

The Design and Evaluation of 3D Positioning Techniques for Multi-touch Displays

Anthony Martinet — Géry Casiez — Laurent Grisoni

N° 7015

August 2009

 *rapport
de recherche*

The Design and Evaluation of 3D Positioning Techniques for Multi-touch Displays

Anthony Martinet* , Géry Casiez † , Laurent Grisoni ‡

Thème : User Interfaces : Input devices and strategies
Équipe-Projet Alcove

Rapport de recherche n° 7015 — August 2009 — 17 pages

Abstract: Multi-touch displays represent a promising technology for the display and manipulation of 3D data. To fully exploit their capabilities, appropriate interaction techniques must be designed. In this paper, we explore the design of free 3D positioning techniques for multi-touch displays to exploit the additional degrees of freedom provided by this technology. We present a first interaction technique to extend the standard four viewports technique found in commercial CAD applications and a second technique designed to allow free 3D positioning with a single view of the scene. The two techniques were then evaluated in a controlled experiment. Results show no statistical difference for the positioning time but a clear preference for the *Z*-technique..

Key-words: Multi-touch displays, 3D positioning task, direct manipulation

* e-mail: anthony.martinet@lifl.fr

† e-mail: gery.casiez@lifl.fr

‡ e-mail: laurent.grisoni@lifl.fr

Design et Évaluation des techniques de positionnement 3D pour Écran Multi-Points

Résumé : Les écran multi-points représentent une technologie prometteuse pour l'affichage et la manipulation de données 3D qui nécessite cependant la création de nouvelles techniques d'interactions. Dans ce document, nous nous intéressons spécifiquement aux techniques de positionnement 3D libre en étudiant la prise en charge des degrés de libertés additionnels que procure cette nouvelle technologie. Nous présentons dans un premier temps une technique qui étend l'usage de la technique à quatre vues que l'on rencontre traditionnellement dans les logiciels de CAD pour ensuite exposer une technique qui permet le positionnement libre avec une vue unique de la scène. Ces deux techniques sont ensuite évaluées dans une expérience contrôlée. Les résultats ne montrent aucune différence significative pour le temps de positionnement mais dénotent cependant une préférence certaine de l'utilisateur pour la Z-technique.

Mots-clés : Écran multi-points, tâches de positionnement 3D, manipulation directe

1 Introduction

Multi-touch input enables users to manipulate and display information in unprecedented ways by using fingers and hand gestures [20]. Multi-touch displays offer a direct interaction using fingers to translate, rotate and scale two dimensional (2D) objects [9, 27], pan and zoom a camera on a 2D map [18] or scroll information with judicious screen animation representing physical metaphor [2]. This technology provides a natural interaction with data in spite of accuracy [1, 3, 17] and occlusion [25, 19] problems.

Positioning is one of the most basic task of any three dimensional (3D) application. It allows users to explore the design space by repeatedly adjusting the position of objects. In desktop environments, the mouse remains the favorite input device to position objects in 3D space, usually using multiple orthogonal views of the scene to control two degrees of freedom (DOF) at a time. An alternative is to use a 3D input device to control the three DOF in an integral way but this technique suffers from accuracy, clutching and fatigue problems.

The development of 3D interaction techniques for multi-touch displays focus mainly on collaboration [13] and 3D object rotation [8]. However to the best of our knowledge, no interaction technique has been proposed to exploit multi-touch displays to freely position objects in 3D environments.

In this paper, we explore the development of 3D positioning techniques for multi-touch displays. How can we exploit the additional DOF provided by this technology to develop intuitive and efficient 3D positioning techniques? How to map finger gestures to 3D movements?

After reviewing the related work on 3D manipulation with multi-touch displays and 3D positioning with mouse-related devices, we present two new interaction techniques for 3D positioning. The first technique called multi-touch viewport extends the current 3 orthogonal views used in CAD softwares by adding multi-touch support. The second technique called *Z*-technique presents a way to map finger gestures to 3D object positioning using a single view on the 3D scene. The two techniques are then evaluated in a controlled experiment.

2 Related Work

2.1 3D Manipulation with Multi-Touch Inputs

Hancock et al. [8] presented one, two and three touch input interaction techniques to manipulate 3D objects on tabletop displays with limited depth. With one touch input, their technique achieves 5 DOF movement by extending the RNT algorithm [12] for 3D rotations coupled with 2D positioning. Two-touch interaction allows to perform pitch and roll rotations with a second finger on the non-dominant hand. With three-touch interaction, users can perform a simultaneous translation and rotation on the surface of the table. Positioning along the depth is proposed as an option by measuring the distance between two fingers. However the technique was mainly designed to control the orientation of a 3D object with 2D positioning. With the depth control position enabled, modifying the orientation of an object changes its depth position without any control. The three-touch technique was shown to be fastest and most accurate as well as preferred by the users.

Wilson et al. [26] presented a physics-enabled 3D environment with multi-touch input. Their technique is able of modeling both multiple contact points and more sophisticated shape information, such as the entire hand. This enables a large variety of interactions, supporting finger-based, whole-hand, and tangible input. They showed that their technique can be used to add real-world dynamics to interactive surfaces such as a vision-based tabletop. However adding physics to the interaction allows here to rotate objects in a more realistic way but positioning remains constraint to two DOF.

Using multi-touch indirect interaction, Zeleznik et al. [28] provided techniques that uses two pointers to translate and rotate objects using both hands. The non-dominant hand (NDH) is used to define an axis to rotate an object of interest with the dominant hand (DH). However, translation was constraint to the plane associated to the manipulated object.

2.2 3D Positioning

Jacob et al. showed that performance is improved when the perceptual structure of the task matches the control structure of the device [11]. According to Jacob et al.'s taxonomy, 3D positioning is an integral task that is best performed with 3D trackers as it is more natural to move diagonally across dimensions. However Theather and Stuerzlinger [22] presented a study comparing 3D positioning techniques with 2 and 3 integral DOF devices showing that input devices with two integral DOF can perform significantly faster than three integral DOF devices using "smart" algorithms. The design of these algorithms depends on the goal followed when positioning an object.

Positioning techniques can be classified into two categories: constraint based positioning where the position of the manipulated object is constrained by the attributes of the other objects of the scene and free positioning where the manipulated object can be freely positioned in 3D space. Constrained positioning

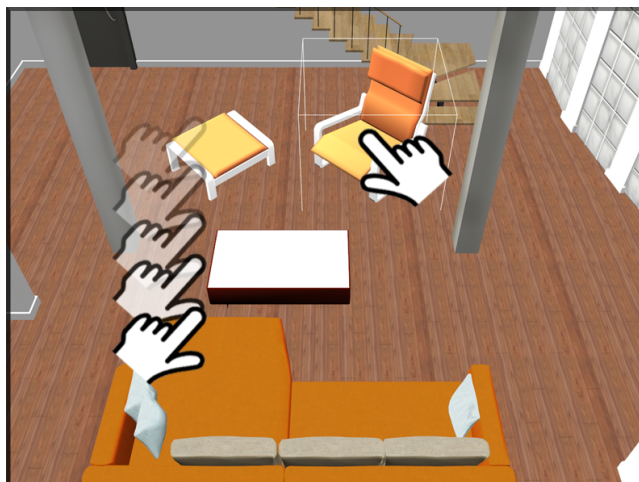


Figure 1: User view of the Z-technique. The right finger is used for direct 2D positioning in the camera plane while the left finger is used for depth positioning.

is used when the objects of a scene have a strong relationship between each other.

2.2.1 Constrained positioning

Constrained positioning techniques rely on the idea of finding some properties from the 3D scene to help positioning an object. Bier proposed Snap-Dragging [4], a technique that searches for visual features (i.e. a vertex, an object edge or a grid-line) close to the cursor position to snap the position of an object manipulated. This technique is used nowadays in commercial CAD applications. Later, Bukowski et al. [6] provided a technique that use the first visible surface behind an object as a constraint plane for translation. Their technique allows to group and manipulate several objects to ease scene building. Oh et al. [16] improved snap-dragging by using the whole area behind the object to search for visual features and determine its position. They proved that their technique was significantly faster than axis-handles used at that time in the software Maya.

In spite of the widespread of these techniques in CAD and sketching softwares, constrained positioning techniques introduce visual discontinuities when jumping from visual feature to visual feature making it difficult to use in a complex scenes. Free positioning techniques, similar to positioning with 3D trackers, provide an alternative when the objects have no clear relationships or when users want to freely control the position of the object manipulated.

2.2.2 Free positioning

Nielson and Olsen [14] proposed a technique to map two integral DOF to a 3D position. Their technique constraints object translation along a single axis defined by the greatest displacement along one of the axis of a coordinate system. This idea is still used nowadays in sketching program like Google Sketchup. Bier [5] provided a more general solution that consists in attaching a different coordinate system to an object to allow translation along axes of any orientation. One drawback of this technique is that the user needs to mentally decompose the scene, making it more difficult to use for novice users. Venolia [24] proposed to use the additional degree of freedom represented by the mouse wheel to control the depth position. His technique used several modalities like sound and visual feedback but was not evaluated.

3D widgets or handles are another alternative for 3D positioning that consist in breaking the displacement into single DOF [7, 21]. These widgets are still used in 3D commercial applications.

Finally, the four-views technique consists in displaying on screen 3 orthogonal views and one perspective view of the 3D scene. The object position is controlled 2 DOF at a time using one of the orthogonal view. This technique represents the most widely used technique to freely position 3D objects and has been shown to be faster than the handles technique with a single view [16].

To sum up, the interaction technique designed for multi-touch displays allow 3D objects positioning constrained to the screen plane and 3D rotations. 3D positioning techniques using two integral DOF can be separated into constrained and free positioning technique. The four-views techniques represents the most widely used technique for the latter category.

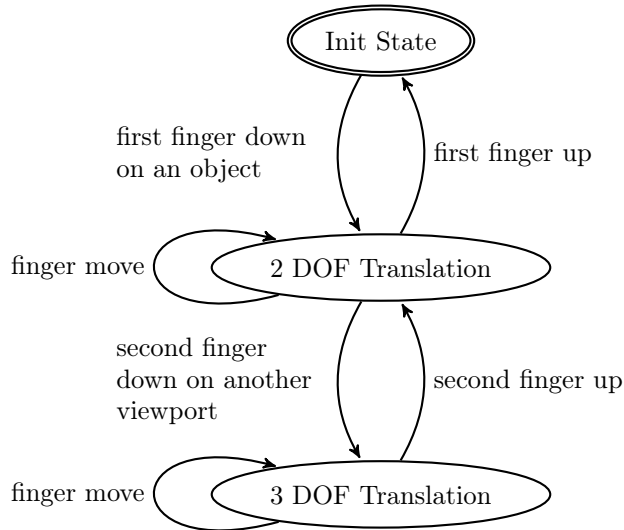


Figure 2: Multi-touch viewport state transition diagram

3 Multi-touch viewport and Z -technique

Our primary goal in mind was to focus our design on free 3D positioning techniques for multi-touch displays. During the design process we noticed that multi-touch displays can reach their full potential when used with direct interaction. As a result, we wanted to develop interaction techniques that extend current interaction techniques using the screen plane to control 2 DOF in a direct way. The question was then to control the third degree of freedom to change the objects depth position.

Among the techniques mentioned in the related work that allow free 3D positioning, the Venolia technique and the four-views technique appear to be the most compatible with direct interaction. This lead us to redesign these two techniques for multi-touch displays.

3.1 Multi-touch viewport

Most CAD softwares (Autocad, Archicad, Blender ...) propose a display technique dividing the screen into three or four views. One of these views presents a perspective view of the scene while the two or three other views are orthographic and orthogonal to each other. These different views are also referenced as viewports. Each orthographic viewport allows to control two DOF in an integral way while the perspective view usually allows positioning along the plane parallel to the ground.

The state transition diagram in Figure 2 describes the technique. It is based on *finger down*, *finger up* and *finger move* events coming from the table SDK. The ids associated to each finger are used to find which finger touched the table first and keep track of each individual finger.

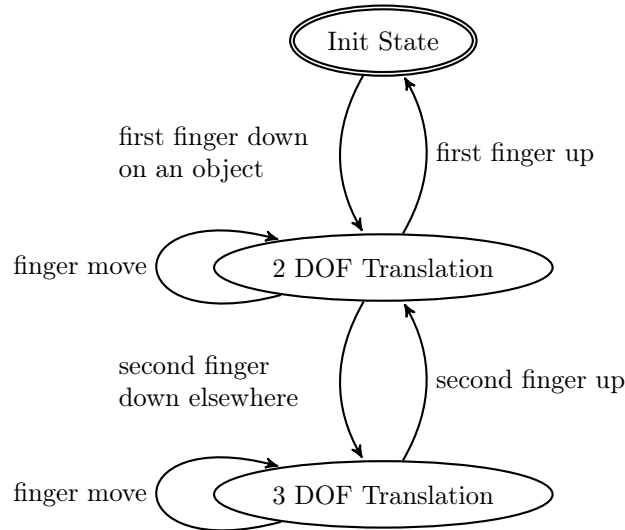


Figure 3: Z-technique state transition diagram

3.1.1 Standard viewport

The two top states in Figure 2 correspond to the standard viewport technique. When a finger first touches the surface, the corresponding viewport is detected and a ray orthogonal to the view is projected into the scene. The first object in intersection is returned. Subsequent finger movements move the object in the plane parallel to the view passing through the object center.

3.1.2 Multi-touch viewport

The multi-touch viewport extension is activated when a second finger touches another viewport. On the finger down event, the object is first teleported to the depth position associated to the second finger position. The user can then adjust the depth position in the direction orthogonal to the view associated to the first finger. We used a visual feedback represented as a line passing through the object center to represent this direction (Figure 4).

3.2 The Z-technique

We designed the second technique to allow 3D positioning using a single view of the scene (Figure 3). We believe this presentation increases the immersion of the user in the 3D environment compared to the multi-touch viewport technique. With this technique, we also use direct manipulation to control the position of objects in the screen plane. Backward and forward movements of a second touch point allow depth positioning. This technique is similar to the one proposed by Venolia [24] where the mouse wheel is used for depth positioning.

3.2.1 2 DOF Translation

When the first finger is in contact with the surface, a ray coming from the camera center through the finger position is casted in the scene and the closest object in intersection is returned. Positioning is then constraint to the plane parallel to the camera plane passing through the object center.

3.2.2 3 DOF Translation

When a second finger is in contact with the surface we measure its relative motion on the surface and use backward forward movement to control the depth position. Backward - forward movements are measured relative to the user position (in our implementation, we assume that the user position is known). Forward movement moves away the object from the user and backward movement moves it closer to the user. As we measure the relative motion, a succession of finger touch and release, also known as clutching, can be used to position the object at any depth position.

As we control the depth position in an indirect way, we use a transfer function to map the finger movement to object displacement. During preliminary tests we first used a constant gain but it appeared that fine positioning was difficult due to the low resolution of our tracking system. We then implemented a non-linear function that map the displacement speed to a scale factor. To a slow displacement speed corresponds a low scale factor and a high displacement speed is associated to a much higher scale factor. It helps fine positioning as well as gross positioning with reduced finger displacement. We designed the curve with two expert users to optimize the mapping between finger movements and object displacement.

3.3 Adding Orientation

The one-touch technique presented by Hancock et al. [8] as an extension of the RNT technique [12] for 3D objects can be used to provide orientation to the multi-touch viewport and *Z*-technique. With that algorithm, the first finger in contact with the object for the multi-touch viewport and the finger in direct contact with the manipulated object for the *Z*-technique controls both the positioning in the plane parallel to the screen and the orientation of the object. The development of efficient orientation techniques is out of the scope of this paper.

4 Experiment

4.1 Goals

The two techniques are similar for the direct positioning of objects in the plane parallel to the screen but they differ in the display of the 3D scene: the multi-touch viewport technique presents four views of the scene while the *Z*-technique shows a single perspective view. The techniques mainly differ in the control for the depth position: the multi-touch viewport uses direct interaction while the *Z*-technique controls the depth position in an indirect manner.

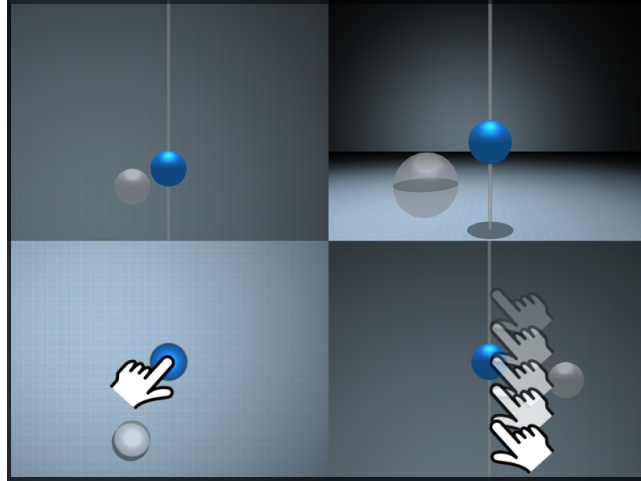


Figure 4: Screen capture of the multi-touch viewport technique. The left finger here is used for 2D positioning in the camera plane corresponding to the viewport while the right finger is used to teleport the object and adjust its depth. The gray line is used as visual feedback to represent the displacement allowed for the second finger.

We want to investigate the impact of the information presentation and the directness of control for depth positioning on both performance and qualitative feedback. The *Z*-technique encourages the simultaneous control of the 3 DOF while the multi-touch viewport seems to favor iterative control in each viewport in spite of the multi-touch support we added to the technique. We also expect the *Z*-technique to increase the feeling of immersion in the environment as the technique presents a single perspective view in full screen compared to the multi-touch viewport where the user focus on each orthographic view. Finally considering the visual presentation and the directness of control for each technique we want to evaluate the influence of the precision required for positioning as a confounding factor.

While we showed that these two techniques can be extended to support orientation we want to focus our evaluation of the positioning itself to avoid any cross interaction between the positioning and orientation tasks.

4.2 Apparatus

The experiment was conducted on an Immersion iLight touch table [10] based on the Diffused Illumination technique and consisting of a 100 cm × 70 cm (42 inches) monoscopic display positioned at a distance of 105 cm from the floor. The video-projector under the table was set at 60 Hz with a 1400 × 1050 pixels resolution giving a pixel density of 14 pixels per cm (36 DPI). A camera running at 120 Hz with a 640 × 480 pixels resolution and positioned under the surface recorded the finger movements on the surface. This gives a maximum resolution of 6.4 dots per cm (16.25 DPI) for finger tracking.

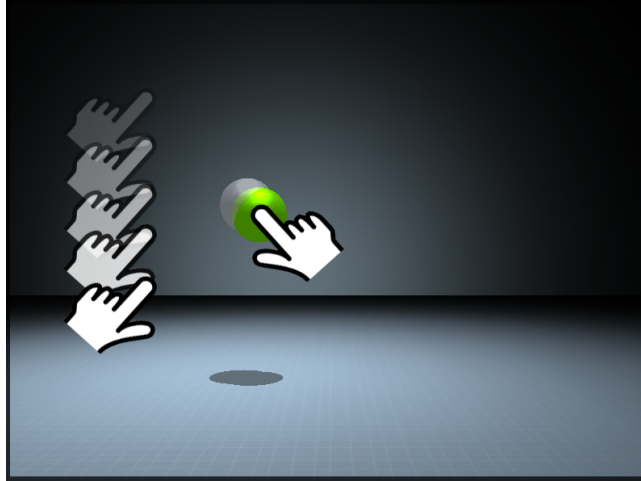


Figure 5: Screen capture of the *Z*-technique. The right finger is used for direct positioning in the camera plane while the left finger is used for depth positioning in an indirect way. Backward-forward movements move the object farther or closer to the user.

We used the iLight framework version 1.6 for fingers detection and tracking. Fingers data were then sent using TUIO messages [23] to a custom built 3D application based on the Ogre3D framework [15].

4.3 Task

The task is a three dimensional positioning task based on the docking task exposed by Zhai [30] (Figure 5). Each experimental trial began after the previous target was successfully positioned and ended with the successful positioning of the current target. Participants were asked to dock a blue sphere in a transparent grey sphere as quick as possible. The blue sphere turned green when positioned inside the grey sphere. This visual feedback helps the participants positioning the sphere in 3D [30]. The trial was considered as fully completed when the distance between the two spheres remained below a threshold distance during 0.8 s. The grey sphere then disappeared and appeared to another position while the green sphere was repositioned at the center of the screen. In addition to perspective and occlusion, we added a ground with shadows projection to improve depth perception. The camera remained fixed during the whole experiment.

4.4 Participants

Eight males with a mean age of 24 (SD 1.5) participated. 7 were right-handed and 1 was left-handed and all had normal or corrected to normal vision. Participants had a variable experience with virtual reality and 3D applications but this is acceptable as we are observing a lower level physical behavior. Half of them were regular users of the touch-table. The others were familiar with tactile

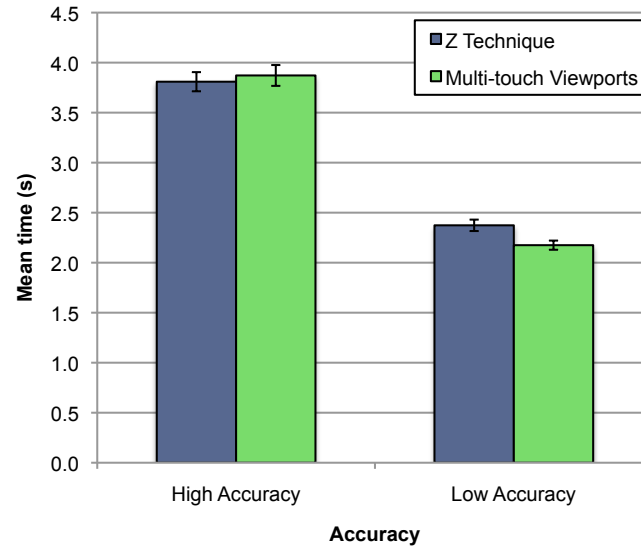


Figure 6: Mean positioning time for each technique and accuracy. Error bars represent 95% confidence interval.

technology such as tactile mobile phone or tablet pc but not with multi-touch displays.

4.5 Design

A repeated measures within-subjects design was used. The independent variables were TECHNIQUE (*Z*-technique and multi-touch viewport), POSITION of the target sphere with 17 positions evenly distributed around a hemi-sphere, and ACCURACY with two levels (low accuracy and high accuracy).

Participants completed five successive BLOCKS of trials. Each BLOCK consisted of 68 trials: 2 repetition of 34 POSITION \times ACCURACY combinations. The presentation order of TECHNIQUE was counter-balanced across participants. A break was encouraged after each set of 17 trials.

Before starting the experiment, participants had a 15 minute training period to get used to the techniques and task. The experiment ended with a qualitative feedback from the participants. The experiment lasted approximately 35 minutes.

In summary, the experimental design was :

8 participants \times
 2 TECHNIQUE \times
 5 BLOCKS \times
 2 ACCURACY \times
 17 POSITION \times
 2 repetitions
 = 5,440 total trials

5 Results

The dependent variables for both techniques were positioning time and coordination value. A third variable is the frequency of use of two viewports for the multi-touch viewport technique.

5.1 Positioning Time

Positioning time is defined as the time it takes to successfully position the current sphere from the last successfully positioned target.

Repeated measures analyses of variance found a significant main effect for BLOCK ($F_{3,21} = 5.68$, $p = 0.016$) which shows the presence of a learning effect. Pairwise comparisons show significant differences ($p < 0.05$) between the first block and the other blocks. As a result we removed the first block for subsequent analysis.

We did not find a significant main effect of TECHNIQUE on the positioning time. The mean positioning time for the Z-technique is 3.10s and 3.03s for the multi-touch viewport technique (Figure 6).

Repeated measures analyses of variance found a significant main effect of ACCURACY ($F_{1,7} = 330$, $p < 0.0001$) on positioning time. As we expected higher accuracy significantly increases the positioning time from 2.28s for the low accuracy condition to 3.86s for the high accuracy condition. More interestingly, we found a significant interaction between TECHNIQUE and ACCURACY ($F_{1,7} = 6.38$, $p = 0.039$) with the Z-technique being slower for the high accuracy condition (2.38s) compared to the multi-touch viewport technique (2.18s) ($p = 0.026$).

We found a significant effect for POSITION ($F_{16,112} = 8.11$, $p < 0.0001$) on positioning time but no significant interaction with other factors. It shows that the positioning is faster for the targets requiring no displacement along the depth.

5.2 Coordination

The coordination is computed as the ratio of the length of the shortest path to the length of the actual path measured from the starting point to the destination [29].

Repeated measures analyses of variance did not show any significant main effect or interaction for the different independent variables on the coordination except for POSITION ($F_{16,112} = 24.5$, $p < 0.0001$) showing a higher coordination for the targets requiring no depth adjustment. This result does not allow us to draw any conclusion. The mean coordination value is 0.59 for the Z-technique and 0.62 for the multi-touch viewport technique.

5.3 Use of a Second Viewport for the Multi-touch Viewport Technique

For the multi-touch viewport technique, we analyzed the use of a second viewport to teleport and adjust depth position. A second viewport is considered as used when two viewports are touched simultaneously at least one time during an individual trial. These trials are marks as 1 and used for computation. On

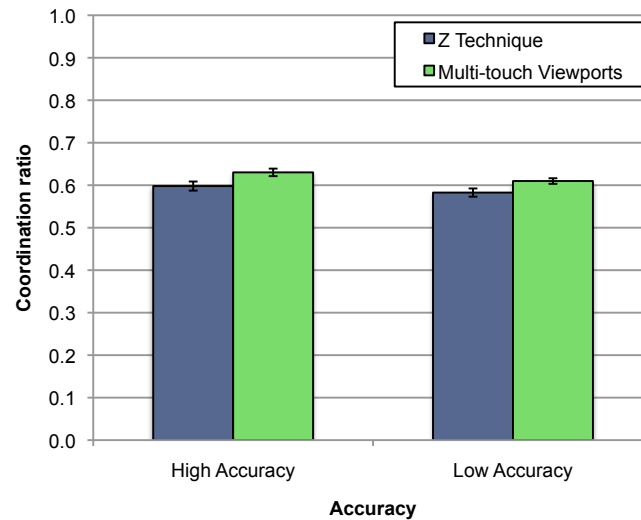


Figure 7: Mean coordination ratio for each technique and accuracy. Error bars represent 95% confidence interval.

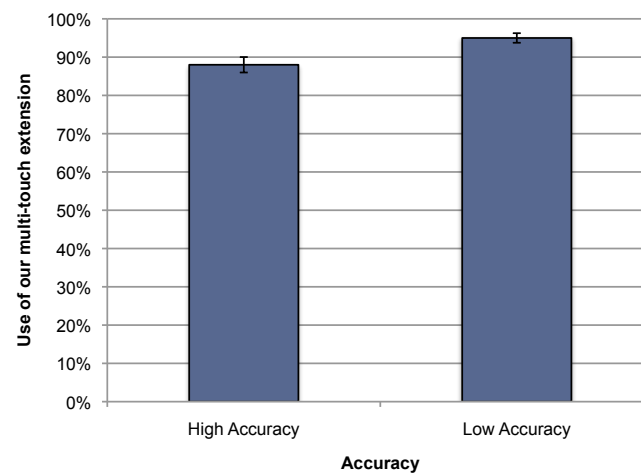


Figure 8: Probability of use of the multi-touch extension for the viewpoints. Error bars represent 95% confidence interval.

average, a second viewport was used 92% of the time. Repeated measures analyses of variance found a marginal significant main effect of ACCURACY ($F_{1,7} = 5.35$, $p < 0.054$) on the use of a second viewport with the second viewport used more for the low accuracy condition (95.5%) compared to the high accuracy condition (88.1%).

We also measured the frequency of use of each viewport and we found that the two viewports closer to the user are used 99.5 % of the time.

5.4 Qualitative Feedback

Among the participants, six preferred the *Z*-technique, one the multi-touch viewport technique and one claimed to have no preference. The participant who declared to prefer the multi-touch viewport found the technique easier to use but this participant reported a lot of experience with CAD applications implementing the viewport technique. The participants who declared to prefer the *Z*-technique found the technique easier to use and appreciated the interaction with the perspective view. The indirect mapping for depth positioning was according to them difficult to apprehend but very efficient when they get used to it.

We also asked the participants to rate the level of fatigue for each technique. Globally, we were not able to draw any specific conclusion on each technique. Tiredness was present in both techniques but the order of presentation appeared to favor the fatigue for the second technique evaluated in spite of the breaks during the experiment. The main conclusion we can draw is that working half an hour with the table is tiring for all user no matter the technique used.

For the *Z*-technique, all participants (including the left handed participant) used their dominant hand to control positioning in the camera plane while depth positioning was controlled with the non-dominant hand with the exception of one participant who used his thumb for depth positioning. Two participants made frequent confusion in the direction to use for depth positioning with backward movements expecting objects to move forward. In a real application, this parameter should be a control setting to be customized for each user.

6 Discussion

These results incline us to conclude that the two techniques evaluated are equivalent in term of performance as we did not find any significant difference for the positioning time. However further experiments would be required to prove that no small difference exists as we found a marginal effect for the high precision condition in favor of the multi-touch viewport technique.

In the experiment the participants used two viewports simultaneously more than 80% of the time, making an effective use of the additional features provided by the multi-touch viewport technique. In addition the results show that participants emphasize the simultaneous use of two viewports with targets requiring coarse positioning showing that when the task become more difficult, users tend to adjust sequentially the degrees of freedom. We also showed that the two top viewports were nearly never used with our hardware configuration, suggesting that participants tried to reduce their fatigue using the two viewports close to them. This suggests that the perspective view can be extended to the entire width of the upper half of the screen.

Oh et al. [16] showed that the classic four-view technique was significantly faster compared to other techniques using full screen in a manipulation task. Here we provide the *Z*-technique that reaches a similar level of performance compared to an improved version of the four-views technique. The interaction with the perspective view in full screen was in addition preferred by most participants. This makes the *Z*-technique a real alternative to the multi-touch viewport, especially when the display surface is limited (e.g. mobile phones).

7 Conclusion and Future Work

In this paper, we have introduced and compared two 3D positioning techniques that allow to freely position an object using a multi-touch display. In a controlled experiment we showed that these two techniques are equivalent in term of performance. However the *Z*-technique was preferred by most participants. Considering its performance and qualitative feedback, the *Z*-technique can be used as a base line comparison for other 3D positioning techniques designed for multi-touch displays. As future work, we plan to study and add the support of orientation to control more degrees of freedom in an intuitive way.

References

- [1] Pär-Anders Albinsson and Shumin Zhai. High precision touch screen interaction. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 105–112, Ft. Lauderdale, Florida, USA, 2003. ACM.
- [2] Apple. iPhone. <http://www.apple.com/iphone/>, 2007.
- [3] Hrvoje Benko, Andrew D. Wilson, and Patrick Baudisch. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 1263–1272, Montreal, Quebec, Canada, 2006. ACM.
- [4] Eric A. Bier. Snap-dragging in three dimensions. In *Proceedings of the 1990 symposium on Interactive 3D graphics*, pages 193–204, Snowbird, Utah, United States, 1990. ACM.
- [5] Eric Allan Bier. Skitters and jacks: interactive 3D positioning tools. In *Proceedings of the 1986 workshop on Interactive 3D graphics*, pages 183–196, Chapel Hill, North Carolina, United States, 1987. ACM.
- [6] Richard W. Bukowski and Carlo H. Séquin. Object associations: a simple and practical approach to virtual 3D manipulation. In *Proceedings of the 1995 symposium on Interactive 3D graphics*, pages 131–ff., Monterey, California, United States, 1995. ACM.
- [7] Brookshire D. Conner, Scott S. Snibbe, Kenneth P. Herndon, Daniel C. Robbins, Robert C. Zeleznik, and Andries van Dam. Three-dimensional widgets. In *Proceedings of the 1992 symposium on Interactive 3D graphics*, pages 183–188, Cambridge, Massachusetts, United States, 1992. ACM.
- [8] Mark Hancock, Sheelagh Carpendale, and Andy Cockburn. Shallow-depth 3d interaction: design and evaluation of one-, two- and three-touch techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1147–1156, San Jose, California, USA, 2007. ACM.
- [9] Mark S. Hancock, Sheelagh Carpendale, Frederic D. Vernier, Daniel Wigdor, and Chia Shen. Rotation and translation mechanisms for tabletop interaction. In *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, pages 79–88. IEEE Computer Society, 2006.

- [10] Immersion. Immersion. <http://www.immersion.fr/>, 2009.
- [11] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and Jr M. Preston Mullen. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1(1):3–26, 1994.
- [12] Russell Kruger, Sheelagh Carpendale, Stacey D. Scott, and Anthony Tang. Fluid integration of rotation and translation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 601–610, Portland, Oregon, USA, 2005. ACM.
- [13] Jun Liu, David Pinelle, Samer Sallam, Sriram Subramanian, and Carl Gutwin. TNT: improved rotation and translation on digital tables. In *Proceedings of Graphics Interface 2006*, pages 25–32, Quebec, Canada, 2006. Canadian Information Processing Society.
- [14] Gregory M. Nielson and Jr Dan R. Olsen. Direct manipulation techniques for 3D objects using 2D locator devices. In *Proceedings of the 1986 workshop on Interactive 3D graphics*, pages 175–182, Chapel Hill, North Carolina, United States, 1987. ACM.
- [15] OGRE. Open source 3D graphics engine. <http://www.ogre3d.org/>.
- [16] Ji-Young Oh and Wolfgang Stuerzlinger. Moving objects with 2D input devices in CAD systems and desktop virtual environments. In *Proceedings of Graphics Interface 2005*, pages 195–202, Victoria, British Columbia, 2005. Canadian Human-Computer Communications Society.
- [17] Alex Olwal, Steven Feiner, and Susanna Heyman. Rubbing and tapping for precise and rapid selection on touch-screen displays. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 295–304, Florence, Italy, 2008. ACM.
- [18] Jun Rekimoto. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, pages 113–120, Minneapolis, Minnesota, USA, 2002. ACM.
- [19] Anne Roudaut, Stéphane Huot, and Eric Lecolinet. TapTap and Mag-Stick: improving one-handed target acquisition on small touch-screens. In *Proceedings of the working conference on Advanced visual interfaces*, pages 146–153, Napoli, Italy, 2008. ACM.
- [20] Ted Selker. Touching the future. *Commun. ACM*, 51(12):14–16, 2008.
- [21] Paul S. Strauss and Rikk Carey. An object-oriented 3D graphics toolkit. *SIGGRAPH Comput. Graph.*, 26(2):341–349, 1992.
- [22] Robert J. Teather and Wolfgang Stuerzlinger. Guidelines for 3D positioning techniques. In *Proceedings of the 2007 conference on Future Play*, pages 61–68, Toronto, Canada, 2007. ACM.
- [23] TUIO. TUIO. <http://www.tuio.org/>, 2009.

- [24] Dan Venolia. Facile 3D direct manipulation. In *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, pages 31–36, Amsterdam, The Netherlands, 1993. ACM.
- [25] Daniel Vogel and Patrick Baudisch. Shift: a technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 657–666, San Jose, California, USA, 2007. ACM.
- [26] Andrew D. Wilson, Shahram Izadi, Otmar Hilliges, Armando Garcia-Mendoza, and David Kirk. Bringing physics to the surface. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, pages 67–76, Monterey, CA, USA, 2008. ACM.
- [27] Mike Wu and Ravin Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, pages 193–202, Vancouver, Canada, 2003. ACM.
- [28] Robert C. Zeleznik, Andrew S. Forsberg, and Paul S. Strauss. Two pointer input for 3D interaction. In *Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 115–ff., Providence, Rhode Island, United States, 1997. ACM.
- [29] Zhai. *Ph.D. Thesis: Human Performance in Six Degree of Freedom Input Control*. PhD thesis, University of Toronto, 1995.
- [30] Shumin Zhai and Paul Milgram. Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 320–327, Los Angeles, California, United States, 1998. ACM Press/Addison-Wesley Publishing Co.



Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399