-S* as well, but not significantly. Furthermore, only the experienced groups of (a) and (b) had significant results for task 4, since they were significantly faster using *3DTouch*. No significant difference in performance between *3DTouch* and *HOMER-S* could be found for the inexperienced users of both groups in task 4.

The results of the evaluation for the overall mean number of interaction steps exposed that *3DTouch* enabled users to perform manipulations in significantly less steps than *HOMER-S* ($F_{1,27}=4.552$, $p<0.0421$), as illustrated in Figure 4-4. However, the evaluation per tasks found only a significant difference in the Positioning in 3D Space task ($F_{1,27}=4.374$, $p<0.046$) and in the Scaling & Positioning task at ($F_{1,27}=12.81$, $p<0.0013$), both in favor of *3DTouch*. Figure 4-6 indicates, that there was no significant difference for the Positioning on a Plane or the Positioning & Rotation task with both ($F_{1,27}=0.685$, $p<0.415$).

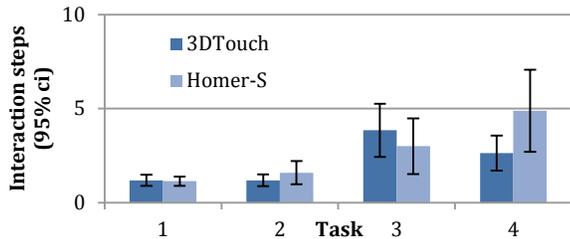


Figure 4-6: Mean number of interactions per task.

The evaluation of mean number of interaction steps, grouped by users' experience revealed, with one exception for task 3, no deviant results than those illustrated in Figure 4-6. The significantly better performance of *3DTouch* in the Positioning in 3D Space task could only be confirmed for the experienced users in a) and b). For the Positioning & Rotation task, the inexperienced group of a) achieved significantly better results with *HOMER-S* than with *3DTouch*. For all other groups no significance could be found for that task. For task 4, only the experienced users of both groups had significantly better results with *3DTouch* than with *HOMER-S*. No significant difference could be found for the inexperienced users of both groups.

4.6.3 Subjective Evaluation

When answering the questions Q1-Q6 and Q9, users were able to choose from a 7-point Likert scale. While all questions feature the highest rating at seven, and the lowest at one, Q1 states the best rating with four (appropriate).

Our participants found the time allotted for practice appropriate ($\mu=4$ and $\sigma=0.46$ at $\alpha=0.05$). Using a smartphone to complete the different tasks was rated to be moderately comfortable ($\mu=5.9$ and $\sigma=1.14$ at $\alpha=0.05$). As illustrated in Figure 4-7, the questions Q3 and Q4 revealed both to be average or good, but *3DTouch* was rated significantly better for ease-of-use and accuracy with ($\chi^2[n: 28]=6.55$, $p<0.0105$) and ($\chi^2[n: 28]=15.696$, $p<0.0000744$) respectively. In terms of speed, no difference was confirmable.

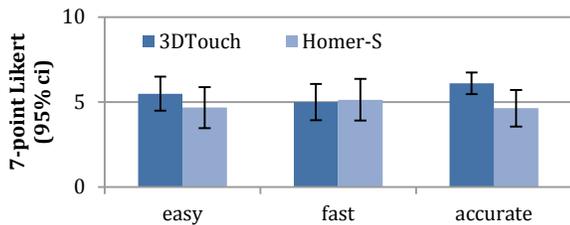


Figure 4-7: Users' average rating of Q3 & Q4.

Analyzing the subjective evaluation of ease-of-use, speed and accuracy, grouped by the user's experience, revealed significant better ratings of *3DTouch* in ease-of-use only of the experienced users of a) and b). *3DTouch*'s better rating for accuracy was independent of the users' experience in a) and b) except for

inexperienced users in b) where no significant difference occurred.

Users' ranking of the two interaction techniques indicated no significant preference (Q8) ($\chi^2[n: 28]=0.57$, $p=0.45$). Inspecting their preferences for specific transformations separately, a deeper insight could be gained. 2D- and 3D translation as well as rotation alone were not found to be significantly different, as shown in Figure 4-8. For the integral 6DOF manipulation of the Positioning & Rotation task, *HOMER-S* is significantly preferred with ($\chi^2[n: 28]=10.67$, $p<0.0011$). For scaling, *3DTouch* is significantly preferred with ($\chi^2[n: 28]=12.57$, $p<0.00039$). This subjective evaluation reflects the results of the quantitative evaluation in terms on completion time.

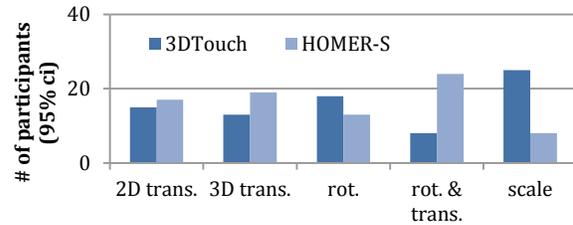


Figure 4-8: Users' preferences given Q7.

No deviant results for 2D- and 3D-translation as well as rotation alone were revealed, when analyzing the ranking of each manipulation, grouped by the users' experience. Looking in detail at users' preference of both groups for integral rotation and translation tasks, *HOMER-S* was significantly preferred by the experienced users. Also the inexperienced users preferred *HOMER-S*, but not significantly. *3DTouch*'s preference for scaling remains independent of user's experience in both groups.

Question Q9 inquiring the users' influence on their decision for questions Q3 and Q4 yields with ($\chi^2_2[n: 28]=3.89$, $p<0.143$) no significant difference for the three options ease-of-use, speed and accuracy. Users stated all aspects of Q9 similarly important, ranging from $\mu=5.5$ (slightly important) to $\mu=6.18$ (important).

5. DISCUSSION

We designed the experiment to compare two different techniques for performing 3D manipulation tasks with a multi-touch handheld device. While *3DTouch* separates the DOFs of the task to improve performance as shown in previous work [13], *HOMER-S* controls 6DOF in an integral way and takes advantage of simulating real-world metaphors.

Results show that for both techniques, no significant difference was found for overall mean task completion time, completion time for the positioning tasks, overall user preference or user preferences regarding the positioning tasks. These results support hypothesis **H1** and confirm the straightforward, intuitive design of both interaction techniques when performing canonical tasks. Inspecting performance and user's preference for compound canonical tasks, two findings can be stated. First, for 6DOF manipulation tasks, as simulated in the Positioning & Rotation scenario, *HOMER-S* performed significantly faster than *3DTouch*. This quantitative evaluation is supported by the user's subjective feedback. *HOMER-S* is significantly preferred for translation and rotation tasks by users as expressed in Q7. These findings support **H2** and indicate the strength of the integral design of *HOMER-S* for compound canonical 6DOF tasks. This is also reflected by users' comments who described *HOMER-S* to be natural, of "more direct contact" and fun. Thus, these real world metaphors tend to be very intuitive and straightforward. The second finding

when inspecting performance and user's preference for composite manipulation tasks reveals the strength of *3DTouch* for scaling tasks. It took considerably less time to complete the Scaling & Positioning task using *3DTouch* than with *HOMER-S*. Furthermore, users significantly preferred *3DTouch* for scaling. Since no significant difference was found regarding the positioning tasks in completion time or user preferences, positioning can be neglected when evaluating scenario 4. This finding supports **H3**, since the scaling tasks required very fine manipulation in all three dimensions. H3 can further be backed up by the significant fewer number of interaction steps *3DTouch* needed in tasks 2 & 4 and the users' rating in Q3 & Q4 attested it a better accuracy.

Besides the assumption, that humans are able to control their fingers more precisely, the underlying metaphor can be another conceivable reason to further explain the underperformance of *HOMER-S* in scaling tasks. In the real world, usually two hands are involved to expand or shrink an object. Since *HOMER-S* only provides one virtual hand to simulate one real hand, this metaphor could not be adapted in a direct way. Thereby, a direct mapping could not be provided that limits *HOMER-S* straightforward usage for scaling. However, the pinch-like gesture to scale an object using *3DTouch* is also not completely intuitive and straightforward. Since, more than half of our test group classified themselves as experienced mobile 3D gamers, they are familiar with using multi-touch for interaction; standard touch gestures such as the pinch-out and -in are known and well trained. This is also backed up by the results including user experience. There, *3DTouch* results for scaling are only significantly better for users who are experienced with smartphones or mobile 3D gaming. Studying further details regarding user experience leads to **H4**. We proposed that prior touch knowledge would result in equal or better performance of *3DTouch* compared to *HOMER-S*, while inexperienced users would perform better with *HOMER-S* due to its integral 6DOF design and adaption of real-world metaphors. For many results of the study, this is true. Regarding completion time, no significant differences between *3DTouch* and *HOMER-S* could be found for positioning when analyzing experienced users. For 3D positioning, experienced users needed significantly less interaction steps when using *3DTouch*. For integral positioning and rotation, experienced users of both groups performed faster with *HOMER-S*, but not significantly. Experienced users performed significantly faster for scaling in terms of completion time and number of interaction steps when using *3DTouch*. They rated *3DTouch* significantly better in terms of ease-of-use, but significantly preferred *HOMER-S* for 6DOF transformation. Regarding inexperienced users, H4 can be further backed up by the significant better performance in terms of completion time and number of interaction steps for the Positioning & Rotation task using *HOMER-S*. Users' comments reflect the quantitative results. Most users, especially the inexperienced, reported to have quickly familiarized with *HOMER-S* for any translations and rotations. However, exceptions when evaluating H4 could be found, too. The quantitative results do not indicate a better performance of inexperienced users using *HOMER-S* for positioning tasks. For scaling, *HOMER-S* did not result in better performance of the inexperienced users. However, despite of the good results of *3DTouch* for scaling, inexperienced users did not significantly perform better using *3DTouch* for scaling. The underlying two-fingers pinch gesture requires prior knowledge and thus, is not as straightforward and direct than the one-finger inputs for translate and rotate. But users' preference of *3DTouch*'s for scaling is independent of the users' experience. This is also reflected by

users' comments. Some users experienced *HOMER-S* as being "too direct", since even small body movements result in a manipulation. Most users complained about *HOMER-S* being unintuitive to use for scaling. Based on these observations, we cannot draw a clear conclusion to support H4. Further research needs to be performed for a detailed evaluation of this hypothesis.

Taking the results and findings into account, we have developed a set of preliminary guidelines:

- Both methods provide intuitive manipulation with similar performance when the canonical manipulation tasks "positioning" and "rotations" are required.
- *HOMER-S* outpaces *3DTouch* in performance and ease-of-use to perform compound positioning and rotation tasks.
- *3DTouch* is the better choice, if scaling is involved in the manipulation task.

6. CONCLUSION & FUTURE WORK

In this work, we provide a foundation for exploring 3D manipulation techniques in one-handed handheld AR environments by evaluating two manipulation techniques. The main motivation of our work was the development of intuitive manipulation techniques for 3D content in handheld AR. Therefore, we aimed at reducing or even eliminating (multi) touch input to avoid abstraction levels, which require prior knowledge.

We introduced a novel multi-touch manipulation technique *3DTouch*, which provides 6DOF transformations by separating degree-of-freedom and combining one- and two-finger touch input with the device pose. Thereby, we could reduce necessary input to a minimum. This improves usability for a one-handed handheld AR environment by utilizing only simple finger gestures; single touches for translation and rotation and two-finger input for scaling. Additionally, we presented the manipulation technique *HOMER-S* that completely decouples object transformation from limited mobile screen space. It does not require any touch input, but uses only the device pose to provide full 6DOF transformations.

The evaluation study clearly reveals the strengths of both techniques depending on given manipulation tasks. Based on promising results of both techniques, we will examine the performance and usability of a combination of *3DTouch* and *HOMER-S*. Our findings on fine manipulation tasks motivate us to further evaluate the capabilities of both techniques. Additionally, we plan to optimize the overall usability of our *HOMER-S* manipulation technique to further exploit its potential. Our focus will be to improve pose stability during manipulation and to improve the gesture for scaling. To overcome stated rotation limitations using the *HOMER-S* approach, a non-direct mapping between device pose and object's rotation will be evaluated. Furthermore, we think about using a tracking approach that provides 360° robust tracking in the whole surrounding of the mobile device, i.e. combining robust inertial tracking with natural feature tracking or using a SLAM-based localization approach. Thereby, all manipulation transformations can be robustly mapped to objects with *HOMER-S* in all device poses.

7. REFERENCES

- [1] J. S. Pierce, A. Forsberg, M. J. Conway, S. Hong, R. Zeleznik, and M. Mine, "Image Plane Interaction Techniques in 3D Immersive Environments," in *Proceedings of the 1997 Symposium on Interactive 3D Graphics (I3D '97)*, 1997, pp. 39–43.

- [2] D. A. Bowman and L. F. Hodges, "An Evaluation of Techniques for Grabbing and Manipulating Objects in Immersive Virtual Environments Arm-Extension Ray-Casting," in *Proceedings of the 1997 symposium on Interactive 3D graphics*, 1997, pp. 35–38.
- [3] D. Bowman, E. Kruijff, J. J. LaViola Jr., and I. Poupyrev, *3D User Interfaces: Theory and Practice*. Addison-Wesley, 2005.
- [4] M. Hancock, S. Carpendale, and A. Cockburn, "Shallow-Depth 3D Interaction : Design and Evaluation of One-, Two- and Three-Touch Techniques," in *ACM SIGGRAPH 2010 Emerging Technologies on - SIGGRAPH '10*, 2007, pp. 1147–1156.
- [5] W. Hürst and C. Van Wezel, "Multimodal interaction concepts for mobile augmented reality applications," *Advances in Multimedia Modeling*, pp. 157–167, 2011.
- [6] W. Hürst and C. Wezel, "Gesture-based interaction via finger tracking for mobile augmented reality," in *Multimedia Tools and Applications*, 2012.
- [7] T. Lee and T. Hollerer, "Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking," in *2007 11th IEEE International Symposium on Wearable Computers*, 2007, pp. 1–8.
- [8] T. Niikura, Y. Hirobe, and A. Cassinelli, "In-air typing interface for mobile devices with vibration feedback," in *ACM SIGGRAPH 2010 Emerging Technologies*, 2010, pp. 1–1.
- [9] I. Poupyrev and M. Billinghurst, "The go-go interaction technique: non-linear mapping for direct manipulation in VR," in *Proceedings of the 9th annual ACM symposium on User interface software and technology*, 1996, pp. 79–80.
- [10] M. S. Hancock, S. Carpendale, F. D. Vernier, and D. Wigdor, "Rotation and Translation Mechanisms for Tabletop Interaction," in *First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)*, 2006, pp. 79–88.
- [11] A. Martinet, G. Casiez, and L. Grisoni, "The design and evaluation of 3D positioning techniques for multi-touch displays," *2010 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 115–118, Mar. 2010.
- [12] J. L. Reisman, P. L. Davidson, and J. Y. Han, "A screen-space formulation for 2D and 3D direct manipulation," in *Proceedings of the 22nd annual ACM symposium on User interface software and technology - UIST '09*, 2009, p. 69.
- [13] A. Martinet, G. Casiez, and L. Grisoni, "Integrality and separability of multitouch interaction techniques in 3D manipulation tasks.," *IEEE transactions on visualization and computer graphics*, vol. 18, no. 3, pp. 369–80, Mar. 2012.
- [14] C. Telkenaroglu and T. Capin, "Dual-Finger 3D Interaction Techniques for mobile devices," *Personal and Ubiquitous Computing*, Sep. 2012.
- [15] G. Lee, U. Yang, Y. Kim, D. Jo, and K. Kim, "Freeze-Set-Go interaction method for handheld mobile augmented reality environments," in *VRST '09 Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, 2009, pp. 143–146.
- [16] S. Guven, S. Feiner, and O. Oda, "Mobile augmented reality interaction techniques for authoring situated media on-site," in *IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2006, pp. 235–236.
- [17] A. Cohé, D. Fabrice, and M. Hachet, "tBox : A 3D Transformation Widget designed for Touch-Screens," in *Proceedings of the 2011 annual conference on Human Factors in Computing Systems*, 2011, pp. 3005–3008.
- [18] J. Rekimoto, "Tilting operations for small screen interfaces," in *Proceedings of the 9th annual ACM symposium on User interface software and technology*, 1996, pp. 167–168.
- [19] A. Henrysson, M. Billinghurst, and M. Ollila, "Virtual object manipulation using a mobile phone," in *Proceedings of the 2005 international conference on Augmented tele-existence - ICAT '05*, 2005, p. 164.
- [20] S. Kratz and M. Rohs, "Extending the virtual trackball metaphor to rear touch input," in *2010 IEEE Symposium on 3D User Interfaces (3DUI)*, 2010, pp. 111–114.
- [21] A. Benzina, M. Toennis, G. Klinker, and M. Ashry, "Phone-based motion control in VR: analysis of degrees of freedom," in *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*, 2011, pp. 1519–1524.
- [22] F. Daiber, L. Li, and A. Krüger, "Designing Gestures for Mobile 3D Gaming," in *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia - MUM '12*, 2012, p. 3.
- [23] R. J. K. Jacob, L. E. Sibert, D. C. Mcfarlane, and M. P. Mullen, "Integrality and Separability of Input Devices," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 1, no. 1, pp. 3–26, 1994.
- [24] M. Veit, "Influence of Degrees of Freedom's Manipulation on Performances During Orientation Tasks in Virtual Reality Environments," in *VRST '09 Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, 2009, vol. 1, no. 212, pp. 51–58.
- [25] A. Mossel, C. Schönauer, G. Gerstweiler, and H. Kaufmann, "ARTiFICE-Augmented Reality Framework for Distributed Collaboration," *To appear in International Journal of Virtual Reality*, 2013.
- [26] Qualcomm, "Vuforia," 2012. [Online]. Available: <http://www.qualcomm.com/solutions/augmented-reality>.
- [27] Unity Technologies, "Unity3D," 2012. [Online]. Available: <http://www.unity3d.com/>.
- [28] M. E. Mündel, *Motion and Time Study: Improving Productivity*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc, 1978.