

# Indirect Touch Manipulation for Interaction with Stereoscopic Displays

Adalberto L. Simeone\*

University of Portsmouth

## ABSTRACT

Research on 3D interaction has explored the application of multi-touch technologies to 3D stereoscopic displays. However, the ability to perceive 3D objects at different depths (in front or behind the screen surface) conflicts with the necessity of expressing inputs on the screen surface. Touching the screen increases the risk of causing the vergence-accommodation conflict which can lead to the loss of the stereoscopic effect or cause discomfort. In this work, we present two studies evaluating a novel approach based on the concept of indirect touch interaction via an external multi-touch device. We compare indirect touch techniques to two state-of-the-art 3D interaction techniques: *DS3* and the *Triangle Cursor*. The first study offers a quantitative and qualitative study of direct and indirect interaction on a 4 DOF docking task. The second presents a follow-up experiment focusing on a 6 DOF docking task. Results show that indirect touch interaction techniques provide a more comfortable viewing experience than both techniques. It also shows that there are no drawbacks when switching to indirect touch, as their performances in terms of net manipulation times are comparable.

**Index Terms:** H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input Devices and Strategies;

## 1 INTRODUCTION

Stereoscopic displays are now a widely available technology and represent the forefront of the media and entertainment industry. Stereoscopia has been found to significantly improve user performance in depth-related tasks [20, 30]. Furthermore, it has provided new interaction challenges to the research community. One of these is its coupling with multi-touch input. When the two technologies are used together, 3D direct touch interaction metaphors [2, 11, 12, 18, 23] are no longer optimal. One of the main issues arises as the consequence of its most striking advantage: the perception of depth in the virtual environment. Objects can be perceived as appearing within the screen (positive parallax), at the screen (zero parallax or at *screen depth*) or outside the screen (negative parallax). While those at screen depth are most suited for direct multi-touch input, interaction is not always restricted to that zone. To directly affect objects having positive or negative parallax users have to touch the projection surface. Due to the vergence-accommodation conflict [14], fixating on our finger at the moment of touch introduces a mismatch with the object's perceived depth due to our eyes converging on the screen. This can cause discomfort or the loss of the stereoscopic effect.

We argue that decoupling the input surface from the output, and onto an external touch-capable device, has the potential of retaining the expressiveness of multi-touch input, while overcoming some of the issues commonly associated with stereoscopia. Using an indirect touch surface to affect objects in a stereoscopic environment has some advantageous features that make it a compelling alternative to direct touch. For example, it does not cause occlusion of parts of the display generating the 3D effect. In a previous study [25],

participants performed more errors in the direct touch condition than with indirect touch, due to occlusion issues. In addition, the ubiquitousness of tablet devices, as an alternative to specialist input devices, can support the adoption of 3DUIs in those contexts outside of research laboratories and industry professionals. Compared to traditional interaction input devices such as the mouse and keyboard, tablets afford more degrees of freedom (DOF) and are less dependant on mode-switching.

In this paper we present two studies comparing indirect and direct-touch interaction techniques for stereoscopic displays in 3D docking tasks. In the first, a 4 DOF task, we compared an interaction technique originally developed for monoscopic visualization (*DS3* [18]) and one explicitly designed for stereoscopic visualization (*Triangle Cursor* [26]) to one based on indirect touch interaction that we designed. Results from this study informed the design of the second one where we focused on a more complex 6 DOF task. We found that the 4 DOF indirect touch interaction technique, while comparable to *DS3*, it was faster than the *Triangle Cursor* and was reported to provide a better quality of vision than both. In the 6 DOF task, the indirect touch technique also offered comparable performance to direct touch interaction when considering net manipulation times. This shows that there are no significant drawbacks when using an indirect touch interaction technique which, coupled with the advantages described, makes it a compelling interaction modality.

## 2 RELATED WORK

Multi-touch technologies provide a blend of intuitive and powerful interaction capabilities. It is the de-facto interaction standard on mobile devices and a compelling alternative in other contexts (e.g., desktop or large displays). Until recently, however, research on multi-touch technologies has mostly focused on 2D interaction.

### 2.1 Direct Touch 3D Interaction

Hancock et al. [11] were among the first to address the problem of 3D multi-touch interaction in their study of *shallow-depth* interaction, aptly named due to the non-extensive depth of the 3D environment. The authors describe three direct touch interaction techniques (ITs) using one, two or three fingers. Their results indicate that the three-touch IT was faster and preferred by users. This work was followed by *Sticky Tools* [12], a technique that maps each finger to 2 DOF that are then integrated. This introduces the notion of *integrability* versus *separability* of input.

Integrated techniques can be thought of as deriving from the well-known 2D Rotate-Scale-Transform (RST) technique. One of the reasons of its widespread application is the preserving of the original location of the user's contact point on an object after each manipulation. This allows the user to conceptualize and predict how the object will be affected by moving their fingers. Reisman et al. [23] applied this concept to 3D manipulation through the use of a constraint solver. The techniques described above have not been formally evaluated: their effectiveness was questioned in two works [4, 18]. These authors advocate the notion of separability in 3D manipulation techniques as it might be difficult for users to estimate the 3D effect a 2D manipulation gesture will have on an object, given its 3D projection. Martinet et al. [17] thus initially proposed the *Z-Technique* to control *xy*-translation with one finger and map *z*-translation to an indirect finger. To address this issue, they presented a new 6 DOF technique, *DS3* [18], where rotation

\*e-mail: adalberto.simeone@port.ac.uk

and translation modalities are strictly separated according to the number of fingers in contact with the manipulable object (one finger for translation, two for rotation).

## 2.2 Touch Manipulation in Stereoscopy

To achieve the stereoscopic effect, it is necessary to simulate the same process occurring in our brain by having each eye perceive only its associated perspective of a scene. There are several possibilities to produce stereoscopic images: synchronized shutter glasses, polarized projection, parallax barrier, etc. In a 3D application, scenes are rendered from two slightly offset viewpoints. The brain then merges the two pictures, giving the user the perception of depth.

When users want to interact with objects that appear to be floating in front of the display, this becomes problematic because most multi-touch screens are only sensitive to touches expressed on the screen surface. In order to interact, users have to inevitably penetrate objects with their fingers. This disturbs the stereoscopic effect and can cause the occurrence of the vergence-accommodation conflict [14].

When the eyes converge on the finger at screen depth, the mismatch between the finger's depth and the object's can break the stereoscopic effect. The brain thus stops fusing the two projections. Users might also experience a feeling of dizziness as depth cues lead us to expect that the 3D object should be in front of the finger which is currently passing through it. By focusing on the 3D position of the object, the perception of our finger is instead disturbed. Furthermore, in those cases where the hand or finger is big enough to completely occlude an object, the awareness of the environment is reduced.

In general, touching a 3D object is prone to ambiguities, due to the presence of two eye projections. Studies in this domain have shown that users are likely to touch the midpoint of the two projections or lean towards one of the two projections [28]. To overcome potential issues, Valkov et al. [27] propose to translate the scene towards the screen when a user wishes to manipulate objects with negative parallax. However, this solution is no longer optimal when the translation applied to the scene becomes noticeable. Bruder et al. [6] propose the identification of an on-surface suitable region capable of capturing 2D touches with reliable accuracy. Mid-air interaction is also prone to these issues when directly manipulating objects above the screen. Bruder et al. [7] studied these issues by comparing direct pointing to an offset-based virtual cursor and hand. The authors found that direct pointing was faster but less accurate than the offset-based techniques. Using camera-based 3D gesture recognition above the surface is an alternative approach [13]. However, interaction with intangible objects is prone to confusion and overshooting issues [8].

## 2.3 Indirect Interaction Techniques

In order to keep the positive aspects of passive haptics provided by touch surfaces and mitigate the above issues, researchers have investigated the feasibility of indirect techniques. In broader terms, all interaction techniques that require users to interact with an object by means of a proxy (such as a cursor or other UI elements) can be considered to be indirect techniques [9]. Various 3D input devices could be used to interact indirectly in a 3D context [3, 24]. In the remainder of this paper, we narrow our scope to the evaluation of the indirect use of touch input. Multi-touch devices are ubiquitous if compared to specialist 3D input devices, so devising effective indirect interaction techniques content has the potential of enabling a wider user base to interact with 3D content. Among these we differentiate between those that express input indirectly *on the same surface* used for display and those that use an *external device*, separated from the output display.

In the first subcategory are those that use touch to specify a position in 3D space, such as the Balloon technique [2] or the Triangle Cursor [26]. The former (originally designed for Augmented Reality scenarios) uses the metaphor of a helium balloon anchored to the position of the finger on the table [2]. The Triangle Cursor was

specifically designed for stereoscopic interaction. Users touch the table surface with two fingers mapped to a 3D cursor. All manipulations affecting it are reflected by any attached object. Void Shadows [10] uses shadows cast by the user's contact point on the display surface to specify a volume at positive parallax, behind the screen, to select and manipulate an object.

Indirect techniques using external surfaces have been mainly developed in the context of monoscopic visualization. Knoedel et al. [15] showed that an adapted RST technique using an external tablet provides better accuracy than its direct counterpart, although slower in terms of completion times. Ohnishi et al. [22] used an external tablet to map its 2D input to the surface of an arbitrary 3D object. In a previous work [25], we designed a study comparing direct touch to indirect touch in a task where users had to control an object within a densely cluttered 3D environment while avoiding distractor objects. The results show that users interacting with an external indirect device performed 30% less collisions. However, the interaction technique used only 3 DOF.

In this work, we explored if, in more complex 4-6 DOF tasks, an external indirect multi-touch device could mitigate the above issues while still providing an effective way to interact with content perceived through stereoscopic displays.

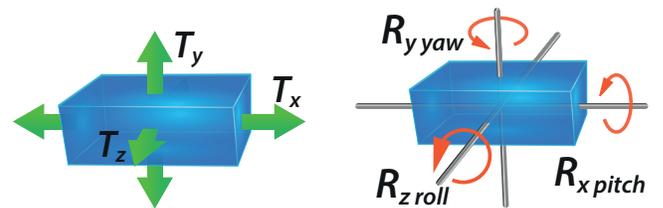


Figure 1: The illustration shows the reference axes used for translation and rotation actions used throughout the paper.

## 3 INTERACTION TECHNIQUES

In the following section we describe the techniques used in the two user studies, the motivations for their choice, their input mappings and the apparatus used throughout this work. For reference, Fig. 1 shows the axes of translation and rotation that we use to describe the interaction techniques.

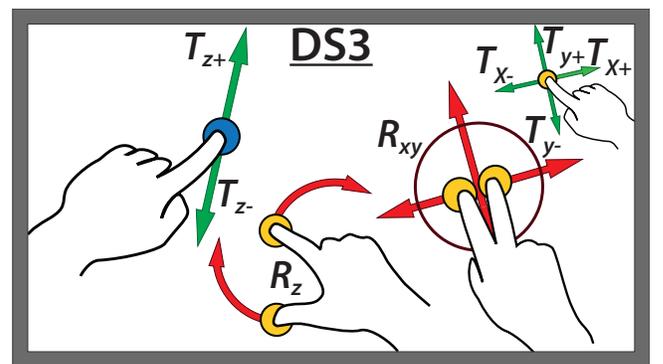


Figure 2: Touch gesture mappings for the DS3 technique. Golden circles indicate contact points with the object, whereas blue ones indicate contact points outside the object's area. Green arrows indicate gestures affecting translation axes, while red arrows indicate rotation axes. The circle around the four arrows indicates that the 2D movement is combined to affect both rotation axes simultaneously ( $R_{xy}$ ). Conversely, only vertical displacement is used in  $T_z$ .

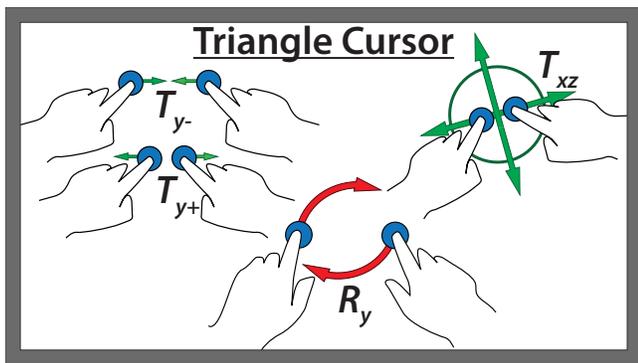


Figure 3: Touch gesture mappings for the *Triangle Cursor* [26]. The distance between the fingers affects the height of the attached object ( $T_y$ ) while moving them in parallel affects its position on the plane ( $T_{xz}$ ). Pivoting affects its yaw rotation ( $R_y$ ).

### 3.1 DS3 — Depth-Separated Screen-Space

*DS3* was designed in the context of monoscopic multi-touch interaction. It was compared to other 3D monoscopic techniques, (*Sticky Fingers* [12] and *Screen Space* [23]), and found to be faster. Since then, Liu et al. [16] have proposed a technique faster than *DS3* but comparable to *Sticky Tools*. This result is inconsistent with those reported by Martinet et al. [18], where *DS3* was found to be faster than *Sticky Tools*. This might be due to the context in which Liu et al.’s technique and *DS3* were used: small form-factor devices (5” and 11” screens) in the former and large screens (30” and above) in the latter. We thus chose *DS3* as a baseline as it is reminiscent of classic 2D direct touch metaphors, as opposed to other techniques based on widgets [4].

In the context of this technique, *DS3* differentiates between *direct* (touch points on the contact area of the manipulable objects) and *indirect* fingers (touch points outside this area). Fig. 2 shows *DS3*’s mappings. It uses one direct finger to manipulate  $xy$ -translation axes. When an indirect finger touch does not intersect the object, it is uni-dimensionally associated to the  $z$ -axis; i.e. moving the finger up or down controls the object’s depth. Two direct fingers cause the object to be rotated through the use of a constraint solver as in *Screen Space* [23]. It maps the position of the projection of the 3D locations of the 2D contact points on the object so that when the user moves their fingers, the object will be rotated so that those same initial 3D projected points will be under the fingers’ new position. Pivoting the two fingers causes the object to roll. An additional indirect finger enables simultaneous rotation and depth translation.

### 3.2 Triangle Cursor

The *Triangle Cursor* was chosen as it was explicitly designed for 4 DOF table-top stereoscopic interaction [26]. It was found to be faster than the *Balloon* technique [2]. When two touch points are detected, a 3D triangle appears. The distance between the two points determines the height of a sphere that indicates the actual 3D location for selection and manipulation purposes. The two intersections with the “floor” of the scene are mapped to the base of the triangle. By moving them in parallel, the user translates the cursor on the plane. When the angle between them changes, the cursor is rotated. Fig. 3 shows the mappings. Objects are selected by tapping with a third finger anywhere on the screen. If the sphere is intersecting an object, then it becomes attached and all cursor manipulations are reflected by the object. By tapping again with a third finger, de-selection occurs. However, if the object is still attached, lifting the two fingers can lead to exit-errors. In the context of their work, the authors

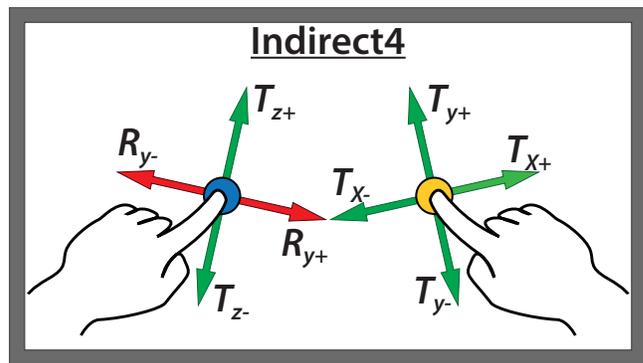


Figure 4: Touch gesture mappings for *Indirect4*. Translation in  $xy$  and  $z$  works analogously to *DS3* with the exception that horizontal displacement affects yaw rotation ( $R_y$ ).

describe it as an indirect technique as it uses *indirect* touches, albeit on the same surface used for display. To avoid ambiguities, in the remainder of this paper by *indirect touch* we explicitly refer to techniques using an external multi-touch device.

### 3.3 Indirect4

As the design space for indirect techniques is largely unexplored, for the first user study we designed a new 4 DOF technique. We wanted to compare the performance of a basic indirect technique against direct techniques that were built on top of previous knowledge in the area. *Indirect4*’s translation mappings (shown in Fig. 4) are analogous to *DS3* with the exception that depth translation and yaw rotation are integrated. The technique differentiates between dominant and non-dominant fingers by checking the order with which they make contact with the screen. The first contact point is assumed to belong to the dominant hand. An additional (second) contact point is assumed to belong to the non-dominant hand. *Indirect6*, a 6 DOF extension, is described in the context of the second user study.

The first design choice of *Indirect4* concerned the use of *absolute* or *relative* inputs. For the first study, we chose absolute mappings as we wanted to see whether a direct touch technique could be adapted to an external device, keeping everything else unchanged. The surface was thus normalised in a  $[0, 1]$  range and mapped to the surface of the screen projection. In the initial object selection phase, once the user touches the input device, a 2D stereoscopic cursor showing the corresponding screen location is displayed in the environment; the external device’s display remains blank. Once the user’s touch point is over the location of the 2D projection of the manipulable object, the cursor disappears for the duration of the interaction. This gives users feedback on the successful selection of the object and does not obstruct its manipulation in those cases where it would occlude the object. It reappears if, after lifting the finger, the successive touch location does not result in an intersection with the object.

### 3.4 Apparatus

We used a rear-projection system based on the diffused illumination technique, measuring 70cm  $\times$  38.5cm (with a diagonal of 31.5”, 80cm). We used a BenQ w1070 to project frame-sequential 3D images at a resolution of 1280  $\times$  720 pixels at 60 fps per eye. In the direct condition, a Point Grey Grasshopper camera was used to capture touch input at 60 fps. The indirect condition used a Microsoft Surface RT tablet (placed on a desk) where an application intercepted touch events. In both conditions, events were streamed through the network to a different rendering system. The 3D applications implemented in this work were built with a custom 3D engine using

*SharpDX*<sup>1</sup>, a C# DirectX port. The stereoscopic projection was based on two parallel left and right perspective projections (i.e. not toed-in) using an interocular distance of 6.35 cm and a distance of 68.5 cm from the screen. Participants were sitting during the whole duration of the experiment. We did not use head-tracking.

#### 4 FIRST STUDY: 4 DOF DOCKING TASK

Our exploration started with a user study focused on a simple 4 DOF 3D docking task. We compared two state-of-the-art techniques (*DS3* and *Triangle Cursor*) to *Indirect4*. Since *Triangle Cursor* is a 4 DOF technique we constrained our implementation of *DS3* to 4 DOF and used *Indirect4* for the indirect condition.

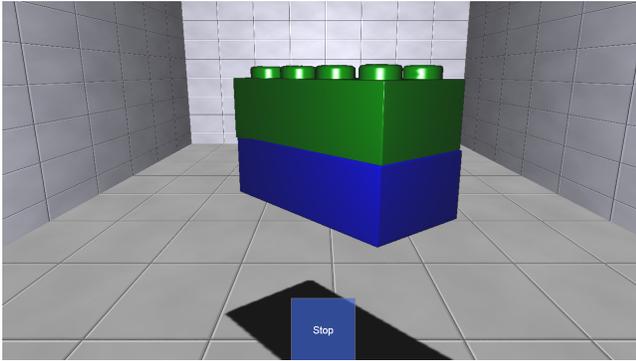


Figure 5: A trial from the first user study showing the manipulable brick (in green) docked to the goal brick (in blue). All 3D images in the paper are rendered through a *monoscopic* projection.

#### 4.1 Task

We chose a 4 DOF 3D docking task [31] for the first study. We asked participants to place a Lego<sup>®</sup> red  $2 \times 4$  brick on top of the blue one as quickly and accurately as possible by using the three chosen *techniques*. When the manipulable brick was within a distance and rotation threshold (15% of the initial mismatch), its colour turned to green; otherwise being red while idle, and golden while being manipulated. The goal brick remained blue. To test the effects that parallax might have on the task, the initial positions of the bricks were placed at different combinations of the three *parallax* regions (excluding those with the same parallax). The two objects were consecutively placed according to each of the six parallax combinations and repeated five times. At each repetition, the position of the goal brick alternated between five positions in the parallax region (the four corners and its center) it was allocated for the trial. Positions were chosen so that the distance to travel was kept constant for all conditions. A random initial rotation was chosen for the goal brick and a subsequent rotation of  $90^\circ$  was applied to the manipulable brick. As the brick does not have an explicit “forward” orientation, either of the two possible  $180^\circ$  orientations was considered valid. Each participant experienced the same combinations of positions, although these were presented to them in a randomised order. The order of presentation of the techniques was counter-balanced.

Users started trials by pressing a *start* button (and waiting a countdown of 3 seconds) and completed them after pressing the *stop* button. This ensured that participants returned their arms to a neutral position between trials. Only times between the first and last touch were considered. Any interval between the completion of the trial and the button press was not part of the measurement. In this way we wanted to be sure that the participant decided to complete

the trial as a conscious choice and not as a product of chance. In order to avoid fatiguing participants, each trial timed out after 120 seconds. To provide more depth cues, we added a grid-like texture to the walls, and applied a shadow mapping algorithm to project shadows on the ground through a directional light source. Collision detection was disabled for the two bricks but enabled for the walls in order to prevent users from inadvertently losing the object. In total we had 90 trials per participant.

#### 4.2 Participants

Twelve participants (7 male, 5 female) aged 22-42 ( $M = 28.18$ ,  $SD = 6.13$ ) took part to the first study. All had prior experience with 3D films (42% watching at least 4 per year). All but two participants had some prior experience with 3D applications ( $M = 3.3$ ,  $SD = 1.57$ ), expressed on a scale ranging from 1 (very low experience) to 7 (very high). Two participants had previously used a stereoscopic display. All but one participant had high prior experience with multi-touch devices ( $M = 6$ ,  $SD = 1.41$ ). One participant was left handed, the rest was right handed. Six participants had corrected vision; they were tested for any stereopsis disorders by reporting the random order in which three objects appeared in depth, from farthest to closest. Each participant received £8.

#### 4.3 Procedure

We let each participant practise the input mappings until they felt comfortable. In addition, three groups of five trials were added with both objects placed in the same parallax region (thus requiring only *xy*-translation). These were considered as training and were not part of the analysis. After each technique block, participants filled two custom questionnaires on the viewing and interaction experience. The study lasted approximately 60 minutes in total.

#### 4.4 Results

In summary, results show that *Indirect4* was faster than the *Triangle Cursor* and comparable to *DS3*. Furthermore, *Indirect4* was reported to not affect the quality of the vision in a significant way, if compared to the other two techniques.

Excluding the 15 training trials, a total of 1080 trials were recorded; 76 of these (7%) were considered invalid due to incorrect positioning or timed out and excluded from further analysis. Respectively, *DS3* resulted in 2 trials (1%) missing, *Triangle Cursor* in 53 (15%) and *Indirect4* in 21 (6%). The majority of missing values in the case of the *Triangle Cursor* are due to participants’ frustration in completing specific trials and skipping them after attempting completion for a minimum of 120 seconds. Completed trials that took longer than 120 seconds were also excluded from the analysis. This time limit was necessary to keep the duration of the study manageable. The higher number of missed trials in *Indirect4* may be due to lower familiarity with the technique and issues with integrated input. It may thus require more training as opposed to *DS3*, which as noted was more representative of direct touch techniques users were already familiar with. From our observations, these issues were more evident in the *Triangle Cursor*.

Due to the presence of missing values we analysed the data using Linear Mixed Models. Repeated-measures ANOVA would require either the removal of every participant who had at least one missing measure or the reduction of the dataset to the lowest amount of common measures, resulting in a considerable loss of power. Linear Mixed Models do not suffer from this drawback.

##### 4.4.1 Completion Ratio and Times

Table 1 shows LMM results of the main effects of *technique* and *block* on completion times (see Fig. 6). Pairwise comparisons found significant differences ( $p < 0.01$ ) between *Indirect4* ( $M = 15.48s$ ,  $SD = 10.12$ ) and the *Triangle Cursor* ( $M = 19.82s$ ,  $SD = 14.62$ ). No significant differences were found between *Indirect4* and

<sup>1</sup><http://www.sharpxd.com>

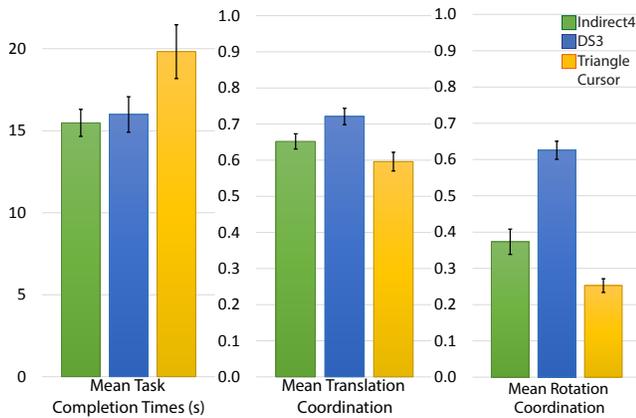


Figure 6: Overall mean values for completion times (left), translation coordination (middle), rotation coordination (right) reported in the first study. Error bars represent 95% confidence intervals.

*DS3* ( $M = 15.99s, SD = 7.96$ ). A significant effect for *block* was observed on completion times between the first measured block (after the training trials) and the other ones ( $p < 0.01$ ).

We observe that under our experimental conditions, the use of a touch-enabled external device resulted in better performance than with the *Triangle Cursor*, which was specifically designed for interaction in stereoscopic displays. From observations made during the experiment, we believe the reason for the slower performance is the integration of translation together with yaw rotation. *Triangle Cursor* introduces unwanted rotation errors while trying to translate the item and vice versa during rotation. These errors always had to be corrected resulting in additional frustration as highlighted by the qualitative results. Our results provide further support to Nacenta et al.’s conclusions [21] that separability is an important attribute of multi-touch manipulation. Although *DS3* and *Indirect4* have comparable mean completion times, further qualitative analysis (detailed successively) has confirmed the presence of disturbances of the stereoscopic perception we expected in the direct touch condition.

In the direct conditions, selection was almost instantaneous (although slower for the *Triangle Cursor*, due to requiring an extra tap after having placed the first two fingers), whereas in the indirect condition, every time participants lifted their fingers, they had to re-select the object. Due to the absolute mappings, if the object did not have a large contact area on the screen, successful selection became a result of trial and error. As a consequence, in the second study we switched to relative mappings and also measured the net manipulation time (from selection to completion).

IV	DV	F/df	Value	p-value
Technique	Time	2, 963.36	18.30	$p < 0.01$
Direction	Time	5, 963.21	10.02	$p < 0.01$
Block	Time	5, 963.08	5.68	$p < 0.01$
Technique	$T_c$	2, 903.64	27.47	$p < 0.01$
Direction	$T_c$	5, 903.54	4.00	$p < 0.01$
Technique $\times$ Dir.	$T_c$	10, 903.75	3.06	$p < 0.01$
Technique	$R_c$	2, 903.16	275.46	$p < 0.01$
Direction	$R_c$	5, 903.13	4.63	$p < 0.01$
Technique $\times$ Dir.	$R_c$	10, 903.18	3.04	$p < 0.01$

Table 1: LMM results for Time, Translation and Rotation Coordination

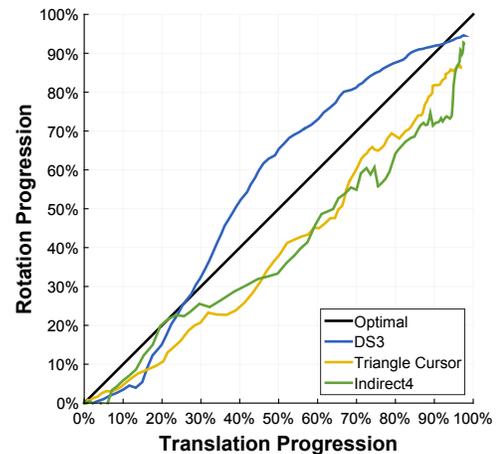


Figure 7: Mean translation-rotation coordination for the three techniques in the first study.

#### 4.4.2 Translation and Rotation Coordination

Translation ( $T_c$ ) and rotation coordination ( $R_c$ ) are metrics introduced by Zhai et al. [31]. They compare the length of the shortest distance to the goal to the distance actually travelled. In the case of  $R_c$ , it represents the initial mismatch between the two orientations, and the total rotation performed. Thus, a value close to 1 represents high coordination without unnecessary movements, close to 0 otherwise. Table 1 shows the main effects on  $T_c$  and  $R_c$ . *DS3* has better translation and rotation coordination than both *Indirect4* and the *Triangle Cursor* (see Fig. 6). The integration of translation and rotation in the *Triangle Cursor* also impacted  $T_c$  and  $R_c$ .

The reason of the under-performance of *Indirect4*, we believe, is due to insufficient training time. Participants were instructed to focus on the screen rather than on the external device as its display was blank. We observed how some participants performed some movements expecting different results. In particular, the indirect finger which controlled  $z$ -translation and worked in the same way both in *DS3* and *Indirect4* caused some confusion. In the original *DS3* implementation, upward movements caused the object to move further inside the screen. Some participants were observed performing a downward movement expecting the object to go instead into the screen and being surprised at the opposite result. Participant #8 also commented “I thought that moving my [direct] finger up I’d control the depth.” Whereas this was less problematic in the direct condition. This and the fact that yaw rotation was integrated together with  $z$ -translation decreased  $T_c$  and  $R_c$  performance.

We instructed participants to focus on speed and accuracy equally. However, by analysing the final progression towards the two goals (measured after pressing the stop button) we can obtain an indication of the accuracy of the three techniques. Pairwise comparisons found that *DS3* and *Indirect4* were significantly more accurate ( $p < 0.01$ ) than the *Triangle Cursor* in both translation (97.7%, 97.5%, and 96.8%, mean final progression, respectively) and rotation (94.5%, 92.5%, and 86.3%) but comparable with each other.

Fig. 7 shows the translation-rotation ratio graph [18, 31] obtained by resampling the average  $T_c$  and  $R_c$  values (from all participants and trials) across 100 equally spaced points. It provides an overview of the strategies adopted by participants to complete the task. The curve represents the progression over time along the two goals of matching both translation ( $x$ -axis) and rotation ( $y$ -axis). The curve begins in the lower-left corner, where the objects are in their starting positions and ends in the upper-right corner, where the object is ideally perfectly positioned and oriented. An optimal coordination,

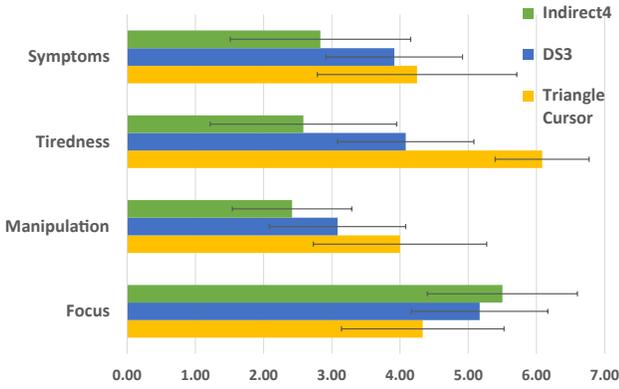


Figure 8: Overall mean values for vision related *symptoms*, *tiredness*, effect on *manipulation* strategy, and effect on user *focus*. Error bars represent 95% confidence intervals.

shown by the black line, means that participants increase their translation and rotation progress equally. A visible difference exists in the way rotation progresses in *DS3* compared to *Indirect4* and the *Triangle Cursor*. Due to the former separating rotation from translation, participants tended to reduce the rotation mismatch earlier, while the manipulable object is still distant from the goal (i.e. they reduced the rotation mismatch by 70% after having covered half the distance in *DS3* while in *Indirect4* and the *Triangle Cursor*, the reduction in rotation is between 30% and 40%). In *Indirect4*, there is an abrupt reduction of rotation mismatch from 70% to 90% while the object is already within the threshold translation distance. We can explain this by assuming that participants preferred to use *DS3* to manipulate the object when it had a larger contact area at the start of the trial, as also found by Liu et al. [16]. Whereas in *Indirect4*, participants preferred to match the orientation after having translated the object to its target location.

#### 4.5 Qualitative Results

For each technique participants completed a custom questionnaire investigating how comfortable each technique was in terms of viewing experience and interaction support (see Fig. 8). When asked if they experienced any vision-related *symptoms* such as double vision or dizziness, we found a significant difference ( $p < 0.05$ ) between *Indirect4* (which received a mean score of 2.83,  $SD = 2.08$ ) on a 7-point scale (1 no symptoms, 7 severe symptoms), and *DS3* ( $M = 3.92, SD = 2.43$ ) and the *Triangle Cursor* ( $M = 4.25, SD = 2.30$ ). *Indirect4* was also significantly less *tiring* ( $M = 2.58, SD = 2.15$ ) than the *Triangle Cursor* ( $p < 0.01, M = 6.08, SD = 1.08$ ) and trended towards significance if compared to *DS3* ( $p = 0.05, M = 4.08, SD = 1.78$ ). We also asked how much each technique affected their *manipulation* strategy (1 not at all, 7 completely) and whether they *focused* more on the position of their hands or of the object (1 hands, 7 object) but we did not find any significant differences.

*Indirect4* limits the frequency of interactions that could lead to breaking the stereoscopic effect (such as interpenetrating 3D objects) and provides a more comfortable viewing experience than *DS3* with comparable performance. The *Triangle Cursor* technique was reported to be the worst in terms of quality of vision. However, a limiting factor in our study was that, in order to compare all three techniques on equal grounds, we adopted a vertical setup with a frontal viewpoint. In their original work, the authors of the *Triangle Cursor* [26] adopted a top-down viewpoint and a shallow depth which might provide a less demanding cognitive effort as *xy*-movements on the 2D screen correspond to changes to the 3D *xz*-plane. With a frontal viewpoint, as the 3D *xy*-plane is orthogonal to

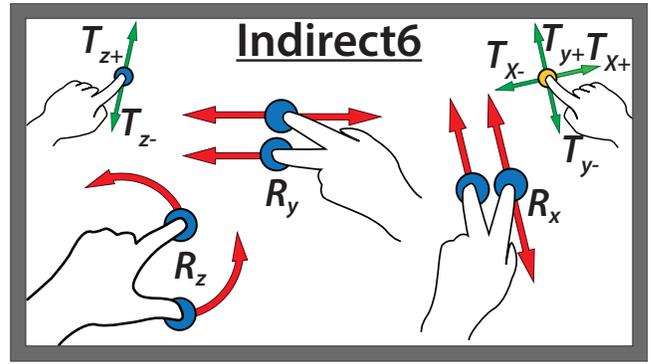


Figure 9: Touch gesture mappings for *Indirect6*. Translation in *xy* and *z* works analogously to *DS3*. Rotation gestures are enabled when one finger from the dominant hand is touching the surface. When two fingers from the other hand are moving in an approximately horizontal direction only yaw rotation is affected ( $R_x$ ); vertical movements affects pitch ( $R_y$ ); pivoting affects roll ( $R_z$ ).

the camera, users have to take into account the perspective projection. Thus, the *Triangle Cursor*'s strengths as an interaction technique for stereoscopic environments might be constrained to top-down projections. During post-study interviews, participants reported that they had to focus on the position of the fingers which controlled the *Triangle Cursor* rather than on the object, which often caused the loss of the stereoscopic effect. Concerning *DS3*, participants reported that due to their hands obstructing the manipulable object they often had to rely on shadows to estimate the correct alignment. In *Indirect4*, participants commented that their attention was on the object itself much more frequently than when compared to *DS3*.

## 5 SECOND STUDY: 6 DOF DOCKING TASK

In the second study we implemented a 6 DOF docking task. We did not consider the *Triangle Cursor* as the first study showed it was the worst performing technique. *DS3* was thus unconstrained to allow full 6 DOF manipulation. Taking into account our previous findings we designed a new technique, *Indirect6*, with an indirect touch interaction paradigm in mind. We included in the comparison another direct technique, *Surface6*, an adaptation of the mappings used for *Indirect6* for direct touch input. With its inclusion we wanted to make sure that the comparison of direct and indirect interaction would not be affected by the use of different touch interaction modalities. Our goal was thus to evaluate how an indirect technique would perform on a complex 6 DOF task compared to direct touch and if a technique designed specifically for an external indirect touch device would perform better than if that same technique were adapted to direct touch interaction.

### 5.1 Indirect6 and Surface6

Techniques that build on the metaphor of “grabbing” an object become less intuitive while interacting on an external device, as there is no visible connection between the user’s fingers and the object. Thus, we designed *Indirect6* so that: 1) it would work with relative input, avoiding the issues that arose in the first study; 2) it would provide a fully separated rotation modality, due to the impracticality of adopting the same principle used in *DS3* and other techniques, which relied on placing two fingers in contact with the objects to be manipulated. Since the indirect-touch device is only used to sense touch input, it becomes difficult to estimate the area of the device which corresponds to the 3D object.

The cursor thus provided traditional mouse-like feedback, and disappears after selection. Regardless of where the user interacts on the

device, changes are relative to the current cursor position or to the object’s (when selected). De-selection is possible by double-tapping once selected. Translation mappings for *Indirect6* are analogous to the ones used by *DS3*. Rotation mode is enabled when two fingers from the non-dominant hand touch the surface (see Fig. 9), in addition to one finger from the dominant hand. In our implementation, rotations were relative to the world axes.

The technique differentiates between fingers from the dominant and non-dominant hand by checking the order with which they touch the surface. Similar to *Indirect4*, the first finger to touch the device is assumed to belong to the dominant hand, whereas the second finger is assumed to belong to the non-dominant hand and enables  $z$ -translation. With a third contact point the technique switches to translation and rotation mode. The two non-dominant fingers control one rotation axis, and the dominant finger can be kept still or moved to translate while rotating. However, although  $xy$ -translation and rotation could be integrated, we later observed that participants used it as a *clutch* and rarely performed 3 DOF manipulations concurrently.

In order to determine the rotation axis, the gesture detection algorithm averages the direction of motion for both fingers of the non-dominant hand over the course of five samples from the start of motion (out of sixty in a second). If the two averaged direction vectors point horizontally or vertically, yaw or pitch is enabled, respectively; if they point in opposite directions, roll is enabled. Once a rotation mode is enabled, the subsequent frames only use the relevant component (displacement in  $x$  or  $y$  for yaw and pitch; angle between the two fingers for roll) and ignore the others. We calibrated yaw and pitch rotations so that a gesture as long as  $\frac{1}{4}$  of the device’s height resulted in a  $90^\circ$  rotation in the direction of motion. The angle determined in the roll mode is used to calculate a 1:1 corresponding rotation. For example by placing the index and thumb fingers aligned at east-west cardinal points and rotating them anti-clockwise so that the fingers would now be aligned north-south, would result in a  $90^\circ$  anti-clockwise rotation around the  $z$ -axis parallel to the ground plane.

If the fingers stop for at least five frames (but are not lifted), the algorithm is reset, thus allowing to switch from yaw or pitch and vice versa. To avoid unintended movements, roll requires the user to first lift one finger, in order to switch to yaw or pitch. The sampling of these five frames takes about 8.33ms. Based on the observation of the recordings, we did not observe the gesture detection algorithm to cause false positives. All rotations are relative to the global axes of the scene, i.e. the same rotation will be applied to the object regardless of its current orientation.

We note that this requires one additional finger to activate rotation. However, it was not reported to cause fatigue in the indirect condition. In the future we plan to improve the technique by determining dominant and non-dominant fingers by analysing their motion also at the two-fingers stage. To differentiate between  $z$ -translation and rotation mode, the technique could check whether the two contact points are both moving (to enable rotation) and are also within a specified range (indicating that they belong to the same hand).

## 5.2 Task

A different group of participants took part to this study. We kept the same design and procedure of the first task and changed some details. We interviewed participants after the study but we did not repeat the questionnaires used in the first study. This second study lasted approximately 100 minutes.

We used the model of a cartoon biplane so that participants could easily tell when the model was upside down. We asked participants to overlap the manipulable and goal objects within the same distance and rotation thresholds as before. The manipulable object was rotated  $120^\circ$  towards a randomised axis from the initial, random, orientation of the goal object. In the indirect condition, the initial

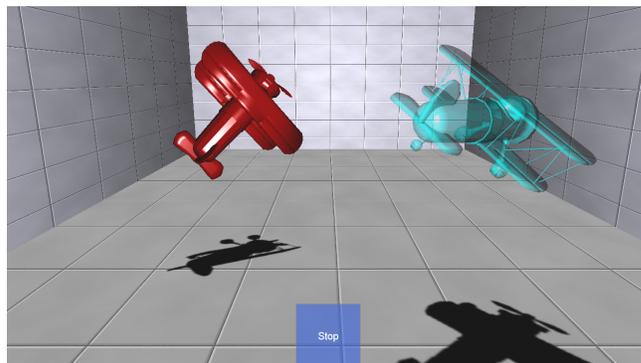


Figure 10: A trial from the second user study showing the manipulable biplane (in red) next to the goal biplane (in transparent blue).

IV	DV	F/df	Value	p-value
Technique	Time	2, 1089.07	21.96	$p < 0.01$
Direction	Time	5, 1089.06	2.81	$p < 0.02$
Technique	$T_c$	2, 1089.08	11.20	$p < 0.01$
Direction	$T_c$	5, 1089.08	17.00	$p < 0.01$
Technique	$R_c$	2, 1089.12	8.56	$p < 0.02$
Direction	$R_c$	5, 1089.12	2.16	$p < 0.05$

Table 2: LMM results for Time, Translation and Rotation Coordination

location of the cursor was reset to the position of the start button after each trial.

## 5.3 Results

In summary, results show that indirect interaction allows comparable net manipulation times to direct touch. *Indirect6* was found to have better accuracy than *DS3*. Furthermore, *Indirect6* was also faster than its direct adaptation, *Surface6*.

### 5.3.1 Completion Ratio and Times

Fourteen participants (9 Male, 5 Female;  $M = 24.36, SD = 3.27$ ) took part to this study. Out of the 1260 recorded trials, a total of 53 trials (4%) were considered invalid due to incorrect positioning/orientation or timed out and excluded from further analysis. Respectively, *DS3* resulted in 15 trials (1.19%) missing, *Surface6* in 18 (1.43%) and *Indirect6* in 20 (1.59%). Table 2 shows the main effects on completion times. In terms of completion times, pairwise comparisons show a significant difference between *DS3* and the two new techniques: *Surface6* ( $p < 0.01$ ) and *Indirect6* ( $p < 0.02$ ), with *DS3* being faster in overall completion times (i.e. including both selection and manipulation). Recall, however, that in *Indirect6* participants had to move the cursor over the manipulable object to select it. In the two direct techniques, selection time is instantaneous ( $M \leq 0.08s$ ), whereas in the case of *Indirect6*, it is a substantial quantity ( $M = 1.62s, SD = 0.61$ , see Fig. 11). Thus, if only the time actually spent to manipulate the object is considered, then *DS3* and *Indirect6*’s performances become comparable. Net manipulation times for *DS3* are 25.90s ( $SD = 15.88$ ); *Indirect6* 26.13s ( $SD = 15.57$ ); *Surface6* 32.14s ( $SD = 20.98$ ). We include this analysis because this study focused on the manipulation task itself. The selection technique used might vary depending on the specifics of the interaction scenario: objects might be always selected or specified through a system-control task (for example, as a result of a filtering operation) rather than being selected directly. However, the requirement of a cursor poses a clear disadvantage. It is conceivable

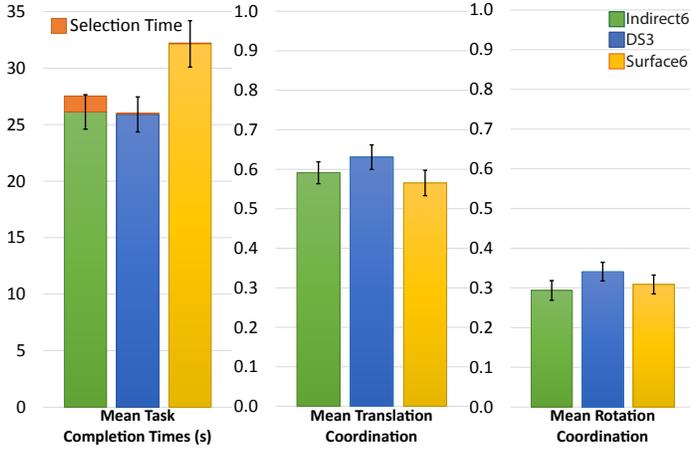


Figure 11: Overall mean values for completion times (selection time in red; left), translation coordination (middle), rotation coordination (right) reported in the second study. Error bars represent 95% confidence intervals (in the first chart they refer to net manipulation times).

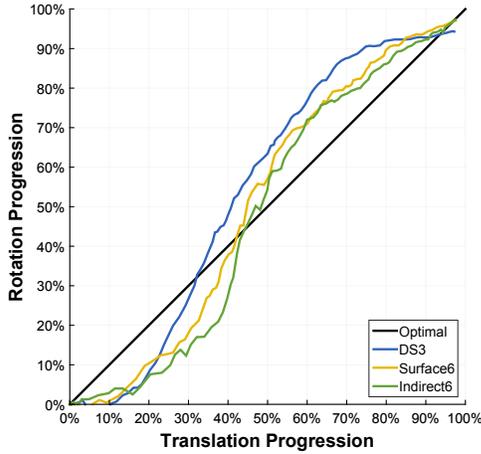


Figure 12: Mean translation-rotation coordination for the three techniques in the second study.

that better performing selection techniques for indirect devices might be devised in the future.

### 5.3.2 Translation and Rotation Coordination

*DS3* was significantly different than both *Surface6* and *Indirect6* in terms of  $T_c$  and  $R_c$  ( $p < 0.01$ ). Mean scores were respectively: 0.63 and 0.34 for *DS3*, 0.56 and 0.31 for *Surface6*, 0.59 and 0.29 for *Indirect6*. A significant effect of *direction* was found but no interaction with *technique* was reported. Nonetheless, translations and rotations taking place mainly in negative regions had a significant difference than those occurring in other regions. An analysis of the final measured progression towards the goal indicates that the two indirect techniques achieved a more accurate position and orientation ( $p < 0.01$ ) than *DS3*. The difference is greater in terms of orientation (*Indirect6*: 97.3%, *Surface6*: 97.1%, and *DS3*: 94.3%) than translation (97.6%, 97.8%, and 97.2%, respectively).

Regarding  $R_c$ , they were obtained with two radically different modalities: totally integrated in *DS3* and strictly separated in *Indirect6*. Qualitative data from post-hoc interviews highlighted that

Technique	1	2	3	4	5	6
DS3	22%	59%	16%	1%	1%	1%
Surface6	23%	43%	20%	12%	1%	0%
Indirect6	36%	38%	18%	7%	1%	0%

Table 3: Mean NDC values for each technique

Mode	DS3	Mode	Surface6	Indirect6
1d	35%	1d	60%	42%
1d + 1i	11%	1d + 1i	15%	17%
2d	54%	1d + 2m	25%	38%
2d + 1i	< 1%			

Table 4: Mean percentage of time spent in each mode.

*DS3* felt more *intuitive* due to the directedness of the interaction but at times became unpredictable due to the small amount of error introduced with each rotation. On the other hand with *Indirect6*, participants knew precisely how to perform the three different types of rotation but they sometimes struggled to understand which rotations would have got them closer to the goal.

Fig. 12 shows the translation-rotation ratio graph for the techniques used in the second study. Users tended to perform rotation earlier in *DS3* as previously noted in the first study. Indeed, in *DS3*, we noted how participants tended to move the manipulable object closer to them (if it was distant) so that they could rotate it with more ease. This supports similar observations noted by Liu et al. [16] where participants found difficulties rotating distant objects that had a smaller contact area.

### 5.3.3 Metrics

As the second study focused on a more complex task requiring use of 6 DOF, we thus wanted to evaluate the level of DOF coordination between the three techniques. Masliah et al. [19] introduced a measure called M-metric which describes the rate of simultaneous error reduction occurring in multiple DOFs. Our results are in line with those of Martinet et al. [18]: we found low values ( $\leq 0.1$ ) for the M-metric apart from the  $xy$  subset ( $DS3 = 0.24$ ,  $Surface6 = 0.21$  and  $Indirect6 = 0.2$ ).

The NDC metric (*Number of DOF Combined* [29]) represents the number of DOF interested by each instantaneous manipulation. From this value it is possible to compute the time spent in each interaction mode. Table 3 shows the amount of time  $n$  DOF were manipulated together. Although all three techniques allow the integration of translation and rotation, we observed how in practice participants used this mode for rotations alone, keeping still the finger assigned to translation. Table 4 shows the amount of time spent in each mode. Mean translation and rotation mode times for *DS3* and *Indirect6* are very close but a substantial difference exists between *Indirect6* and *Surface6*, two techniques that differ in terms of input modality only. *Indirect6*'s rotation mode led participants to split their usage between its translation and rotation mode almost equally, while *Surface6* has a greater proportion of time spent in its translation mode. Recalling *Surface6*'s slightly worse  $T_c$  measure and its slower task completion times, our observations suggest that this difference might be due to occlusion issues [25]. While *DS3* does not require the user to have both arms extended at all times, in *Surface6* users need to have both their arms extended to rotate the object. As the technique was designed to be used on an external device rather than directly on the screen, the chance of potentially occluding the goal object increases while in the direct adaptation. Indeed, we observed users moving the manipulable object out of the way in order to look at the goal object from a better viewpoint.

## 6 DISCUSSION

In this work, we investigated the potential of indirect multi touch interaction as an alternative to direct touch for 3D stereoscopic displays. Knoedel et al.’s study [15], showed slower performances of an adapted indirect 3D RST technique compared to its direct equivalent on a monoscopic display. In a previous study on stereoscopic display [25], we found that indirect touch led to 30% less error than direct touch due to occlusion in a 3 DOF task. In this work, we wanted to evaluate the performance of indirect multi-touch interaction for tasks involving more than three DOF. We investigated the impact of this choice on the quality of the viewing experience and on performance. Finally, we provide an overview of the advantages and limitations of the indirect touch interaction paradigm.

### 6.1 Quality of the Viewing Experience

In the first study, we asked participants to perform a 4 DOF docking task using three techniques. *DS3* was representative of direct touch metaphors and thus assumed to be the most likely to cause viewing discomfort; the *Triangle Cursor* represented an alternative way that, by virtue of indirectly specifying a 3D position, aimed to overcome the issues caused by direct touch. We designed our indirect touch interaction technique, *Indirect4*, on the idea of removing the cause of these issues by interacting on an external device. We learned that these vision-related issues, although frequently reported by participants, did not impede performance. Completion times were comparable across the two conditions. The *Triangle Cursor* was the most distressful technique for the eyes. Due to its integrated nature, participants focused on their fingers instead of the object, thus incurring in vergence-accommodation problems. Comments reported on the questionnaires and during interviews show that the direct techniques were described as “hurting [their] eyes” (*DS3*, participant #2, 1<sup>st</sup> study) or “causing discomfort and double-vision” (*Triangle Cursor*, #12, 2<sup>nd</sup> study). *Indirect4* was described as “untiring and enjoyable” (#9, 1<sup>st</sup> study), as also reported in a previous study [25]. We did not repeat the questionnaires in the second study.

### 6.2 Performance

The results of the two studies show that, in a 4 DOF task, indirect touch interaction does not cause any substantial drop in performance; the two modalities were comparable. In a complex 6 DOF task, *Indirect6* resulted in a mean net manipulation time comparable to the mean net manipulation time of the direct technique. Furthermore, the significantly different results obtained by *Indirect6* in relation to *Surface6*, show that techniques explicitly designed to take advantage of an indirect paradigm do not provide comparable level of performance when they are adapted to direct interaction.

The separation of all rotation axes in *Indirect6* resulted from the difficulty of adapting metaphors based on a direct connection between the user’s fingers and the underlying object (such as *DS3*, *Sticky Tools*, or *Screen Space*). These metaphors cannot be transferred to indirect touch interaction, as the user would ideally be focusing on what is happening on the screen rather than looking at the position of her/his hands on the external touch device. Although it resulted in worse  $R_c$  ratios, overall manipulation performance was not affected in a substantial way. *Indirect6* requires users to decompose a 3D rotational task along its individual 1D components. This caused users to spend more time handling each axis individually. A better solution might lie in combining both approaches, following conclusions reached in studies by Veit et al. [29] and Balakrishnan et al. [1]. Indeed, they identify an initial phase in which users perceive the task in an integral way. When fine-tuned adjustments are needed, integrated rotation becomes more imprecise and individual control is needed. This is also highlighted by more accurate results obtained by *Indirect6* in terms of final reported  $T_c$  and  $R_c$  ratios. Thus allowing users the ability to perform both integrated rotations and fine-tune when needed could lead to better overall performances.

### 6.3 Advantages of Indirect Touch

*Scenarios* — In light of these results, indirect touch may be more suited for interaction scenarios involving predominantly objects at negative or positive parallax. Indirect touch will be less likely to break the stereoscopic effect. Due to occlusion issues already noted in a previous study [25], indirect touch might be also appropriate for collaborative manipulation of 3D objects from arbitrarily sized screens. Users can interact through indirect touch from a distance without occluding the interested parts of the screen to other collaborators. In our study the tablet was placed on a desk but it is not an operational requirement. One hand can hold the tablet leaving the thumb and the other hand free to interact.

*Ubiquity* — Direct touch 3D stereoscopic screens are rare outside research laboratories and specialist professional environments. On the other hand, multi-touch devices such as tablets and smartphones are ubiquitous. Indirect interaction has the potential of turning these devices into 4 or 6 DOF input devices.

*Performance* — These results were obtained by adapting direct touch techniques, such as in the case of *Indirect4*. *Indirect6* represents the first attempt at building on the limited pre-existing knowledge of indirect touch interaction by addressing usability issues highlighted during an experiment. It is thus conceivable that by investing more time in the design of new indirect touch interaction techniques it might be possible to improve their performance.

### 6.4 Limitations

*Choice of Indirect Touch* — Indirect input requires the use of an additional device if compared to direct touch. We believe this is counter-balanced by the benefits it brings in terms of the quality of vision. On the desktop, the mouse and keyboard are the most widely used indirect devices in 3D applications. Reasons to use touch devices include the affordance of more DOF and the lesser dependence on mode-switching than the mouse and keyboard. Our results show that indirect touch interaction is a compelling alternative in scenarios where its advantages are beneficial. We leave to future research the task of comparing indirect touch against these devices. However, indirect touch does not have to be an alternative to the mouse and keyboard. Past research has shown positive bi-manual interactions between pen and direct touch [5]: it would be interesting to study whether indirect touch interaction with the non-dominant hand can complement mouse input better than the keyboard.

*Relative and Absolute Mappings* — During the first study, we observed that *absolute* mappings added some overhead in terms of initial selection, which we addressed in the next study. We initially used absolute mappings for *Indirect4* because we wanted to evaluate an approach to the indirect touch paradigm with the least amount of intervention, compared to techniques that were built on top of prior knowledge in the area. Our results indicate that the use of *relative* instead of *absolute* mappings may be more suited for the 3D interaction scenarios we considered. For example, during the use of *Indirect4* we observed that participants looked at the tablet to understand the location of their fingers in relation to the tablet’s screen and thus to their mapping to the main screen. Relative mappings introduce some issues, namely: requirement of the initial selection of a target and less awareness of the boundaries of the device, as reported in Simeone et al.’s study [25].

Although this was not formally evaluated in the studies we ran, we observed how users expected relative mappings while interacting with the tablet. Relative mappings allow the user to reinforce the connection made between the object being manipulated and the actions on the device, regardless of the exact location in which they are performed. This allows more room for performing gestures: for example to rotate an object in *DS3*, users need to place two fingers on the object, which becomes problematic with objects that appear small on the screen [16] as the fingers have to be close together. Indirect touch is not affected by this issue.

*Selection* — Although net manipulation times of *Indirect6* and *DS3* were comparable, the use of a relative cursor makes selection a slower task if compared to directly touching the object one wishes to manipulate. We thus note that selection tasks represent a challenge for indirect touch interaction that warrants further research.

*Integration and Separability* — *Indirect4* introduces some unnecessary rotations when affecting the object's depth and vice versa. This problem might be averted by requiring a second finger (as in *Indirect6*) to activate yaw rotation. *Indirect6* will encounter difficulties in interpreting diagonal dragging movements while rotating. This can be corrected by stopping and changing direction of motion.

## 7 CONCLUSION AND FUTURE WORK

We proposed a novel approach based on the use of indirect multi-touch interaction and evaluated it in two user studies. We found *Indirect4* to be faster than the *Triangle Cursor*, and comparable to a version of *DS3* constrained to 4 DOF input. However, indirect interaction can also mitigate common stereoscopic issues affecting the overall quality of vision. We used these previous results to design a new technique, *Indirect6*, which we compared to *DS3* on a full 6 DOF task. We found no significant differences in net manipulation times between *Indirect6* and *DS3*'s direct touch, showing that there are no drawbacks when adopting this interaction style.

Our research is but an example within the largely unexplored design space of indirect techniques. Further research will be needed to unlock its potential. Future challenges include expanding our knowledge of how to use touch devices at both ends of the size scale (i.e. smartphones or tables) for indirect 3D interaction and improving indirect touch selection performance. Combining indirect touch with other devices in bi-manual interaction is another potential direction.

We believe that indirect and direct touch are not in opposition, as they have different strengths and weaknesses. Rather, we believe that by studying areas where indirect performs better than direct touch and vice versa, we can expose and then address issues in either modality, leading to a mutually beneficial synergy.

## ACKNOWLEDGEMENTS

The author was supported by a Marie Curie Intra European Fellowship (*3STARS*) within the 7<sup>th</sup> European Community Framework Programme.

## REFERENCES

- [1] R. Balakrishnan, T. Baudel, G. Kurtenbach, and G. Fitzmaurice. The rockin' mouse: integral 3D manipulation on a plane. In *Proc. CHI 1997*, pages 311–318. ACM.
- [2] H. Benko and S. Feiner. Balloon selection: A multi-finger technique for accurate low-fatigue 3D selection. In *Proc. 3DUI 2007*, pages 79–86. IEEE.
- [3] F. Bérard, J. Ip, M. Benovoy, D. El-Shimy, J. R. Blum, and J. R. Cooperstock. Did minority report get it wrong? Superiority of the mouse over 3D input devices in a 3D placement task. In *Proc. Interact 2009*, pages 400–414. Springer.
- [4] B. Bollensdorff, U. Hahne, and M. Alexa. The effect of perspective projection in multi-touch 3D interaction. In *Proc. GI 2012*, pages 165–172. CIPS.
- [5] P. Brandl, C. Forlines, D. Wigdor, M. Haller, and C. Shen. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In *Proc. AVI 2008*, pages 154–161. ACM.
- [6] G. Bruder, F. Steinicke, and W. Stuerzlinger. Touching the void revisited: Analyses of touch behavior on and above tabletop surfaces. In *Proc. Interact 2013*, pages 278–296.
- [7] G. Bruder, F. Steinicke, and W. Stuerzlinger. Effects of visual conflicts on 3D selection task performance in stereoscopic display environments. In *Proc. 3DUI 2013*, pages 115–118. IEEE.
- [8] L.-W. Chan, H.-S. Kao, M. Y. Chen, M.-S. Lee, J. Hsu, and Y.-P. Hung. Touching the void: direct-touch interaction for intangible displays. In *Proc. CHI 2010*, pages 2625–2634. ACM.
- [9] C. Forlines, D. Wigdor, C. Shen, and R. Balakrishnan. Direct-touch vs. mouse input for tabletop displays. In *Proc. CHI 2007*, pages 647–656. ACM.
- [10] A. Giesler, D. Valkov, and K. Hinrichs. Void shadows: multi-touch interaction with stereoscopic objects on the tabletop. In *Proc. SUI 2014*, pages 104–112. ACM.
- [11] M. Hancock, S. Carpendale, and A. Cockburn. Shallow-depth 3D interaction: design and evaluation of one-, two- and three-touch techniques. In *Proc. CHI 2007*, pages 1147–1156. ACM.
- [12] M. Hancock, T. Ten Cate, and S. Carpendale. Sticky tools: full 6DOF force-based interaction for multi-touch tables. In *Proc. ITS 2009*, pages 133–140. ACM.
- [13] O. Hilliges, S. Izadi, A. D. Wilson, S. Hodges, A. Garcia-Mendoza, and A. Butz. Interactions in the air: adding further depth to interactive tabletops. In *Proc. UIST 2009*, pages 139–148. ACM.
- [14] D. M. Hoffman, A. R. Girshick, K. Akeley, and M. S. Banks. Vergence–accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of Vision*, 8(3), 2008.
- [15] S. Knoedel and M. Hachet. Multi-touch RST in 2D and 3D spaces: Studying the impact of directness on user performance. In *Proc. 3DUI 2011*, pages 75–78. IEEE.
- [16] J. Liu, O. K.-C. Au, H. Fu, and C.-L. Tai. Two-finger gestures for 6DOF manipulation of 3D objects. *Comput. Graph. Forum*, 31(7):2047–2055, 2012.
- [17] A. Martinet, G. Casiez, and L. Grisoni. The design and evaluation of 3D positioning techniques for multi-touch displays. In *Proc. 3DUI 2010*, pages 115–118. IEEE.
- [18] A. Martinet, G. Casiez, and L. Grisoni. Integrality and separability of multitouch interaction techniques in 3D manipulation tasks. *IEEE Trans. Vis. Comput. Graphics*, 18(3):369–380, 2012.
- [19] M. R. Masliah and P. Milgram. Measuring the allocation of control in a 6 degree-of-freedom docking experiment. In *Proc. CHI 2000*, pages 25–32. ACM.
- [20] J. P. McIntire, P. R. Havig, and E. E. Geiselman. What is 3D good for? a review of human performance on stereoscopic 3D displays. In *SPIE Defense, Security, and Sensing*. SPIE, 2012.
- [21] M. A. Nacenta, P. Baudisch, H. Benko, and A. Wilson. Separability of spatial manipulations in multi-touch interfaces. In *Proc. GI 2009*, pages 175–182. CIPS.
- [22] T. Ohnishi, N. Katzakis, K. Kiyokawa, and H. Takemura. Virtual interaction surface: Decoupling of interaction and view dimensions for flexible indirect 3D interaction. In *Proc. 3DUI 2012*, pages 113–116. IEEE.
- [23] J. L. Reisman, P. L. Davidson, and J. Y. Han. A screen-space formulation for 2D and 3D direct manipulation. In *Proc. UIST 2009*, pages 69–78. ACM.
- [24] U. Schultheis, J. Jerald, F. Toledo, A. Yoganandan, and P. Mlyniec. Comparison of a two-handed interface to a wand interface and a mouse interface for fundamental 3D tasks. In *Proc. 3DUI 2012*, pages 117–124. IEEE.
- [25] A. L. Simeone and H. Gellersen. Comparing indirect and direct touch in a stereoscopic interaction task. In *Proc. 3DUI 2015*, pages 105–108.
- [26] S. Strothoff, D. Valkov, and K. Hinrichs. Triangle cursor: Interactions with objects above the tabletop. In *Proc. ITS 2011*, pages 111–119. ACM.
- [27] D. Valkov, A. Giesler, and K. Hinrichs. Evaluation of depth perception for touch interaction with stereoscopic rendered objects. In *Proc. ITS 2012*, pages 21–30. ACM.
- [28] D. Valkov, F. Steinicke, G. Bruder, and K. Hinrichs. 2D touching of 3D stereoscopic objects. In *Proc. CHI 2011*, pages 1353–1362. ACM.
- [29] M. Veit, A. Capobianco, and D. Bechmann. An experimental analysis of the impact of touch screen interaction techniques for 3D positioning tasks. In *Proc. VR 2011*, pages 75–82. IEEE.
- [30] C. Ware and P. Mitchell. Visualizing graphs in three dimensions. *ACM Trans. Appl. Percept.*, 5(1):2, 2008.
- [31] S. Zhai and P. Milgram. Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices. In *Proc. CHI 1998*, pages 320–327. ACM.