Freeman, E., Williamson, J., Subramanian, S. and Brewster, S. (2018) Point-and-Shake: Selecting from Levitating Object Displays. In: 2018 CHI Conference on Human Factors in Computing Systems, Montréal, QC, Canada, 21-26 Apr 2018, p. 18. ISBN 9781450356206.

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

© The Authors 2018. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in the 2018 CHI Conference on Human Factors in Computing Systems, Montréal, QC, Canada, 21-26 Apr 2018, p. 18. ISBN 9781450356206, https://doi.org/10.1145/3173574.3173592.

http://eprints.gla.ac.uk/156185/

Deposited on: 25 January 2018

Enlighten – Research publications by members of the University of Glasgow
http://eprints.gla.ac.uk

Point-and-Shake: Selecting from Levitating Object Displays

Euan Freeman[1], **Julie Williamson**[1], **Sriram Subramanian**[2], **and Stephen Brewster**[1]
Glasgow Interactive Systems Section
University of Glasgow
Glasgow, United Kingdom
{first.last}@glasgow.ac.uk
Interact Lab
University of Sussex
Brighton, United Kingdom
{first}@sussex.ac.uk

ABSTRACT
Acoustic levitation enables a radical new type of human-computer interface composed of small levitating objects. For the first time, we investigate the selection of such objects, an important part of interaction with a levitating object display. We present *Point-and-Shake*, a mid-air pointing interaction for selecting levitating objects, with feedback given through object movement. We describe the implementation of this technique and present two user studies that evaluate it. The first study found that users could accurately (96%) and quickly (4.1s) select objects by pointing at them. The second study found that users were able to accurately (95%) and quickly (3s) select occluded objects. These results show that *Point-and-Shake* is an effective way of initiating interaction with levitating object displays.

Author Keywords
Acoustic Levitation; Pointing; Ray-Cast Pointing; Selection.

ACM Classification Keywords
H.5.2. User Interfaces: Input devices and strategies.

INTRODUCTION
Levitation enables new forms of computer interface, where the content that users interact with is composed of physical objects levitating in mid-air, rather than pixels on a screen. This novel type of display has advantages: the interactive content is physical, leveraging spatial understanding, yet dynamic, unlike other physical media; users can 'see through' the display and its content, which can enhance collaborative interactions in groups; users can view the content from many angles and change perspective simply by moving around the display.

These properties make levitating object displays suitable for interfaces where users view and manipulate 3D content. For example, design and modelling software could compose a physical representation of a model using a 'point cloud' of small levitating objects, and users could then manipulate the control point objects to change the shape. This would allow users to experience the models in ways not possible on screen and could be useful for viewing prototypes before fabrication. As another example, visualisation software could construct physical representations of data using levitating objects, which users could then interact with and query, by selecting a data point to learn more about it. Physical visualisation of data in this way may improve information retrieval from 3D datasets, compared to visualisations on a screen [11].

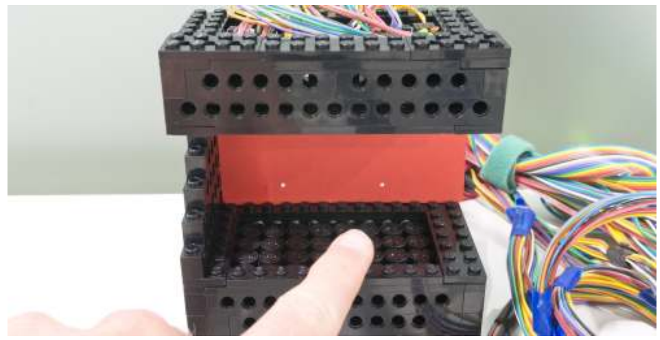

Figure 1. Our interaction technique allows users to select a levitating object by pointing at it. The selected object shakes to give feedback.

Research into the technical challenges of levitation has led to novel methods involving magnets [13], airflow [1, 23] and acoustics [14, 15, 17], moving us closer to applications like those described earlier. Acoustic levitation, in particular, has benefited from recent work [14, 15, 17, 20] and many objects can now be levitated and moved independently. Research is now required to establish the interaction techniques needed to manipulate such displays. Basic interactions, like the selection of a single object, are unexplored, but nuances of levitation mean that existing input techniques cannot simply be applied without consideration. In this paper, we describe interaction with levitating objects and develop a technique for selection. Selection is fundamental, happening before other actions can take place [21], like moving an object or querying it, so this is a key step towards richer interaction with levitating objects.

Selecting a levitating object by physically touching it is not possible because physical contact may knock the object out of the air. Close proximity to the levitation may also disrupt the acoustic field or airflow, for those approaches, affecting the position of the objects. This means a non-contact interaction technique is necessary. In this paper, we describe a mid-air gesture technique called *Point-and-Shake*, which allows users to select a levitating object by pointing a finger at it (as in Figure 1). We chose a pointing gesture because it allows immediate use, supports seamless transition between users in a group, and allows direct selection. Feedback is necessary

Accepted for publication at CHI 2018

to show which object is being selected (e.g., when pointing at two objects close together) so we also investigate feedback based on object movement: the objects shake from side to side when being selected. Since only the position of the object is manipulated, no other output modalities are necessary.

We present two studies in the design of *Point-and-Shake*. First, we evaluated our pointing interaction with an acoustic levitation system to see how easily users can select a levitating object by pointing at it. We then extended the technique to allow the selection of occluded objects (i.e., when one object is behind another), which we evaluate in our second study. This technique was designed with the capabilities and nuances of object levitation in mind. Our studies found *Point-and-Shake* to be a successful way of selecting objects, a first step towards interaction with levitating object displays.

RELATED WORK

User interfaces based on object levitation are one realisation of Ishii's *radical atoms* vision [10], a vision of physical materials that can change form and appearance dynamically, to allow new and richer ways of interacting with information. Small levitating objects can be the 'atoms' from which physical objects are composed; these composite objects exist in mid-air and can change form dynamically, by rearranging the 'atoms'.

This vision has inspired the development of approaches for levitating objects and manipulating them in mid-air. *ZeroN* [13] used magnetic levitation to levitate a single large sphere. While limited to one 'atom', *ZeroN* supported rich visual output by projecting onto the surface of the sphere. *Floatio* [23] and *Aerial Tunes* [1] used airflow to suspend and move objects in mid-air, using air jets to levitate the objects. This approach offered limited control over object position, but was scalable to levitate multiple objects. *Pixie Dust* [15] used acoustics to levitate objects: arrays of ultrasound transducers created an inaudible sound field that 'trapped' the objects. This supported larger numbers of smaller objects (⌀1–5mm) positioned close together (at least 4.5mm apart). They used objects to form 2D graphics in a flat plane, and also projected onto the objects, using the dense layout to form a 'screen' in mid-air.

The higher resolution of levitated objects allowed by acoustic levitation means it has been more actively pursued in recent research. The state-of-the-art has recently been advanced to allow multiple levitated objects to be moved independently of each other [17], to allow objects to be rotated [14, 20], and to allow levitation over a single array [14]. We use an acoustic levitation system in our research because the ability to position and move multiple objects allows for richer interactions than the other methods currently do, although the interactions we develop are not limited to this approach.

The hardware used for acoustic levitation has other uses in HCI. Phased ultrasound arrays have been used to generate a variety of mid-air haptic sensations (e.g., [2, 7, 8, 9, 12]), allowing users to experience tactile feedback from gesture systems. They have also been used to direct audio (e.g., *Holographic Whisper* [16]), to create the illusion of sound coming from remote objects or to direct audio feedback towards specific users. These applications use 40 kHz ultrasound like acoustic levitation systems do, and they are also based on similar acoustic principles. In future, acoustic levitation systems could be enhanced with mid-air haptic and steerable audio feedback to create rich multimodal interfaces. However, it would be necessary to minimise interference between the different audio fields (e.g., so the haptic feedback does not affect the levitation) so advances in acoustic field synthesis are necessary before this becomes feasible.

Few applications using levitating objects have been explored, since the work has mainly focused on the technical challenges of levitation. A small body of work has investigated physical visualisations (visualisations that map data to physical form [11]) composed of levitating objects. This has advantages over other physical visualisation media (e.g., those that are 3D printed or hand-crafted) because they can be updated and animated dynamically, since their form is changeable. *LeviPath* [17] was motivated by physical visualisation and used acoustic levitation to position and move small spherical objects) independently. *Floating Charts* [18] also used acoustic levitation to visualise chart data. For example, spheres represented the data points in a bar chart. The physicalisations were enhanced by using coloured balls, joining the balls by threads, and placing acoustically-transparent objects in the levitation space. In other work, we demonstrated the use of levitating objects to create user interface controls [3], but this has not been properly studied yet.

Interaction with the systems discussed has been limited to moving a single levitating object, either by grasping it [13, 23] or mapping it to hand position [15, 17]. 'Hands on' grasping interactions are not possible with acoustic levitation and are also unsuitable for levitations with several mid-air objects (e.g., the many data points in *Floating Charts* [18]), because the levitated objects may be too small to grasp and position correctly, they may be out of reach, or they may be inaccessible behind other objects in the physicalisation. Existing mid-air interactions [15, 17] have demonstrated how a single object might be translated with hand movements, but have not addressed how to create complete and usable interactions with such levitated objects (e.g., initiating interaction, selecting a target object, and ending the interaction). In this paper, we investigate object selection as a fundamental part of interaction with levitating objects and we demonstrate the first complete interaction technique with a levitating object display.

Non-contact 3D selection

Selection is a primary interaction for 3D interfaces [21] and is often required before other actions can occur. Users cannot manipulate mid-air objects by hand because some levitation methods (notably acoustics) are not strong enough to keep objects in place while being touched, meaning remote selection is necessary. Mid-air gestures could be used for selection and we take this approach because it has advantages for levitating object displays, as discussed before. We take inspiration from mid-air selection techniques for volumetric displays (e.g., [5, 6, 21]), as these have similarities to levitating object displays. However, a key difference is that volumetric displays allow rich visual feedback to support complex interaction techniques.

Levitating objects allow less visual feedback so interactions must be designed with such limitations in mind.

Two of the most popular selection gestures for 3D content are pointing and hand extension (‘virtual hand’) [5]. Hand extension gestures (e.g., the selection method used by Xbox Kinect) map a virtual cursor position to the position of the hand [19, 24]. This technique relies on visual feedback that shows users the relationship between the on-screen cursor and hand position. This is not ideal for levitating object displays, because the cursor would have to be shown in situ, using another levitating object. This would affect object spacing: objects would need to be further apart, so the cursor could move around them. This could also lead to selection ambiguity; the cursor object cannot be moved right next to a target object so would be separated like any other object. This means the cursor could end up a similar distance to multiple targets.

Pointing gestures, however, allow users to select “that one there” with less need for feedback, since the spatial relationship between finger and target is less ambiguous than with hand extension; feedback only needs to show which object is targeted. Comparisons of 3D selection techniques have also favoured pointing [5], therefore we used it as the basis for our selection technique.

Pointing techniques are generally based on ray-casting, where a ray extends from the finger towards the targets [6]. This is easy for users to understand, although it can lead to difficulty when there is a high object density. Objects may be positioned behind other objects, meaning the ray always targets the one in front. Grossman *et al.* [5] developed ray-cast techniques for “selection disambiguation”, allowing users to select occluded objects. Their techniques were based on the concept of a “depth cursor” that can be positioned along the ray to control which object is being targeted. Two implementations of this, *Depth Ray* and *Lock Ray*, were particularly successful and we discuss these later, when we describe our own selection technique. Vanacken *et al.* [21] also investigated *Depth Ray*, comparing it with a new technique and enhancing them with non-visual feedback about selection.

These pointing techniques [5, 21] rely on detailed visual feedback to show cursor position in 3D space, which is difficult in a system where the only visible things are the objects being levitated. This imposes a constraint for design: any necessary feedback must be given by the levitating objects themselves. Two ways of doing this are to manipulate an object’s position or appearance. *ZeroN* [13] used projectors to augment objects and *JOLED* [20] rotated the objects, which were coloured differently on each side. Instead, we take the first approach: manipulating the object position. This is better for the small objects used in acoustic levitation (versus the larger *ZeroN*) and does not require enhancement of the levitation system.

In summary, our work investigates the selection of acoustically-levitated objects. Acoustic levitation is a new area of research and the current state of the art shows the potential to achieve the example applications we presented earlier. There has been very little work on interaction techniques for these types of display. We focus on object selection, a fundamental interaction necessary for others to take place. We now describe a technique, called *Point-and-Shake*, that allows users to select levitating objects by pointing at them in mid-air.

POINT-AND-SHAKE: SELECTING LEVITATING OBJECTS

Point: Targeting mid-air objects

Users target objects by pointing at them with an extended finger (Figure 2). Rather than select targets instantly when pointed at, we use ‘dwell’ to confirm selection: the selection is complete after pointing at the target for 1000ms. Confirming the selection is necessary because users may inadvertently point at other objects before targeting the intended one, which would cause unintended selection. There are other ways of confirming selections, including clutched gestures, gestures with the other hand, and the use of buttons or foot switches. We chose dwell instead of these because of its simplicity: users just need to point to select, and the system can give simple feedback about the object being targeted. We chose a 1000ms dwell as a starting point, as other pointing gestures used this period successfully [4], citing a balance between giving users enough time whilst not rushing them.

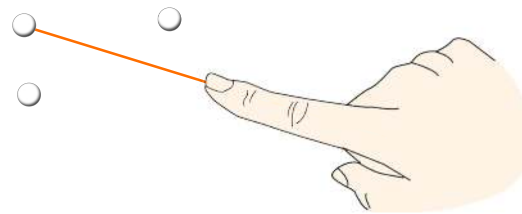

Figure 2. Users target an object by pointing at it. The line shows the ray extending in the same direction as the finger, towards the target.

Our system (described later) levitates small spherical objects (⌀1–2mm). These present a very small target for users to point at, which is likely to make selection difficult. To make things easier, we enhance the target size using virtual spheres centred around the object position (Figure 3). Users can select a levitating object by pointing at its sphere, without having to point precisely at the object. We use spheres, rather than circles, since users can point at a levitating object display from many angles. The virtual targets are not visible and users do not need to be aware of them to interact.

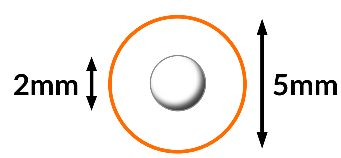

Figure 3. Virtual spheres increase the effective target volume, making it easier to select the small objects. This example shows the increased target volume of a ⌀5mm sphere, versus the ⌀2mm levitating object.

When there is a small number of levitating objects, users should be able to easily select each of them by pointing. In more dense levitations, like the 3D model scenario we envisioned in the introduction, it is likely that levitating objects will be difficult to target because of object occlusion. As Figure 4 shows, the ray used for targeting may intersect several objects, leading to ambiguity about which one should be selected. A naive implementation would choose the closest object to the finger, meaning users would be unable to select and interact with the distant objects. A novel aspect of a levitating object display is that users can move around it and view content from many angles. This could partly mitigate the occlusion problem, but moving may not always be an option; for example, if there are multiple users around the levitation, if the input sensor has limited range (e.g., the Leap Motion we used), or if the user is manipulating a specific part of the content. Being able to resolve occlusion would allow users to target occluded objects in all circumstances, if necessary.

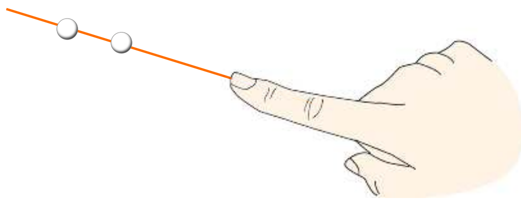

Figure 4. Occluded objects can be difficult to select with ray-casting, because the ray might also intersect objects in front of them.

Selection disambiguation using Lock Ray

We developed a selection disambiguation method based on *Lock Ray* [5], a ray-cast pointing method that uses a moveable depth cursor to select one target intersected by the ray. Users aim the ray by pointing, then they 'lock' it in place. With the ray locked, forward/backward finger movements can be used to reposition the cursor along the ray, as in Figure 5. The nearest object to the depth cursor is targeted. By locking the ray, users can position the depth cursor without their finger movements causing the ray to deviate from the target. Since disambiguation takes place in two stages, less information needs to be conveyed during selection. This is ideal for levitating object displays, which can only give limited feedback: as we later describe, we use the same *Shake* feedback in two contexts to mean different things.

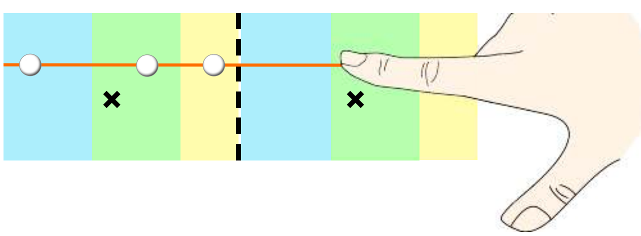

Figure 5. Fore/aft finger movement controls the 'depth cursor' used to select from the targets intersected by the ray. This shows how the finger position ('X') is used to select an occluded object.

Depth Ray [5] is a similar technique, except it combines aiming with depth cursor movement. This style of continuous 3D cursor control may be too difficult with the limited amount of feedback that can be given from levitating objects. Another alternative would be to reconfigure the levitation so that targets are not occluded, but this is undesirable. We chose *Lock Ray* because it works in situ without reconfiguration and it requires minimal feedback for each stage of the interaction.

For our version of the *Lock Ray* technique, we use extension of the thumb to 'lock' the ray, as shown in Figure 5. When the ray is locked, we use the index fingertip position to determine which object along the ray is being targeted. We use a 1:1 mapping between the levitation volume and the space in front of the levitation. Users can move their fingertip towards, or away from, the levitating objects to change their selection. So far, we have described how our adaptation of *Lock Ray* can be used to target occluded objects. An additional mechanism is needed to confirm the selection and end the targeting.

If only one levitating object is intersected by the ray (like in Figure 2), then that object can be selected without using the disambiguation mechanism; you just point at the object and dwell, without having to lock the ray and position the depth cursor. We also use the *Dwell* method for selection using *Lock Ray*: users confirm their selection by keeping the 'depth cursor' on the target for 1000ms. In Grossman's version of *Lock Ray* for a handheld remote [5], they used a button press to lock the ray and confirmed selection upon releasing the button. We also evaluate a similar method here that we call *Quick Release*: when users close their thumb again (i.e., to 'unlock' the ray), the targeted object is selected. Our second study, described later, compares the *Dwell* and *Quick Release* methods for confirming selection.

So far we have described how *Point-and-Shake* can be used to select a single levitating object. This can be extended to allow multiple objects to be sequentially selected: i.e., the user selects object one then selects object two. Selected objects can be deselected by pointing at them again. The *Shake* feedback, described in the following section, is used to support these interactions, by showing which objects have already been selected even if they are not currently being pointed at.

Other actions, like moving objects or querying their properties, can be easily performed after selection. For example, a user could select an object then reposition it using a gesture that maps object position to finger position (as in Figure 6). Or they could select multiple objects and gesture to have the system speak their values. Other interactions are outside of the scope of this paper, but these examples show how our selection gesture may precede other actions.

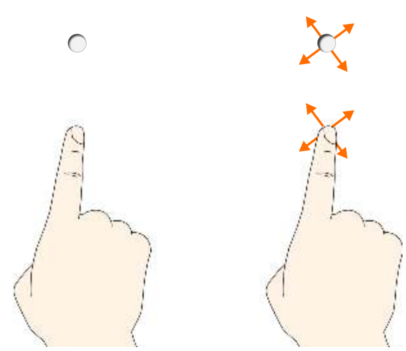

Figure 6. Selection can be followed by other actions, like repositioning objects by mapping object position to finger position.

Shake: Feedback through object movement

Feedback is necessary to show which objects are being targeted and if selection is complete. We use object movement to give this feedback, because the levitating objects are the only visual element of our levitation system. With acoustic levitation, objects can be moved in three dimensions, giving scope for complex feedback patterns. We use continuous side-to-side movement to give feedback. This motion is relative to the user's position, so it appears the same from all sides of a levitation. We chose side-to-side movement because it is simple: the motion is only in one axis. Informal testing found that the side-to-side motion of a levitating object was more noticeable than up-and-down and fore-and-aft motion, so we chose this for our feedback. We discuss the properties of movement in the implementation section.

Point-and-Shake needs to give feedback that shows which objects are intersected by the targeting ray. When the user points (like in Figures 2 and 4), we move all intersected objects from side-to-side (i.e., they *Shake*), showing where the ray is aiming. If only one object is targeted, then it will be selected directly by dwelling; once selection is complete, the object shakes at a slower speed, making it distinct from unselected objects. When many objects are targeted, disambiguation is necessary. When using the *Lock Ray* method, all targeted objects stop shaking once the ray is 'locked' in position, except for the object currently targeted by the 'depth cursor', which continues to shake as before. As the user moves their hand closer to, or away from, the levitation (like in Figure 5), the newly targeted object will shake instead.

A limitation of using object movement to convey feedback is that it constrains the minimum spacing between levitating objects. In an acoustic levitation system, objects must be at least $\frac{\lambda}{2}$ apart [20], which is approximately 4.3mm for the 40 kHz ultrasound used in our system and in the state-of-the-art. Additional space is required for object movement, therefore increasing object spacing. Moving a levitating object may also have unintentional effects on other objects, because of the necessary changes in the acoustic field. Unlike the first constraint, which is defined by physics, the latter can be mitigated through advances in acoustic field synthesis. The algorithms for levitating multiple objects are currently formative, but research is ongoing to develop techniques that better support multiple objects; for example, minimising unwanted areas of amplitude (like Carter did for ultrasound haptics [2]). We minimised movement to reduce the impact on the levitation, but we anticipate this being less of a problem in future.

Implementation: acoustic levitation and apparatus

Acoustic levitation is typically achieved using standing waves, created by opposing transducers. Small objects ($\varnothing \leq \frac{\lambda}{2}$) can be levitated in the low-pressure nodes of the standing waves. These can be moved in 3D by changing the relative phases of the transducers (see [17]). We use polystyrene balls as the objects for our levitation system, as their small size and low density is appropriate for our apparatus.

Our acoustic levitation system (Figure 1) uses two 8x4 arrays of 40 kHz ultrasound transducers (Murata MA40S4S, $\varnothing$1cm). The arrays are held in laser-cut panels that slot into a Lego frame. They face each other and are vertically separated by 65mm, a similar arrangement to *LeviPath* [17]. We use this setup because it offers the strongest levitation effect [18], but recent acoustics work is improving one-sided levitation [14], which means future iterations could offer a full hemisphere of visibility. We use the algorithms described by Omirou *et al.* [17] to control the transducer phases to allow levitation at the desired points. These levitation algorithms use a real-world coordinate system, where the origin is the centre of the bottom transducer array. Control signals to drive the transducers are generated by a custom board with two XMOS processors. Note: 40 kHz ultrasound is inaudible so the levitation is silent.

We implemented *Point* using a Leap Motion sensor. This gives the position of each finger, whether or not it is extended and a vector for the direction it is pointing. We use this information to determine when the user is pointing and when the thumb is extended for selection disambiguation (Figure 5). We consider the thumb extended if its angle to the index finger is $\geq$50°. Hand tracking positions are mapped to the same real-world coordinates used by the levitation system, enabling ray-casting to determine which objects the user is pointing at. The Leap Motion is held in a Lego mount attached to the transducer frame, so the known offset is added to the Leap Motion tracking coordinates for accurate conversion. During interaction, an object is targeted if the ray from the finger intersects the virtual sphere at its position. The ray starts at the fingertip position and follows the finger direction vector. It is important to choose an ideal sphere size, so that pointing accuracy can be enhanced without having to place objects further apart (to avoid overlapping targets). Our first user study investigates selection with a range of target sizes (5–30mm).

When the user extends their thumb, we 'lock' the ray position and orientation. We then use the positions of the intersected objects and the fingertip to determine which object the user is targeting. Like Grossman's implementation of *Lock Ray* [5], we use a 1:1 mapping between hand movement and cursor movement and select the nearest target object to the cursor (Figure 5). For the *Quick Release* selection method, we detect when the thumb is no longer extended (angle $< 50°$). For the *Dwell* selection method, we check the time spent aiming at the targeted object. Our second user study compares the performance of these techniques.

As discussed, the *Shake* feedback is accomplished by rapidly moving a levitating object from side-to-side. Our implementation creates this feedback by moving an object one step to the left, two steps to the right, then one step to the left (Figure 7); this is repeated continuously to create object movement centred around the object position. When targeting stops (i.e., feedback stops), the object returns to its origin.

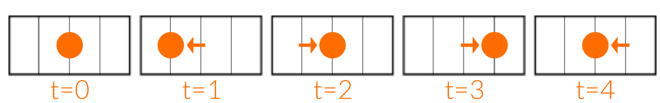

Figure 7. The feedback motion consists of four steps (1–4) that loop continuously; t=0 shows the starting position of the object.

This implementation of the feedback has two parameters: the distance the object is moved at each step; and the speed of movement. We now consider these in turn. As discussed earlier, minimising the movement distance is desirable because it means objects can be placed closer together. We investigated the minimum perceptible displacement in a pilot study, finding that each step should move the object by 0.3mm (resulting in a total side-to-side range of 0.6mm). This is the minimum distance a levitating object was moved before the change in position was noticeable by participants. We then tested how closely together objects could be positioned so that moving one by 0.3mm did not affect the position of the other. The theoretical minimum separation was 4.6mm ($\frac{\lambda}{2} + 0.3 \approx 4.6$), but we found they needed to be 10mm apart before moving one object rapidly did not disturb the other. As discussed in the previous section, advances in acoustic field synthesis could mitigate these side effects of object movement.

We vary the movement speed to convey the two object states discussed previously: 1) the object is actively being targeted; and 2) the object has been selected but is not being targeted. We investigated the fastest movement speed that still allowed stable levitation (i.e., the object was not 'ejected' from the levitation). The fastest reliable movement speed was $0.06m.s^{-1}$ (i.e., 5ms between each step). We used $0.01m.s^{-1}$ (i.e., 30ms delay between each step) to show when an object was targeted; although faster speeds were possible, they increase computational demand. Selected objects moved at $0.005m.s^{-1}$ (i.e., 60ms delay), visibly distinct from targeted objects.

We described *Point-and-Shake*, our interaction for selecting levitating objects. Applying existing techniques to this novel context is not straightforward: it is not known if levitating objects can give enough feedback for successful input and it is not clear how accurate users can be when interacting with these objects. Our first user study investigates the usability of *Point-and-Shake* with different target sizes. Our second user study evaluates our version of *Lock Ray* [5], to see if it is usable with the feedback our system gives, and to compare *Quick Release* with *Dwell*, two methods of confirming selection.

USER STUDY 1: SELECTING LEVITATING OBJECTS

This study investigated the usability of our pointing interaction for selecting levitating objects. For this study, none of the objects were occluded so we could focus on the simplest use case of *Point-and-Shake* with dwell. The aim was to see how quickly and accurately users could select objects with different virtual sphere sizes. Six sphere sizes were used: {5, 10, 15, 20, 25, 30}mm. The objects in our system need to be 10mm apart, but we included the 5mm target size to see how well users could make selections if smaller separations were feasible.

Experimental task and procedure

We used our acoustic levitation system to levitate two objects side-by-side, as shown in Figure 8. Objects were separated by 30mm to accommodate all sphere sizes. Study participants were asked to select a particular object ("left" or "right") using our pointing technique. The instruction was given verbally by the experimenter. After each selection, participants were asked to rest their hands on the table, before the experimenter started the next task.

Tasks were presented in blocks, with one block per sphere size. The order of the blocks was counterbalanced and there were 20 targeting tasks per block. Participants were given 20 seconds to complete each task, otherwise the task ended unsuccessfully. They were not told about the use of virtual targets or that this was varied between blocks; as far as they were concerned, they were to point directly at the levitating objects. At the beginning of the study, participants were given a short set of training tasks to become familiar with the interaction.

Figure 8. This photo shows the levitating objects that users were presented with for the experimental tasks.

For every task, we measured the total selection time, starting from when the user's hand was first detected directly over the sensor. All tasks started with the user's hands on the table, making this a consistent measurement. We also measured selection success rate (% complete selections within task time limit). Finally, we recorded the average finger position during each selection, to see how close to the objects users gestured. This was because if users point too close then they might disrupt the position of the objects by reflecting the sound. All performance data is provided in the supplementary material.

There were 15 participants (7 female), with a mean age of 32.6 years (sd 12 years). None of the participants had used this style of pointing interaction before, although some were familiar with gesture-based games (e.g., Xbox Kinect).

Results

Mean selection time was 4137ms (sd 4281ms), including the 1000ms dwell period for the selection: Figure 9. Times were not normally-distributed (Shapiro-Wilk W = 0.85, $p < 0.001$) so the Aligned-Rank Transform [22] was applied before analysis. A repeated-measures ANOVA found a significant main effect of sphere diameter on selection time: $F(5,69) = 65.1$, $p < 0.001$. *Post hoc* pairwise comparisons found significant differences between all sphere sizes ($p \leq 0.001$), except for {15mm vs 20mm}, {20mm vs 25mm}, and {25mm vs 30mm}; all p-values were Bonferroni-corrected for multiple comparisons. In all significant differences, the larger sphere size had the shorter selection time.

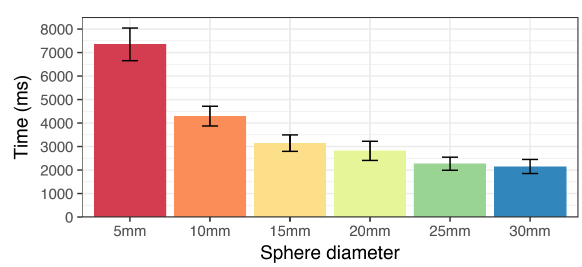

Figure 9. Mean selection times. Error bars show 95% CIs.

Mean task success rate was 96.4% (sd 5.4%); see Table 1. For all 65 failed tasks, the 20 second time limit elapsed; 55 of these were for the 5mm condition. Logistic regression was used to analyse the effect of sphere size on selection success. A repeated-measures ANOVA on the regression model found a significant main effect of sphere size: $\chi^2(5) = 192.8$, $p < 0.001$. *Post hoc* Tukey comparisons found a significantly lower success rate for 5mm than all other sizes (all $p < 0.001$). No other differences were significant ($p \geq 0.3$).

5mm	10mm	15mm	20mm	25mm	30mm
81.7%	97.7%	99.3%	100%	100%	99.7%

Table 1. Selection success rate per target sphere size.

We used the mean fingertip position to calculate the proximity of the finger to the levitated object during successful tasks. The mean distance was 59.9mm (sd 22.5mm): see Figure 10. Distance did not have a normal-distribution (Shapiro-Wilk W = 0.95, p = 0.001) so the Aligned-Rank Transform [22] was applied prior to analysis. A repeated-measures ANOVA found a significant effect of sphere diameter on distance: $F(5,69) = 33.5$, $p < 0.001$. *Post hoc* pairwise comparisons found significant differences between: 5mm and all others (all $p < 0.001$); 10mm and {20mm, 25mm, 30mm} (all $p < 0.001$); and 15mm and {25mm, 30mm} (all $p \leq 0.03$). In all cases, proximity was lower (i.e., closer) for smaller sphere sizes.

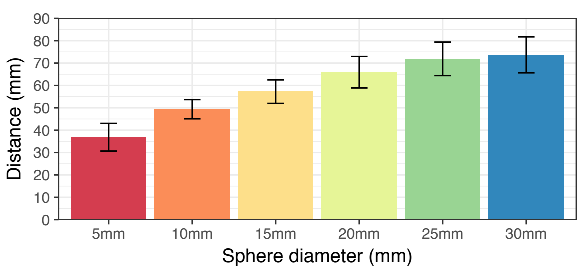

Figure 10. Mean finger proximity to targets. Error bars show 95% CIs.

Discussion

One aim of this study was to investigate a 'usable' target sphere size, because it was not known if users could accurately and easily select targets using our pointing gesture and object shake feedback. We found consistently high success rates for sizes ≥ 10mm, suggesting this is an ideal minimum target size. Increasing the target size beyond 10mm did not improve the success rate but did decrease selection time. For 10mm targets, selection took a mean of 4.6 seconds, including the 1000ms dwell period. This seems reasonable but may be cumbersome when interacting with complex levitations and selecting many objects. Just increasing the size to 15mm reduces the mean selection time to 3.2s; excluding the dwell time, this is a 39% reduction in time spent trying to point at the levitating object.

One of the main reasons larger targets have faster selection times is that users find the target sooner as they move their hand forward to start pointing. When this happens and they get feedback from the correct object, they tend to stop moving their hand, as shown by the proximity data. Conversely, for smaller targets, users move their finger closer to the object rather than stay in place and redirect their aim. Getting too close can be a problem because the user's hand might occlude their own, and other people's, view of the levitation. For acoustic levitation systems, getting too close may also disrupt the acoustic field; 40 kHz ultrasound is almost entirely reflected by skin [2], meaning the reflected sound could affect the levitated objects. This only appeared to be a problem for the 5mm target size, however, when close fingers would often result in a visible 'wobble' of the object being pointed at.

We included 5mm target spheres in this study despite being smaller than our minimum object separation, to evaluate our pointing interaction for use in systems not subject to the same acoustic constraints as ours. The significantly longer selection times and higher error rate suggests that targets should not be this small. A disadvantage of ray-cast pointing is that small finger movements can result in a significant ray movement, meaning users had to be really accurate to keep pointing at the 5mm target sphere for the dwell period. A limitation of our study design was that hand position was not fixed, allowing users to move closer if necessary to improve accuracy. This means the tasks did not have a consistent difficulty and our results should be interpreted with this in mind. Our data suggests that if hand position was fixed (e.g., at 5cm from the levitation) then performance for 5mm targets would be worse.

USER STUDY 2: SELECTION DISAMBIGUATION

We implemented a selection disambiguation mechanism based on *Lock Ray* [5], so that users can target occluded objects. Our first study showed that *Point-and-Shake* enables accurate aiming of the ray, so our second study investigated if the *Lock Ray* technique allowed successful disambiguation with the feedback our system provides. This study also compares two confirming methods: QUICK RELEASE and DWELL. DWELL was successful for selecting unoccluded objects in Study 1, but the more complex *Lock Ray* interaction may mean that users prefer QUICK RELEASE instead. This may be preferable because it gives users more time to interpret the feedback and make sure they are selecting the correct object.

Experimental task and procedure

We used our acoustic levitation system to levitate two objects, spaced 20mm apart and facing away from the user so that the furthest object would be occluded by the closest one, as shown in Figure 11. This means the object is occluded to the ray-cast technique, but it was still visible to users. We rotated our system by 90° from its orientation in Study 1. We used a 20mm sphere size, as this performed well in the first study and we wanted to make it easier to target both objects at the same time, so that we could focus on the disambiguation method. Participants were asked to select a particular object ("closest" or "furthest") using the *Lock Ray* method. We instructed them to point at both objects at the same time before locking the ray; this meant they had to use the disambiguation mechanism, rather than simply pointing 'around' the occluding object. Our experiment did not allow progress until this condition was met. After each selection was completed, participants were asked to rest their hands on the table, before the next task started.

Figure 11. This photo shows the levitating objects that users were presented with for the experimental tasks.

Tasks were presented in four blocks (two blocks per selection method). The order of the blocks was counterbalanced. There were 30 targeting tasks per block. Participants were given 45 seconds to complete each task, otherwise the task was ended unsuccessfully. At the beginning of the study, participants were given a set of 10 training tasks for each selection mechanism, to familiarise them with the interaction.

For every task, we measured the total selection time, starting from when the user's hand was first detected over the sensor. All tasks started with the user's hands on the table, making this a consistent measurement. We also measured the time spent aiming the ray (before 'locking' it), as well as the time spent confirming the selection. These would give insight into how long each part of the *Lock Ray* interaction took. Finally, we measured selection success rate (% correct selections). After the study, we asked participants to identify their preferred method for confirming selection. All performance data is provided in the supplementary material.

There were 15 participants (7 female, 2 left handed), with a mean age of 26.6 years (sd 6.3 years). Two also did Study 1.

Results

Mean selection time was 3034ms (sd 1780ms), see Figure 12. A t-test found that selection time was significantly higher for DWELL than QUICK RELEASE: 3219ms vs 2849ms; t(14) = 4.04, p = 0.001.

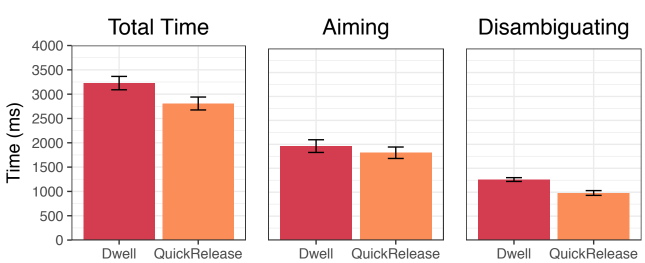

Figure 12. Mean times for task (left), aiming (middle) and disambiguating (right). Y-axis is the same in each plot. Error bars show 95% CIs.

Mean aiming time (i.e., time before 'locking' the ray) was 1891ms (sd 1434ms), see Figure 12. A t-test found that the difference between DWELL and QUICK RELEASE was not significant: 1960ms vs 1822ms; t(14) = 1.72, p = 0.11.

Mean time to confirm selection (i.e., after 'locking' the ray) was 1120ms (sd 518ms), see Figure 12. A t-test found significantly longer confirmation times for DWELL than QUICK RELEASE: 1259ms vs 982ms; t(14) = 5.04, p < 0.001.

Mean task success rate was 94.6% (sd 3.3%). Five tasks were unsuccessful because the time limit elapsed. There were 48 and 50 incorrect selections for DWELL and QUICK RELEASE, respectively. Logistic regression was used to analyse the effect of selection method on task success. The difference was not significant: Z = 0.21, p = 0.83.

Twelve chose QUICK RELEASE as their preferred method of confirming selection (80%). Most of them said it was because it offered more control over when the selection ended. This worked both ways, allowing faster selections when confident about the targeted object, and allowing more time when unsure about which object was being targeted. Others liked that it ended with an affirmative action, rather than waiting for time to elapse. Those who preferred DWELL said it was "simpler", since they just waited for the selection to complete.

Discussion

The selection success rate (94.6%) suggests that our version of *Lock Ray* was usable for selecting between two levitating objects. We were unsure if the amount of feedback given from *Point-and-Shake* would be sufficient, since it was unable to show the position of the 'depth cursor' used to disambiguate between two objects, but this did not appear to be a problem and the *Shake* feedback was enough to show which object was currently being targeted. Technical constraints meant we could not investigate selection between three or more objects so more work is needed to see how this interaction scales successfully to more dense levitations.

The selection times in this study were closer to the selection times from the first study than expected, despite the added complexity of occlusion (3s in Study 2 and 2.8s in Study 1 for the equivalent 20mm target size). The times are not directly comparable; the difficulty in Study 1 was aiming the ray, and in Study 2 it was disambiguating between targets. Disambiguation was easier than we anticipated: the data for each step of the *Lock Ray* technique shows that most of the time was spent aiming the ray (1.9s), rather than disambiguating (1.1s). This may be because users pre-empted the disambiguation step, by positioning their finger closer to the levitation system when they were asked to select the furthest away object. This is supported by the small difference between the dwell period (1s) and the targeting time for DWELL (1.2s), which suggests users generally 'landed' on the target right away. We investigated this *post hoc* and found that 65.8% of DWELL selections instantly 'hit' the target object. This may explain the good performance with the *Shake* feedback even though it could not show a cursor: users may only have needed it to confirm that they had indeed landed on the intended target.

Users strongly preferred the QUICK RELEASE method for confirming selection, often stating that this gave them more control than DWELL. Some liked that QUICK RELEASE gave them more time to make sure they had selected the correct object, as the DWELL method ended after 1s on a target. This was our initial motivation for investigating QUICK RELEASE, to give users more time. However, they also used this method to end selection quickly, shown by the lower mean confirmation time. A *post hoc* investigation found that 57.5% of QUICK RELEASE confirmations took less than 1000ms (the minimum time needed for DWELL) and 20.4% took less than 500ms. This could be a further reason for the short task times.

Our results show that DWELL was successful in this study and we recommend using it for consistency with our *Point* interaction for selecting unoccluded objects (as in Study 1). However, our findings suggest compelling reasons for using QUICK RELEASE to confirm selections, so we suggest a hybrid of the methods: users can disambiguate between levitating objects using *Lock Ray*, confirming selections by either dwelling for 1000ms, or ending selection early using QUICK RELEASE.

OVERALL DISCUSSION AND CONCLUSION

We presented *Point-and-Shake*, an interaction for selecting levitating objects, combining ray-cast pointing input with object movement as feedback. It was designed with the capabilities of acoustic levitation in mind and uses only the objects for feedback. We described two studies that evaluated the performance of *Point-and-Shake*. The first investigated the effect of virtual target size when selecting levitating objects, finding that users could quickly and accurately make selections when the target size was 10mm or greater. Users adapted to the increased difficulty of selecting very small targets (5mm) by moving closer to the levitating objects, which may affect acoustic levitation by reflecting sound back towards the objects.

Point-and-Shake used a variation of the *Lock Ray* technique [5] to allow users to select occluded objects. We evaluated this aspect of the interaction in our second study, which also compared two methods for confirming selection. Users could successfully select occluded objects with both methods, but especially liked *Quick Release* because of the extra control it gave them over the interaction. Our studies looked at selection between two levitating objects, as the state-of-the-art in acoustic levitation does not yet allow more than two objects to be independently and reliably animated. Our results suggest *Point-and-Shake* would be effective with more objects, however. They show that our *Shake* feedback supported accurate ray aiming (Study 1) and depth cursor positioning (Study 2) and we expect this to also apply to more complex levitations.

To conclude, we presented *Point*, a novel interaction for selecting levitating objects, and *Shake*, a feedback technique based on object movement. Together, these allow efficient selection of mid-air objects for the first time, paving the way to more complex interactions with levitating object displays and showing the potential for interaction with this exciting new technology.

ACKNOWLEDGEMENTS

This research is funded by the European Union Horizon 2020 research and innovation programme (#737087: Levitate). We thank Gianluca Memoli for his technical help and wizardry, and we thank the reviewers for helping to improve this paper.

REFERENCES

1. Tobias Alrøe, Jonas Grann, Erik Grönvall, Marianne Graves Petersen, and Jesper L. Rasmussen. 2012. Aerial Tunes: Exploring Interaction Qualities of Mid-air Displays. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction - NordiCHI '12*. ACM Press, 514–523. DOI: http://dx.doi.org/10.1145/2399016.2399095

2. Thomas Carter, Sue Ann Seah, Benjamin Long, Bruce Drinkwater, and Sriram Subramanian. 2013. UltraHaptics: Multi-Point Mid-Air Haptic Feedback for Touch Surfaces. In *Proceedings of the 26th Symposium on User Interface Software and Technology - UIST '13*. ACM Press, 505–514. DOI: http://dx.doi.org/10.1145/2501988.2502018

3. Euan Freeman, Ross Anderson, Carl Andersson, Julie Williamson, and Stephen Brewster. 2017. Floating Widgets: Interaction with Acoustically-Levitated Widgets. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces - ISS '17 Demos*. ACM Press, 417–420. DOI: http://dx.doi.org/10.1145/3132272.3132294

4. Euan Freeman, Stephen Brewster, and Vuokko Lantz. 2014. Tactile Feedback for Above-Device Gesture Interfaces: Adding Touch to Touchless Interactions. In *Proceedings of the 16th International Conference on Multimodal Interaction - ICMI '14*. ACM Press, 419–426. DOI:http://dx.doi.org/10.1145/2663204.2663280

5. Tovi Grossman and Ravin Balakrishnan. 2006. The design and evaluation of selection techniques for 3D volumetric displays. In *Proceedings of the 19th annual ACM symposium on User interface software and technology - UIST '06*. ACM Press, 3–12. DOI: http://dx.doi.org/10.1145/1166253.1166257

6. Tovi Grossman, Daniel Wigdor, and Ravin Balakrishnan. 2005. Multi-finger gestural interaction with 3D volumetric displays. *ACM Transactions on Graphics* 24, 2 (2005), 931. DOI: http://dx.doi.org/10.1145/1073204.1073287

7. Keisuke Hasegawa and Hiroyuki Shinoda. 2013. Aerial Display of Vibrotactile Sensation with High Spatial-Temporal Resolution using Large-Aperture Airborne Ultrasound Phased Array. In *Proceedings of World Haptics 2013*. IEEE, 31–36. DOI: http://dx.doi.org/10.1109/WHC.2013.6548380

8. Takayuki Hoshi, Masafumi Takahashi, Takayuki Iwamoto, and Hiroyuki Shinoda. 2010. Noncontact Tactile Display Based on Radiation Pressure of Airborne Ultrasound. *IEEE Transactions on Haptics* 3, 3 (2010), 155–165. DOI:http://dx.doi.org/10.1109/TOH.2010.4

9. Seki Inoue, Yasutoshi Makino, and Hiroyuki Shinoda. 2015. Active touch perception produced by airborne ultrasonic haptic hologram. In *Proceedings of IEEE World Haptics Conference - WHC '15*. IEEE, 362–367. DOI:http://dx.doi.org/10.1109/WHC.2015.7177739

10. Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. 2012. Radical Atoms: Beyond Tangible Bits, Toward Transformable Materials. *Interactions* XIX.1 (2012), 38–51. DOI: http://dx.doi.org/10.1145/2065327.2065337

11. Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2013. Evaluating the Efficiency of Physical Visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, 2593–2602. DOI: http://dx.doi.org/10.1145/2470654.2481359

12. Georgios Korres and Mohamad Eid. 2016. Haptogram: Ultrasonic Point-Cloud Tactile Stimulation. *IEEE Access* 4 (2016), 7758 – 7769. DOI: http://dx.doi.org/10.1109/ACCESS.2016.2608835

13. Jinha Lee, Rehmi Post, and Hiroshi Ishii. 2011. ZeroN: Mid-Air Tangible Interaction Enabled by Computer Controlled Magnetic Levitation. In *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*. ACM Press, 327–366. DOI: http://dx.doi.org/10.1145/2047196.2047239

14. Asier Marzo, Sue Ann Seah, Bruce W. Drinkwater, Deepak Ranjan Sahoo, Benjamin Long, and Sriram Subramanian. 2015. Holographic acoustic elements for manipulation of levitated objects. *Nature Communications* 6 (2015), Article 8661. DOI: http://dx.doi.org/10.1038/ncomms9661

15. Yoichi Ochiai, Takayuki Hoshi, and Jun Rekimoto. 2014. Pixie Dust: Graphics Generated by Levitated and Animated Objects in Computation Acoustic-Potential Field. *ACM Transactions on Graphics* 33, 4 (2014), Article 85. DOI: http://dx.doi.org/10.1145/2601097.2601118

16. Yoichi Ochiai, Takayuki Hoshi, and Ippei Suzuki. 2017. Holographic Whisper: Rendering Audible Sound Spots in Three-dimensional Space by Focusing Ultrasonic Waves. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, 4314–4325. DOI: http://dx.doi.org/10.1145/3025453.3025989

17. Themis Omirou, Asier Marzo, Sue Ann Seah, and Sriram Subramanian. 2015. LeviPath: Modular Acoustic Levitation for 3D Path Visualisations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, 309–312. DOI:http://dx.doi.org/10.1145/2702123.2702333

18. Themis Omirou, Asier Marzo Perez, Sriram Subramanian, and Anne Roudaut. 2016. Floating Charts: Data Plotting using Free-Floating Acoustically Levitated Representations. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 187–190. DOI: http://dx.doi.org/10.1109/3DUI.2016.7460051

19. Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. 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 - UIST '96*. ACM Press, 79–80. DOI: http://dx.doi.org/10.1145/237091.237102

20. Deepak Ranjan Sahoo, Takuto Nakamura, Asier Marzo, Themis Omirou, Michihiro Asakawa, and Sriram Subramanian. 2016. JOLED: A Mid-air Display based on Electrostatic Rotation of Levitated Janus Objects. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16*. ACM Press, 437–448. DOI: http://dx.doi.org/10.1145/2984511.2984549

21. Lode Vanacken, Tovi Grossman, and Karin Coninx. 2009. Multimodal selection techniques for dense and occluded 3D virtual environments. *International Journal of Human Computer Studies* 67, 3 (2009), 237–255. DOI: http://dx.doi.org/10.1016/j.ijhcs.2008.09.001

22. Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 2011. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only ANOVA Procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '11*. ACM Press, 143–146. DOI: http://dx.doi.org/10.1145/1978942.1978963

23. Toshiya Yui and Tomoko Hashida. 2016. Floatio: Floating Tangible User Interface Based on Animacy Perception. In *Adjunct Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16 Adjunct*. ACM Press, 43–45. DOI: http://dx.doi.org/10.1145/2984751.2985699

24. Shumin Zhai, William Buxton, and Paul Milgram. 1994. Investigating The "Silk Cursor": Transparency for 3D Target Acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '94*, Vol. 1. ACM Press, 459–465. DOI: http://dx.doi.org/10.1145/191666.191822