Cursors for 3D Pointing

Robert J. Teather
Dept. of Computer Science & Engineering
York University
Toronto, ON, Canada
rteather@cse.yorku.ca

Wolfgang Stuerzlinger
Dept. of Computer Science & Engineering
York University
Toronto, ON, Canada
wolfgang@cse.yorku.ca

CHI'12, May 5–10, 2012, Austin, Texas, USA.

Abstract
We present a study of cursors for 3D pointing/selection interfaces. We compared a stereo- and mono-rendered (one-eyed) cursor used with two mouse-based and two remote pointing techniques. This comparison was performed in a 3D Fitts' law pointing experiment with varying target depths. Results indicate that the one-eyed cursor is beneficial only for some pointing techniques. While the mouse-based techniques performed best, our new ray-screen technique outperforms traditional ray pointing. This is likely because it is less affected by target depth.

Author Keywords
3D pointing; cursors; selection; Fitts' law.

ACM Classification Keywords
H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems – virtual reality. H.5.2 [Information Interfaces and Presentation]: User Interfaces – input devices, interaction styles.

Introduction
Recent graphics hardware can automatically convert non-stereo imagery for stereo 3D display. This is sometimes used in modern games, but raises the issue of cursor rendering/control: should drivers display a stereo cursor in the screen plane (i.e., the standard cursor depth), or choose the cursor depth based on the

geometry? With such stereo cursors, a major concern is the avoidance of stereo cue conflicts and diplopia (double vision) due to conflicting depth information. Current graphics drivers display a stereo-rendered 2D cursor using the disparity of the closest occluded surface. A similar idea uses a sliding 3D cursor by ray casting through the system cursor position, and then displaying a stereo cursor at the ray-scene intersection. Both techniques avoid diplopia as the cursor and geometry depths are guaranteed to be the same. Another approach is Ware's one-eyed cursor [14], which was found to be superior to stereo cursors. The one-eyed cursor displays a mono-rendered cursor to the dominant eye only. This effectively eliminates cursor stereo cues altogether, and thus avoids stereo cue conflicts.

Our work investigates the effects of cursor style on both mouse-based and remote pointing and extends work previously presented as a poster [12] with additional analysis and discussion. Both input devices are tested using both a screen-plane cursor and a geometry-sliding cursor. Based on previous work [11, 14], we expect that screen-plane cursors will perform better, since they are controlled by fewer degrees of freedom. However, this may be device dependent.

To re-investigate the results by Ware and Lowther [14], we compare stereo and mono rendered cursors. It is impossible to discriminate the depth of a one-eyed cursor in a static image. However, sliding cursor depth can perceive cursor depth through motion. Previous work [10] reported that users tend to move sliding cursors on object front surfaces to reach targets, even though the cursor depth is resolved automatically. We speculate that removing stereo cues may eliminate this behaviour and thus improve performance. Consequently, we include both a one-eyed *sliding* cursor, and a one-eyed *screen* cursor, neither of which suffer from diplopia. The one-eyed screen cursor serves as a benchmark against the 2D pointing literature.

We also investigate the effect of perspective in pointing. According to Fitts' law [2] and assuming targets are displayed *at the same depth*, pointing at perspectively distorted targets with screen-plane cursors should exhibit *constant* performance regardless of the presented target depth. This is because both the projected target size and distance scale by the same factor. Screen-plane cursors only interact with target projections; this thus has the same difficulty regardless of depth. Note that this does not generalize to targets *at different depths*, which project differently to the display surface.

Finally, we compare eye-origin and device-origin selection rays for 3D pointing. Studies looking at this on large displays [5] and stereo 3D displays [1] have so far yielded conflicting results.

To summarize, our contributions are:

- A comparison of one-eyed and stereo cursors. This extends Ware's work [14]. We provide evidence that one-eyed cursors improve performance only in some cases.
- A novel eye-centric screen cursor ray technique that outperforms traditional ray pointing.
- A comparison of mouse and remote pointing in stereo 3D displays.
- Evidence that consistent target depth does not affect performance with screen-plane cursors.

Related Work

Ray-based pointing techniques work with both 2DOF devices, like the mouse, as well as 3/6DOF devices, such as 3D trackers. Both 2D [5, 8, 13] and 3D [3, 6, 7, 9, 15] interface research investigates these techniques. Ray-based techniques cast a virtual ray from the eye through the tracking device/cursor into the scene. This ray is tested for intersections with the objects and the closest object is selected. A drawback of ray-based techniques is the relative difficulty in selecting remote objects [6]. Far away objects take up proportionally less screen space due to perspective. Assuming a dynamic viewpoint and an unchanging target arrangement, they may also appear proportionally closer together. Thus as Fitts' law [2] is (largely) scale independent, screen-plane pointing at projected object images should be unaffected by object depth. Moreover, 6DOF ray pointing has higher angular precision up close, and thus closer objects can be treated as effectively larger than far objects [6]. Conversely, remote objects are harder to select, as subtle device movements/rotations (and noise) are amplified down the ray.

Ware and Lowther [14] report that a "one-eyed" cursor outperforms a stereo 3D cursor in 3D selection tasks with a 3DOF tracker. Their one-eyed cursor ignored tracker depth and moved the mono-rendered cursor in the screen plane, requiring only a match in screen space, i.e., pointing at the object projection. The 3D cursor required accuracy in all three dimensions. Thus, there are large differences between these two techniques. Our study re-investigates one-eyed cursors to determine their benefits for remote pointing techniques.

Jota et al. [5] recently investigated eye and device-centric rays. They found that device-centric rays perform well in pointing tasks, while eye rays are better for tracing/steering tasks. Their study used only 2D screen-plane targets. In contrast, our study includes targets displayed at varying depths in a stereo display. A recent study by Argelaguet et al. [1] also investigated the difference between eye and device-centric rays. They developed a technique that used tracker *rotation* to control the orientation of an eye ray. This new "RCE" technique was significantly faster than traditional ray-casting, especially for targets that were (partially) occluded. Our new pointing technique lets the user point at the screen instead, and the selection (eye) ray is cast through that screen cursor. This is similar to mouse pointing, in that the object projection is ultimately what the user is pointing at.

Fitts' Law and Pointing

Fitts' law [2] is an empirical model of the well-known tradeoff between speed and accuracy in pointing tasks. The model is $MT = a+b \times \log_2(D/W+1)$. MT is movement time, D is target distance, and W is target size, while a and b are empirically derived. The log term is the index of difficulty (ID) and indicates overall pointing task difficulty. This implies that the smaller and farther a target, the more difficult it is to hit it accurately.

An extension supported by an international standard [4] is the use of "effective" measures. This post-experiment correction adjusts the error rate to 4% by re-sizing targets to their "effective" width (W_e). This enables the computation of throughput, a measure that incorporates both speed and accuracy by "normalizing" the accuracy as effective scores. Throughput is computed as $TP = \log_2(D_e/W_e+1)/MT$ where MT is the

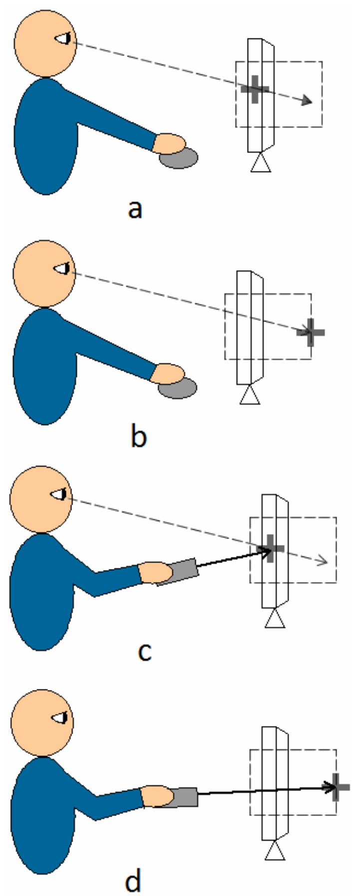

Figure 1.(a) Mouse cursor, (b) Sliding mouse cursor, (c) ray-screen, (d) ray pointing. The "+" is the cursor. The dashed box is the 3D scene volume. The dashed arrow is the selection ray, and the solid arrow is the device ray, these two coincide in (d).

average movement time, D_e is the effective distance (average of measured movement distances). The effective width, W_e, is computed by projecting the cursor onto the task axis (the line between subsequent targets) and multiplying the standard deviation of these distances by 4.1333. This adjusts the study error rate to 4%. Previous 3D pointing research [11] suggests that one should use the point closest to the target along the ray to compute a more accurate representation of the effective width W_e, as using the actual 3D cursor position would artificially inflate the effective measure. In essence, this suggestion projects the 3D task into 2D before computing throughput.

The main advantage of effective measures is to reduce throughput variability for the same device or condition. Consequently, results of pointing studies are more consistent and comparable. We previously reported [11] consistent mouse throughput across three studies using quite different conditions. Conversely, other measures such as movement time are less consistent due to speed-accuracy tradeoffs.

Pointing Techniques

While our study focuses on cursor properties, we still have to consider the effect of input devices since the two are not completely independent. To account for this, we used two different cursor modes with each device. The first mode employs a screen plane cursor. The second mode is based on a sliding cursor [11], as described earlier. Thus, our study investigates all four cursor/device combinations, as depicted in Figure 1.

These combinations include a screen-plane (standard) mouse cursor, a sliding 3D mouse cursor, our novel screen-plane ray-controlled cursor, and the classical ray pointing paradigm, i.e. device-centric ray pointing with a sliding cursor. The first mouse technique (Figure 1a) displays a mouse-controlled cursor in the screen plane and uses the eye-cursor ray for selection. This represents typical 3D selection using a mouse. The sliding mouse cursor instead displays the cursor where the (same) selection ray intersects the *scene*. The resulting cursor thus slides across the scene geometry.

Our novel remote-based "ray-screen" technique (Figure 1c) displays a screen cursor where the device ray intersects the *screen*, but does not use this ray for selection. Instead, the eye ray through said screen cursor is used for selection. This effectively affords selection of object projections. The final technique (Figure 1d) is "traditional" ray pointing: a device-centric ray that requires users to point the device directly at the 3D target volume.

Methodology

Participants

We recruited sixteen paid participants (mean age 23.1 years, *SD* 6.1), all undergraduate students at our university. Eight were female. All use the mouse with their right hand. All participants had normal stereo viewing capability, and six had previously used 3D input devices in pointing studies.

Apparatus

We used a 3 GHz PC, an Nvidia Quadro 4400, and a 24" stereo LCD. The participant sat about 70 cm from the display and head-tracking was disabled. Stereo was synchronized at 120 Hz via an RF hub with NVidia 3DVision Pro LCD shutter glasses. A NaturalPoint

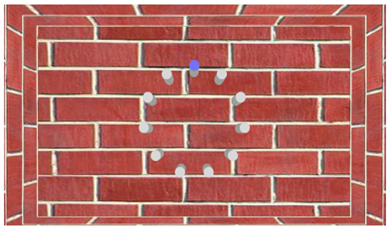

Figure 3. The scene depicting a target circle at -20 cm depth (displayed in stereo to participants).

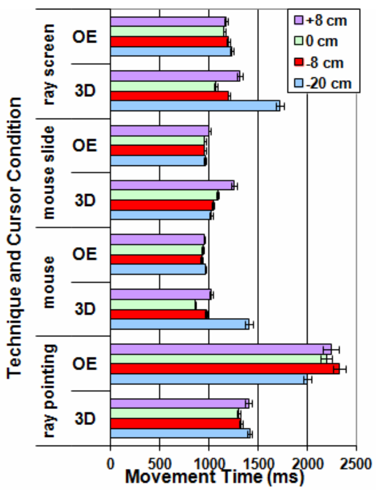

Figure 3. Movement time for each condition. One-eyed cursor conditions are represented with "OE", while stereo cursor conditions are represented with "3D". Error bars show ±1 S.E.

Optitrack with five S250e cameras tracked the remote pointing device. The tracker was calibrated to approximately 0.7 mm RMS. End-to-end system latency was about 65 ms. No smoothing was used, as noise was already very low, and the latency cost of filtering may outweigh the benefits [10]. Mouse acceleration was disabled, and the gain was set very low.

The 3D scene was a 30 cm deep box matching the display size, see Figure 2. Target cylinders were arranged in a circle. Blue target spheres were displayed on top of the cylinders. These spheres highlighted red when selected for better feedback. The cursor was displayed as a small 3D crosshair, either at the screen plane or in the 3D scene, depending on the current technique. In the one-eyed cursor mode, the cursor was displayed only to the viewer's dominant eye. In the ray mode, the device ray was also displayed to improve feedback. Stereo display was active in all conditions. Target size, distance, and depth were constant within target circles, but varied between circles. Target depth was measured from the screen surface, i.e., negative depth indicates a target behind the screen.

Procedure

Participants were first instructed on the task. As our participants had extensive mouse experience but only limited experience with remote pointing, we asked them to perform 10–20 practice trials with the ray pointing technique. This training partially compensates for the lack of familiarity. In all conditions, participants were instructed to select the blue highlighted target as quickly and accurately as possible.

Design

The study used a 2×4×4 within-subjects design. The independent variables were cursor style (one-eyed, or stereo), technique (mouse cursor, sliding cursor, ray screen, ray pointing), and target depth (+8, 0, -8, -20 cm). The dependent variables were movement time (ms), error rate (percentage of targets missed), and throughput (bits per second). There were 10 recorded trials per target circle. Each target circle represented a different index of difficulty, combinations of 3 distances and 2 sizes. Target distances were 7, 15, and 19 cm apart, while target sizes were 0.9 or 1.5 cm in diameter. This yielded six distinct *IDs* ranging from 2.5 to 4.5 bits, representing a typical range of pointing task difficulties. Thus each participant completed a total of 1920 trials, for a total of 30720 recorded trials overall.

Results

Results were analyzed using repeated measures ANOVA and Tukey-Kramer multiple comparisons at the 5% significance level (with Bonferonni correction).

Movement time

Cursor style had a significant main effect on movement time ($F_{1,15}$ = 16.9, p < .001). On average, the one-eyed cursor (μ = 1321 ms, σ = 554 ms) increased movement time compared to the stereo cursor (μ = 1211 ms, σ = 839 ms). However, there are strong interaction effects with technique, as the ray pointing technique was far worse with the one-eyed cursor. There was a significant two-way interaction effect between cursor style and technique ($F_{3,45}$ = 46.7, p < .0001). Ray pointing with the one-eyed cursor was significantly worse than all other conditions. The other conditions all benefitted from the one-eyed cursor. A

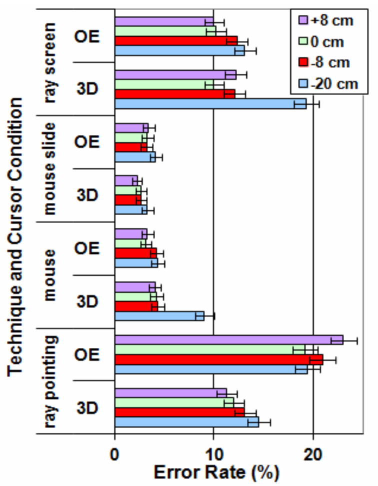

Figure 5. Error rates. Error bars show ±1 S.E.

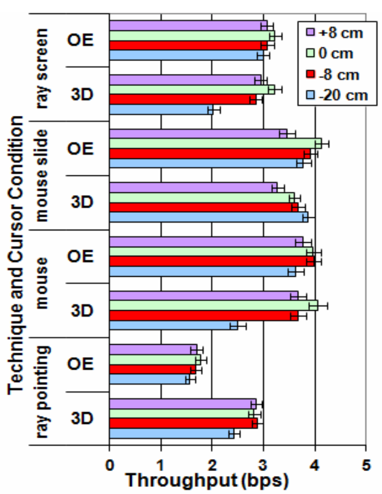

Figure 5. Throughput for each condition. Error bars show ±1 S.E.

significant three-way interaction effect between technique, cursor style, and target depth was also found ($F_{9,135}$ = 4.3, p < .0001). This revealed that the screen-plane conditions (mouse and ray-screen) with the 3D cursor performed significantly worse at -20 cm depth. There was also a significant main effect for technique on movement time ($F_{3,15}$ = 62.7, p < .0001), see Figure 3. A Tukey-Kramer post-hoc test (p < .05) revealed that both mouse techniques were significantly faster than the ray-based techniques. The ray screen technique was significantly faster than standard ray pointing

Error Rate

Cursor style alone did not have a significant effect on error rate ($F_{1,15}$ = 3.4, p > .05). Both mouse techniques had significantly lower error rates than both ray-based techniques ($F_{3,15}$ = 13.5, p < .0001). A significant interaction between technique and cursor style ($F_{3,45}$ = 8.7, p < .001) revealed that the one-eyed cursor increased error rates with the ray technique. Overall, the mouse error rate was around 4%, consistent with the 2D pointing literature. See Figure 4 for error rates.

Throughput

Throughput was computed as described earlier. The point on the selection ray closest to the target was used as the "cursor position" for the throughput computation. Cursor style alone did not have a significant effect on throughput ($F_{1,15}$ = 0.26, ns). However, there was a significant main effect for technique on throughput ($F_{3,15}$ = 103.1, p < .0001) and a significant interaction effect between technique, cursor style, and target depth ($F_{9,135}$ = 4.9, p < .0001).

Throughput for the mouse conditions was close to 4 bits per second, consistent with the 2D pointing literature. However, this fell dramatically for targets at -20 cm depth using the 3D cursor style. This fall-off is also present for the ray screen condition. The ray screen technique afforded significantly higher throughput than standard ray pointing (approximately 3 bps vs. 2.5 bps). The one-eyed cursor hindered the ray pointing technique, which was the worst condition overall, regardless of target depth.

Modeling

Fitts' law can also be used as a predictive model, by regressing movement time on index of difficulty. We performed this analysis for each technique under both the stereo cursor and one-eyed cursor styles. These results are presented in Figure 6 and Figure 7. The predictive quality of the model (as expressed by the R^2 values) is very high. However, it is worth noting that the one-eyed cursor consistently improved the R^2 values. The mouse cursor condition increased from 0.86 to 0.97, indicating that the model almost perfectly predicts one-eyed cursor mouse control. The stereo cursor style reduced the correlation for the mouse cursor, likely due to stronger stereo conflicts on deeper targets. The sliding cursor is affected less, likely because the cursor and target depths are effectively identical. Overall, this demonstrates that the predictive capabilities of Fitts' law are unaffected by target depth for techniques that use 2DOF input and a 2D cursor visualization.

Discussion

Consistent with previous results [14], the one-eyed cursor improved performance, albeit only for certain conditions. More precisely, the mouse, mouse-slide,

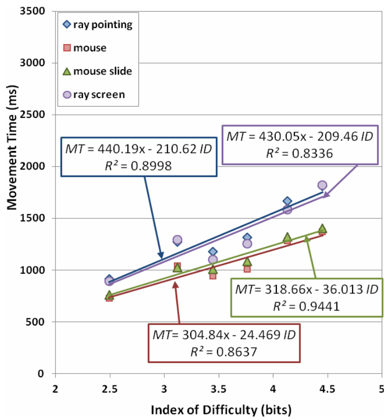

Figure 7. Models for stereo cursor style.

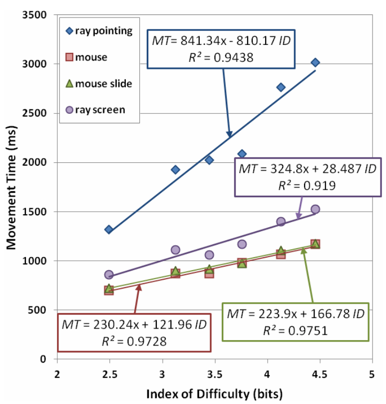

Figure 7. Models for one-eyed cursor style.

and ray-screen conditions all benefitted from the one-eyed cursor, while ray-pointing performed much worse with a one-eyed cursor. Our results also quantify the benefits of the one-eyed cursor with a more robust and widely accepted experimental paradigm compared to the original study [14]. The one-eyed cursor slightly improved performance with the mouse-based techniques, but its greatest benefit was to reduce the negative effect of deep targets with these conditions. This effect of depth is most noticeable in the screen-plane stereo cursor conditions (mouse and ray screen). In particular, throughput peaked at the 0 cm depth (i.e., at the screen surface) and fell for targets at different depths. The +8 cm and -8 cm depths show similar throughput, but the -20 cm condition shows a dramatic degradation of performance. This is clearly the effect of diplopia. The one-eyed cursor does not suffer from this problem, see Figure 5.

Movement time for the mouse slide technique using the stereo cursor was significantly faster for deeper targets compared to closer ones. This seems to be related to participants sliding the cursor up the sides of the target cylinder instead of relying on it "popping" to the front. This suboptimal behavior has been observed in previous work [10]. The one-eyed cursor *eliminated* this problem, and participants reported that they could not tell the difference between that condition and the one-eyed mouse (screen) condition. Indeed, the movement times for these conditions are nearly identical regardless of target depth.

Our results reveal also the differences between pointing techniques. The mouse techniques performed best, but the new ray screen technique was still competitive and outperformed standard ray pointing. We thus recommend this style of image plane technique over classical ray-pointing for VR systems and games alike. Note that this is similar to Argelaguet's results [1], but does not agree with Jota's work [5]. Our study used a stereo desktop VR system, while Jota used a large non-stereo display system. This difference may account for the discrepancy in results, and our results may thus not generalize to large displays. The multiple interaction effects indicate that some techniques work best with a one-eyed cursor, while others require a stereo 3D cursor. Similarly, some techniques perform best for deeper targets, while others perform best for close targets.

Finally, the one-eyed mouse cursor afforded throughput similar to a standard 2D mouse cursor. This was fairly consistent for both one-eyed mouse conditions. The one-eyed ray-screen condition was also unaffected by target depth. The movement times confirm that performance is unaffected by the perspective scaling of a scene with targets displayed *at the same depth* when using screen-plane techniques. This confirms our earlier prediction that according to Fitts' law, perspective scaling both the distance and target size by the same amount will not change the index of difficulty, and performance thus remains constant. This is not the case in scenes with varying target depths, which project to different sizes and distances. We plan to investigate this in future work.

Conclusion

We conducted a study to compare a one-eyed and a stereo cursor in the presence of targets displayed stereoscopically at varying depths. This also involved a comparison between four pointing techniques. We found that the one-eyed cursor was not universally beneficial. However, the mouse, mouse-slide, and ray-screen conditions all benefitted from the one-eyed cursor. In some cases, notably for deep targets, using a one-eyed cursor substantially improved performance. Our results quantify the benefits of the one-eyed cursor using a more robust and widely accepted experimental paradigm compared to previous work. Overall, mouse-based techniques tended to perform best, but our new "ray screen" selection technique also outperformed traditional ray pointing. Consequently, we recommend this new technique for 3D systems that use remote pointing devices.

References

1. Argelaguet, F. and Andujar, C., Efficient 3D Pointing Selection in Cluttered Virtual Environments, *Computer Graphics and Applications, 29*, 2009, 34-43.

2. Fitts, P. M., The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology*, *47*, 1954, 381-391.

3. Grossman, T. and Balakrishnan, R., The design and evaluation of selection techniques for 3D volumetric displays, in *UIST '06*, 3-12.

4. ISO 9241-9 Ergonomic requirements for office work with visual display terminals (VDTs) - Part 9: Requirements for non-keyboard input devices. International Organization for Standardization, 2000.

5. Jota, R., Nacenta, M. A., Jorge, J. A., Carpendale, S., and Greenberg, S., A comparison of ray pointing techniques for very large displays, in *Graphics Interface 2010,* 269-276.

6. Kopper, R., Bowman, D. A., Silva, M. G., and McMahan, R. P., A human motor behavior model for distal pointing tasks, *International Journal of Human-Computer Studies*, *68*, 2010, 603-615.

7. Kunert, A., Kulik, A., Lux, C., and Fröhlich, B., Facilitating system control in ray-based interaction tasks, in *VRST '09*, 183-186.

8. Oh, J.-Y. and Stuerzlinger, W., Laser pointers as collaborative pointing devices, in *Graphics Interface '02*, 315-320.

9. Steinicke, F., Ropinski, T., and Hinrichs, K., Object selection in virtual environments using an improved virtual pointer metaphor, in *Computer Vision and Graphics*, vol. 32, 2006, 320-326.

10. Teather, R. J., Pavlovych, A., Stuerzlinger, W., and MacKenzie, I. S., Effects of tracking technology, latency, and spatial jitter on object movement, in *3DUI '09,* 43-50.

11. Teather, R. J. and Stuerzlinger, W., Pointing at 3D targets in a stereo head-tracked virtual environment, in *3DUI '11*, 87-94.

12. Teather, R. J. and Stuerzlinger, W., Investigating one-eyed and stereo cursors for 3D pointing tasks, poster at *3DUI '12,* 167-168.

13. Wang, G., McGuffin, M. J., Berard, F., and Cooperstock, J. R., Pop-up depth views for improving 3D target acquisition, in *Graphics Interface '11,* 41-48.

14. Ware, C. and Lowther, K., Selection using a one-eyed cursor in a fish tank VR environment, *TOCHI*, *4*, 1997, 309-322.

15. Wingrave, C. A., Tintner, R., Walker, B. N., Bowman, D. A., and Hodges, L. F., Exploring individual differences in raybased selection; strategies and traits, in *VR '05,* 163-170.