Efficient 3D Pointing Selection in Cluttered Virtual Environments

Ferran Argelaguet and Carlos Andujar ■ *Universitat Politècnica de Catalunya*

Most pointing techniques for 3D selection in spatially immersive virtual environments rely on a ray originating at the user's hand. However, a mismatch might exist between visible and selectable objects. The authors propose a new device-ray mapping, Ray Casting from the Eye, to overcome this problem.

The virtual pointer is arguably one of the most popular metaphors for object selection in virtual environments (VEs). Numerous studies have demonstrated that virtual-pointing techniques often result in more effective selection than competing 3D interaction metaphors such as the virtual hand. Virtual pointing allows the selection of objects beyond the user's reach and requires less physical hand movement. Here, we focus on how the pointing direction is defined—that is, how the user's movements are mapped into the pointing direction.

Pointing techniques roughly fall into two families:

- The selection ray originates at the user's hand (including ray casting and variants[1]).
- The selection ray originates at the user's eye (also called *image-plane techniques*[2]).

We call these two families *hand-rooted* and *eye-rooted* techniques. Hand-rooted techniques often provide visual feedback by drawing a ray or cone extending from the user's hand. Eye-rooted techniques must rely on a different feedback strategy because no matter how you control the ray direction, the ray's projection onto the viewing plane is a single point.

With hand-rooted techniques, the set of objects visible from the user's eye might differ from those that appear unoccluded from the hand position (which often determines which objects are selectable). Research has largely ignored this issue, concentrating instead on pointing facilitation techniques suited for fast, accurate selection of small, distant targets. Although some researchers have studied the selection of partially occluded objects,[3,4] they have focused solely on either the eye position or the hand position. Consequently, no research has focused on the interaction of both and its effects on selection performance. (For more on previous work in virtual pointing, see the sidebar.)

In this article, we study the impact of such eye-hand visibility mismatch on selection tasks performed with hand-rooted pointing techniques. We propose a new mapping for ray control, called Ray Casting from the Eye (RCE), which attempts to overcome this mismatch's negative effects. In essence, RCE combines the benefits of image-plane techniques (the absence of visibility mismatch and continuity of the ray movement in screen space) with the benefits of ray control through hand rotation (requiring less physical hand movement). This article builds on a previous study on the impact of eye-to-hand separation on 3D pointing selection.[5] Here, we provide empirical evidence that RCE clearly outperforms classic ray casting (RC) selection, both in sparse and cluttered scenes.

Eye-Hand Visibility Mismatch

Throughout this article, we focus on basic RC, although the discussion also applies somewhat to all RC hand-rooted variants. (See the sidebar for more discussion on RC and its variants.) In this context, eye-hand visibility mismatch involves two main issues. The first relates to the solid angle subtended by potential targets with respect to the eye position E and the hand position H. It affects all kinds of scenes, including sparsely occluded ones. The second issue considers interobject occlusion (some objects can appear occluded from the hand but not from the eye, and vice versa), which mainly affects densely occluded scenes.

Published by the IEEE Computer Society 0272-1716/09/$26.00 © 2009 IEEE

Solid-Angle Mismatch

A distinct feature of RC with respect to image-plane techniques is that the user selects objects by pointing directly at them rather than their screen projections. A first consequence is that objects with the same screen projection might require very different ray orientations (see Figure 1). Indeed, as we shall see, the accuracy required to select objects with RC isn't given directly by their screen projections.

Let $\Omega_E(S)$ and $\Omega_H(S)$ be the solid angle subtended by an object S with regard to the user's eye and the user's hand, respectively. $\Omega_E(S)$ is proportional to the screen projection of S, whereas, in absence of occluding objects, $\Omega_H(S)$ is a good measure of the object's effective width (W) and thus a measure of how much accuracy is required to select S. Figure 1 shows two objects with the same screen projection but quite different solid angles. Again, this is a distinct feature of RC with regard to image-plane techniques. We analyzed previously how $\Omega_H(S)$ varies depending on the distance $d = \|S - E\|$ and the eye-hand separation $h = \|E - H\|$.[5] For h = 80 cm, which is a typical eye-hand distance when the user is standing, an object 70 cm from the user will subtend with regard to the hand only 50 percent of its solid angle with regard to the eye.

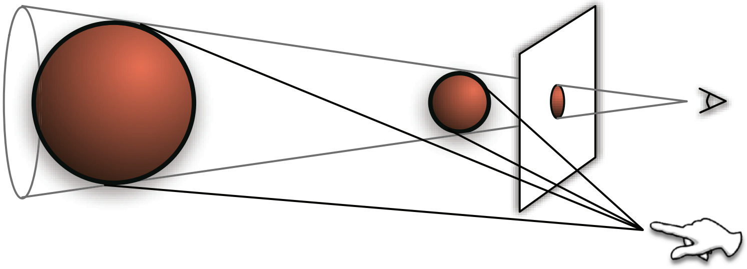

Figure 1. Objects with the same screen projection subtend different solid angles with respect to the hand position, which might require different levels of accuracy for its selection.

Occlusion Mismatch at Object Level

Let V_E and V_H be the sets of objects visible from the eye and hand positions, respectively. Here, we consider partial visibility; that is, we consider that an object S is visible from a point P if at least one point in ∂S is visible from P. We focus on the subset $V_E \cup V_H$, which we can decompose into three disjoint sets: $V_E \cap V_H$, $V_H - V_E$, and $V_E - V_H$. Objects in $V_E \cap V_H$ are visible and selectable; for now, we'll suppose they pose no problem.

The case of $V_H - V_E$ corresponds to objects that are hidden to the user's eyes but are reachable from a ray emanating from the user's hand. For example, in Figure 2a, object B is occluded from the eye position but not from the hand position. So, it's currently selected even though it isn't visible. Object A might appear to be selected, because the ray apparently intersects with A (because the screen projection P' of the ray's last visible point is on the silhouette of the screen projection of A), as shown in Figure 2a. However, in the absence of additional feedback, if the user triggers the selection confirmation at this point, the hidden object B will be erroneously selected.

On the other hand, objects in $V_E - V_H$ are visible but aren't reachable from a ray emanating from the hand. For example, in Figure 2b, object A is visible from the eye position but completely obscured from the hand position. If the user starts moving the ray upward, increasing its elevation and thus trying to bring P' to the screen projection of A, P' will jump from object B to object C. The discontinuous path of P' on the screen appears at the bottom of Figure 2b. In this situation, changing only the hand direction isn't enough; the only way to select A is to move the hand to a location from which A appears unoccluded. Again, this situation is a distinct feature of hand-rooted pointing and doesn't apply to image-plane selection.

Occlusion Mismatch at Point Level

As we mentioned before, so far we've supposed

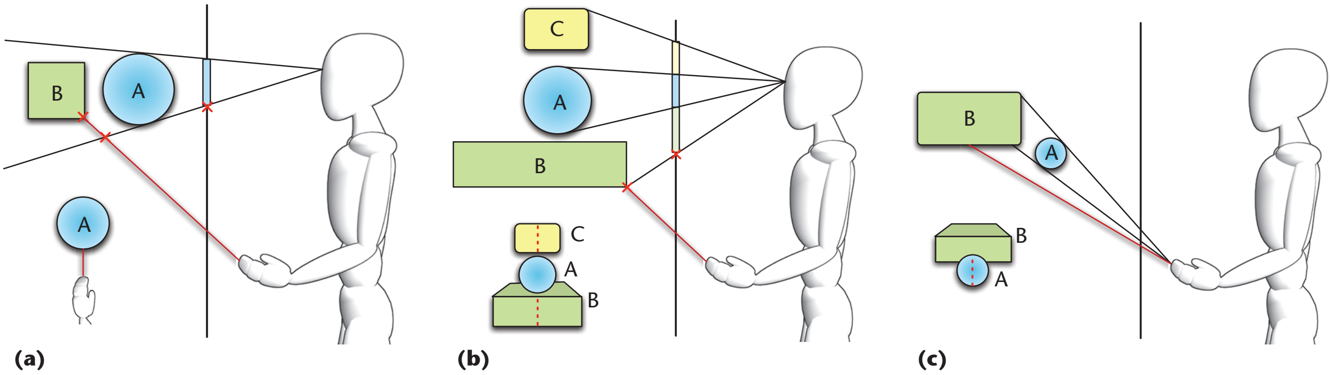

Figure 2. (a) A situation where the user can select an object hidden by another object. (b) A visible object can't be selected because it can't be reached by a ray emanating from the user's hand. (c) Object A is visible from both the eye position E and the hand position H, but no point on its boundary is simultaneously visible from E and H.

Previous Work in Virtual Pointing

A review of all the strategies for facilitating 3D selection in virtual environments (VEs) is beyond this article's scope (for a comprehensive survey, see *3D User Interfaces: Theory and Practice*[1]). Here we review only virtual-pointing techniques, focusing on how they define the pointing direction (Table 1 summarizes the techniques). In our notation, the eye position is *E*, the hand position is *H*, and the hand direction is $\vec{h}$.

Hand-Rooted Techniques

Ray casting[2] is the simplest virtual-pointing technique used in VEs. The default implementation employs a six-degree-of-freedom sensor attached to the user's hand to isomorphically give the pointing direction, and draws a ray extending from the hand to provide visual feedback. The ray can intersect several objects, but only the first intersected object can be selected (two-handed techniques[5,9] often circumvent this rule). Usually, the user presses a button to signal that he or she intends to select the intersected object.

Jiandong Liang and Mark Green proposed an extension of ray casting that replaces the selection ray with a conic volume with a constant apex angle, facilitating the selection of small, isolated targets.[3] Again, the hand orientation isomorphically gives the cone axis direction.

Researchers have also proposed a two-handed technique to define the pointing direction.[5,9] One hand specifies the ray origin; the other specifies where the ray is pointing.

All these techniques use *H* as the ray's origin and thus could suffer from the eye-hand visibility mismatch discussed in the main article.

Eye-Rooted Techniques

In image-plane selection, the user points directly at the object's screen projection rather than the object itself.[7] The simplest approach, occlusion selection, uses a selection ray defined by joining the user's eye position with the user's hand position. The aperture technique replaces the selection ray with a conic volume whose apex angle is controlled by the user.[8] The user can interactively control

Table 1. A summary of virtual-pointing techniques. *k* indicates a constant value, H_n is the nondominant hand's position, and $\vec{n}$ is a vector perpendicular to the screen surface for controlling the selection volume. The "origin" column refers to the ray's origin or the cone apex.

Technique	Origin	Direction	Volume	Aperture tan(θ)
Ray casting[2]	H	$\vec{h}$	Ray	n/a
Flashlight[3]	H	$\vec{h}$	Cone	k
Shadow cone selection[4]	H	$\vec{h}$	Cone	k
Two-handed pointing[5]	H_n	$H - H_n$	Ray	n/a
Direct image-plane pointing[6]	H	$\vec{n}$	Ray	n/a
Occlusion selection[7]	E	$H - E$	Ray	n/a
Aperture[8]	E	$H - E$	Cone	$k/\|H - E\|$
Ray Casting from the Eye	E	$\vec{h}$	Ray	n/a

that objects in $V_E \cap V_H$ pose no problem, regardless of the portion of the object that's visible from *H* and *E*. An approach that seems more accurate for predicting potential selection problems is to consider an object *S* as a potentially difficult target whenever any of $\Omega_E(S)$ or $\Omega_H(S)$ is below a certain value. However, this approach is still inaccurate because an object *S* with large $\Omega_E(S)$ and $\Omega_H(S)$ can still be difficult to select. Figure 2c depicts this situation. Both $\Omega_E(A)$ and $\Omega_H(A)$ are large, but *A* is still difficult to select because no point on its boundary is simultaneously visible from *E* and *H*. Consequently, the user can intersect *A* with the ray, but the intersection point will be hidden by *B*, keeping the user from having visual feedback.

So, we can define a more appropriate measure of the accuracy required to select *S* in terms of its simultaneous visibility. Let S' be the set of points of *S* that are simultaneously visible from *E* and *H*. We claim that $\Omega_H(S')$ is an accurate measure of the effective *W* of the object in terms of selection difficulty.

The analysis of the behavior when *h* increases requires distinguishing two causes for a point in ∂S not being simultaneously visible:

- self-occlusion by faces of the same object and
- interocclusion by other objects.

If we increase *h*, self-occlusion obviously increases, because more front faces from *E* become back faces from *H*, and vice versa. However, predicting the interocclusion's behavior is more difficult because it strongly depends on the object layout and the alignment of *E* with the scene objects. Our experiments with different complex 3D scenes show that $\Omega_H(S')$ tends to decrease as *h* increases, although not uniformly.[5]

Figure 3 shows the empirical evaluation of visibility mismatch for several test scenes. The color temperature represents the eye-to-hand distance *h* at which each point appears occluded from *H*. Visibility mismatch affects a large part of the scene: approximately 25 percent of the visible pixels the selection volume's spread angle simply by bringing his or her hand closer to or farther from the eye.

The eye-rooted techniques proposed so far use the hand position to control the ray direction.[10] So, they require more physical effort from the user than techniques where the hand direction controls the ray direction. This limitation is more apparent in immersive displays such as CAVEs (Cave Automatic Virtual Environments), where the user is standing and selection of front objects requires roughly aligning the hand with the eye position.

Pointing-Facilitation Techniques

The application of Fitts' law to human-computer interaction has led to successful techniques for improving pointing performance.[11] Fitts' law asserts that the time T to acquire a target of effective width W at distance D is governed by the relationship $T = a + b\log_2(D/W + 1)$, where a and b are empirically determined constants and the logarithmic term is called the *index of difficulty*.

References

1. D.A. Bowman et al., *3D User Interfaces: Theory and Practice*, Addison-Wesley, 2004.
2. M. Mine, *Virtual Environment Interaction Techniques*, tech. report TR95-018, Dept. of Computer Science, Univ. North Carolina at Chapel Hill, 1995.
3. J. Liang and M. Green, "JDCAD: A Highly Interactive 3D Modeling System," *Computers & Graphics*, vol. 18, no. 4, 1994, pp. 499–506.
4. A. Steed and C. Parker, "3D Selection Strategies for Head-Tracked and Non-Head Tracked Operation of Spatially Immersive Displays," *Proc. 8th Int'l Immersive Projection Technology Workshop*, 2004, pp. 163–170.
5. M. Mine, J. Frederick Brooks, and C. Sequin, "Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction," *Proc. Siggraph*, ACM Press, 1997, pp. 19–26.
6. S. Lee et al., "Evaluation of Pointing Techniques for Ray Casting Selection in Virtual Environments," *Proc. 3rd Int'l SPIE Conf. Virtual Reality and Its Application in Industry*, Soc. Photo-Optical Instrumentation Engineers, 2003, pp. 38–44.
7. J. Pierce et al., "Image Plane Interaction Techniques in 3D Immersive Environments," *Proc. 1997 Symp. Interactive 3D Graphics*, ACM Press, 1997, pp. 39–43.
8. A. Forsberg, K. Herndon, and R. Zeleznik, "Aperture Based Selection for Immersive Virtual Environments," *Proc. 9th Ann. ACM Symp. User Interface Software and Technology* (UIST 96), ACM Press, 1996, pp. 95–96.
9. A. Olwal and S. Feiner, "The Flexible Pointer: An Interaction Technique for Augmented and Virtual Reality," *Proc. 2003 ACM Symp. User Interface Software and Technology* (UIST 03), ACM Press, 2003, pp. 81–82.
10. C.A. Wingrave et al., "Exploring Individual Differences in Ray-Based Selection: Strategies and Traits," *Proc. 2005 IEEE Conf. Virtual Reality* (VR 05), IEEE CS Press, 2005, pp. 163–170.
11. R. Balakrishnan, "'Beating' Fitts' Law: Virtual Enhancements for Pointing Facilitation," *Int'l J. Human-Computer Studies*, vol. 61, no. 6, 2004, pp. 857–874.

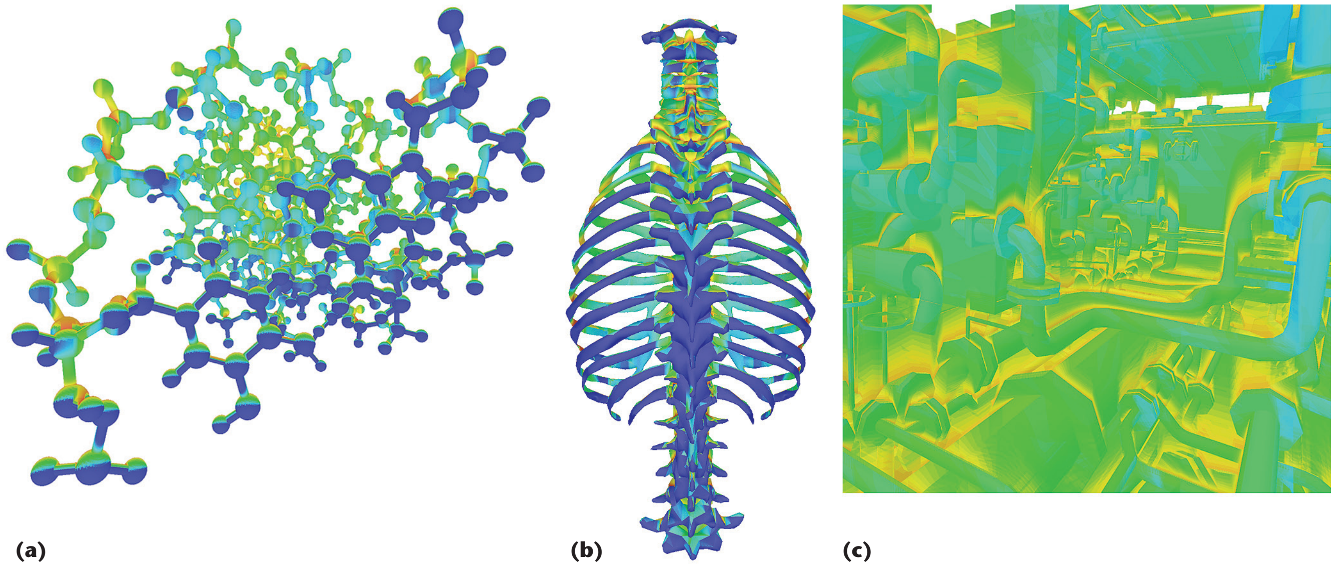

Figure 3. Simultaneous visibility on three test models: (a) chemical, (b), anatomical, and (c) industrial. The color temperature represents the eye-to-hand distance *h* at which each point appears occluded from *H*. Warmer colors represent areas of the model where the eye-hand visibility mismatch is higher.

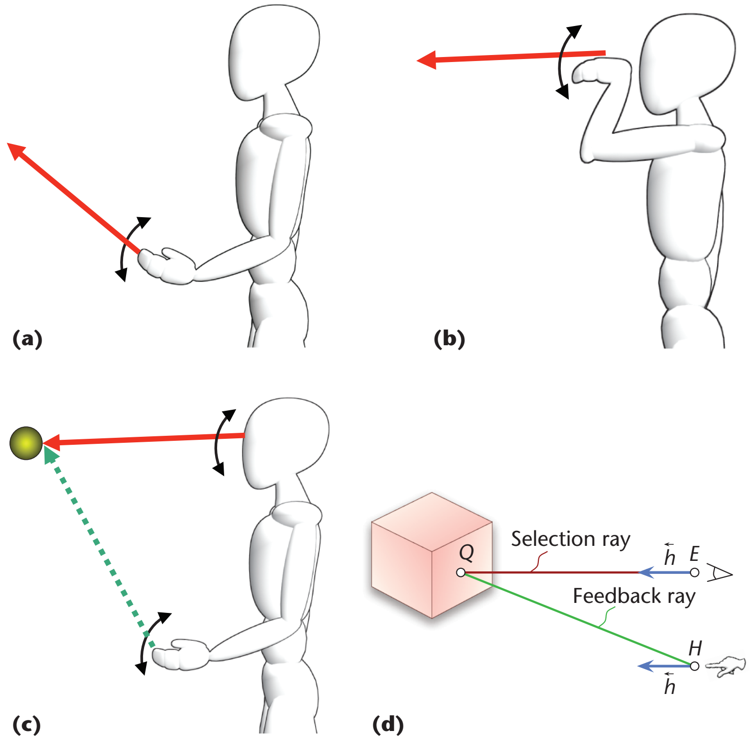

Figure 4. Dealing with the eye-hand visibility mismatch problem. (a) In classic ray casting (RC), the selection ray is cast from the user's hand, thus potentially causing eye-hand visibility mismatch. (b) This problem persists unless the user aligns his or her hand with the pointing direction, which results in an uncomfortable position. (c) Ray Casting from the Eye (RCE) uses a selection ray cast from the eye, whose direction is controlled by the hand orientation. Because the selection ray is insensitive to hand position, users can select objects as in (b) but with less physical effort. (d) Because the selection ray isn't visible to the user, RCE provides visual feedback by drawing a ray cast from the hand to the first intersection of the selection ray with the scene.

correspond to parts difficult to select, for typical eye-hand distances.

RCE

Figure 4 summarizes the visibility mismatch problem and how RCE handles it. Conceptually, RCE uses two rays: a *selection ray* and a *feedback ray*. We define the selection ray using the parametric equation $E+\lambda\vec{h}$, where $\vec{h}$ is the hand direction. To the best of our knowledge, no other research has explicitly proposed or evaluated this mapping between user movements and pointing direction. The first intersection of this ray with the scene objects, if any, is called Q and indicates the selected object. If the ray doesn't intersect any object, we preserve the last valid depth.

As we mentioned before, because the selection ray is cast from the eye, the ray projects onto a single point in the viewing plane. Consequently, we must choose another strategy for providing appropriate visual feedback. First, we considered drawing a 3D cursor somewhere along the ray. Until now, we've ignored the fact that, because VE displays are stereoscopic, there's no single eye position. Fortunately, the previous discussion also applies to stereoscopic displays by simply taking *E* as the middle point in the line joining both eyes. However, we can't ignore stereoscopic displays when deciding at what distance λ along the ray to place the 3D cursor.

In a monoscopic display, the value chosen for λ is irrelevant, but in a stereoscopic display this value plays a key role because the resulting cursor parallax is proportional to λ. A first option is to draw the cursor at Q, so that the user perceives the cursor as attached to the object being pointed at. Unfortunately, each time the ray intersects a new object, the cursor's parallax changes abruptly to match the new object's depth. So, the user's eyes are forced to converge at different distances at a high rate. Because two objects far apart in 3D space can have arbitrarily close screen projections, the cursor parallax can potentially vary too quickly, becoming a serious distraction. This problem is particularly acute in plano-stereoscopic displays, owing to the breakdown of the natural relationship between eye accommodation and convergence. Failure to properly converge at the right distance causes diplopia (double vision), which is inconvenient.

We also explored the opposite solution, using a constant value for λ. For example, we can compute λ as the user's distance from the screen surface, so that the cursor will appear always at zero parallax. Because the cursor and underlying scene object are displayed with different parallax values, simultaneous binocular fusion of them is impossible, and the user will perceive the cursor or the underlying object in diplopia. Near the fovea, the maximum binocular disparity resulting in binocular fusion corresponds to a visual angle of about 10 minutes of arc. This means that the previous problem also applies to small scenes with low depth variation. You can easily reproduce the resulting effect by trying to point at a real object with the thumb, leaving both eyes open.

Our solution for visual feedback is to draw a feedback ray defined by $H + \lambda(Q - H)$. More precisely, we only draw the segment *HQ*. Obviously, the parallax of the endpoint of Q also changes rapidly, but the replacement of a 3D cursor by a ray notably alleviates the problem. (In this respect, RCE behaves like any RC variant.)

The feedback ray has these properties:

- Because the feedback ray originates at the hand position and responds to hand orientation, it feels like a normal RC ray. In fact, both techniques tend to be the same because the user aligns his or her hand with the pointing direction, as in Figure 4b.

- Unlike classic RC, the ray's endpoint is insensitive to the hand position; it depends only on the hand orientation. When the user is standing in a spatially immersive display, this allows a more comfortable posture, requiring only hand rotation, not translation.
- The movement of the feedback ray's endpoint is continuous (Q behaves like a 3D cursor), and an object's screen projection is a good measure of its effective size.

In the next section, we look at an empirical evaluation verifying these properties.

User Evaluation

Our previous study focused on selection tasks using real-world models including a variety of objects.[5] Overall, we found that mismatch significantly affected selection performance, although we didn't study mismatch's impact in a per-object, controlled fashion. So, we didn't identify which intervening factor (solid-angle mismatch, occlusion mismatch, or index of difficulty, which we explain later) played a key role in performance loss.

We then designed a new experiment with two goals in mind. First, we wanted to evaluate mismatch's impact on selection tasks in a more controlled environment. In this new experiment, we now control the index of difficulty and the occlusion level on a per-object basis. Mismatch will likely increase the difficulty of selection tasks, but the resulting difference might be nonsignificant. For instance, users can develop strategies to compensate for the mismatch—for example, by moving their hand quickly to an unoccluded location.

Our second goal was to compare RCE's performance with that of traditional RC. Given that the eye-hand visibility mismatch doesn't affect RCE, you might expect it to outperform RC, at least in scenes where mismatch plays an important role. In practice, however, this might not be the case. For example, the ray's behavior might be counterintuitive because the ray's endpoint is controlled solely by hand rotation, being insensitive to hand translation. Furthermore, the feedback ray might intersect occluding objects other than the target object, which might cause distraction.

Our previous study suggested that RCE might reduce selection time in cluttered scenes.[5] However, we hadn't determined which objects would be easier to select with our technique or determined under which viewing circumstances objects would be easier to select.

In this new experiment, we explored the correlation between the selection time and index of difficulty, $ID = \log_2(D/W + 1)$. Because hand rotation controls the selection ray, we estimate the distance D to the target as the minimum rotation angle required for the selection ray to hit the target from the start position. We estimate the target size W as the apex angle of the minimum cone, with the apex in the user's hand containing the target.

Apparatus

We conducted the experiment in a four-sided CAVE (Cave Automatic Virtual Environment) with stereo projectors at 1,280 × 1,280 resolution. The input device was a 6-degree-of-freedom Wanda and an electromagnetic (EM) tracking system providing 60 updates per second with 4 ms latency. At the user position (90 cm from the EM emitter), position and orientation root-mean-square errors were below 0.5 mm and 0.06 degrees, respectively. To avoid the Heisenberg effect and reduce the number of selection errors, we provided users with a wireless mouse to trigger selection confirmation. A cluster of 2.66 GHz QuadCore PCs with Nvidia Quadro FX 5500 cards drove the experiment. Twelve volunteers, ages 22 to 35, participated; six had no experience with VE applications, three had some experience, and two were experienced users.

Procedure

We asked the participants to select a sequence of objects as quickly as possible. We clearly highlighted the next object for selection. To provide additional visual feedback, we highlighted in red the object intersected by the selection ray.

We used two different test models, each having 17 spheres of varying sizes (0.2 m to 60 m) and distances (0.6 m to 70 m), placed such that all spheres had approximately the same screen projection. (Because the participants were head-tracked, the actual screen projection varied according to each participant's head movements.)

In the first test model, we arranged the spheres to avoid interobject occlusion from the eye and hand positions, with nearby spheres appearing above more-distant ones from the user's perspective (see Figure 5a). We call this the *unoccluded model*.

In the second test model (see Figure 5b), we distributed the spheres to create varying levels of interobject occlusion from the user's hand position but not from the eye position (all spheres were clearly visible so as to keep discovery time from distorting the results). We call this the *occluded model*.

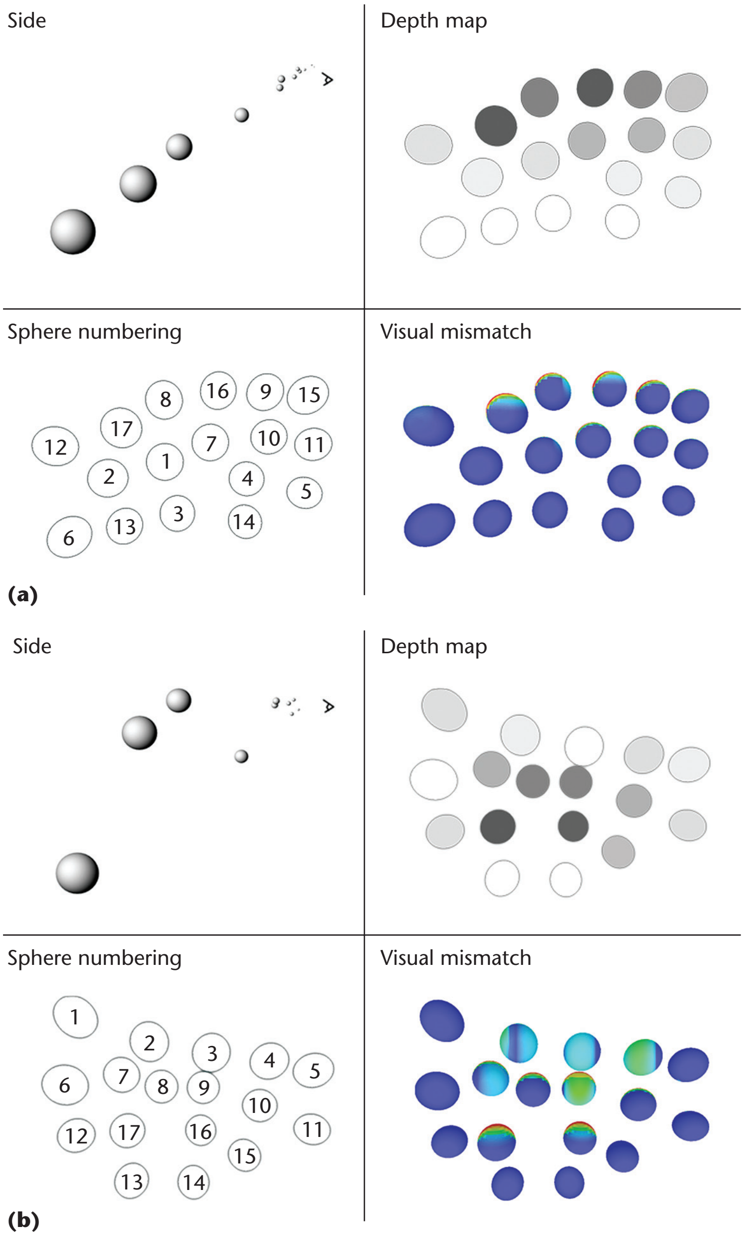

Figure 5. Test models used in the study: (a) unoccluded and (b) occluded. The color temperature encodes the potential level of visibility mismatch. Only the spheres numbered from 1 to 16 were potential targets; we introduced sphere 17 solely to increase mismatch. Whereas the mismatch in (a) was due to object self-occlusion, in (b) it was also due to object interocclusion.

To ensure that the difficulty of all trials remained constant among the participants, we precomputed a random sequence of objects that each participant had to select. We used the same sequence in all the trials, each trial consisting of 300 selections.

Design

We used a repeated-measures, within-subjects design. The independent variables were the test model (occluded or unoccluded) and the selection technique (RC or RCE).

Each participant performed the experiment in one session lasting approximately 25 minutes. We divided the experiment into two blocks, one for each technique, and each block had two trials, one for each model. Before each block, participants received a one-minute training session. We randomized the order of the blocks and trials for each participant to avoid learning effects.

The dependent measures were total selection time, error rate, and focus changes. We measured the error rate by counting the number of erroneous clicks. Focus change indicates the number of times the target object changed its selection status before confirming its selection.

Results

To analyze mismatch's impact on RC, we compared user performance with the occluded and unoccluded models (see the left part of Figure 6a). The average selection time was 319 seconds for the occluded model and 264 seconds for the unoccluded model. One-way ANOVA (analysis of variance) confirmed the test model's significant effect on selection time ($p < 0.005$). Because the screen projection of the spheres in both models was roughly the same, this result seems to confirm our hypothesis that, for RC selection, eye-hand visibility mismatch significantly affects selection time.

Regarding RCE, the selection difficulty depends mainly on the objects' screen projection, so we expected the eye-hand visibility mismatch to have little or no impact on performance. The right part of Figure 6a compares the selection times. The average times were slightly greater with the occluded model. One-way ANOVA of selection time versus the test model confirmed the difference's significance ($p < 0.02$).

During the experiments, we noticed that some novice users had difficulty selecting some objects because of visual-feedback conflicts. The worst-case scenario for RCE seems to be the selection of a very close object whose screen projection is surrounded by distant objects.

Figure 7 depicts this situation. Suppose the user wants to select the blue sphere. Although both spheres have similar screen projections, the blue sphere is much smaller and closer to the viewer (the picture shows the two spheres with reduced depth disparity for clarity). Starting from the situation on the left, an untrained user might think that the action required to select the blue sphere involves intersecting it with the feedback ray.

Because the blue sphere occludes the ray, the visual feedback indicates clearly that the feedback ray is behind it, so the user might rotate his or her hand upward (the top right of Figure 7). This movement will cause the feedback ray to intersect the object, but this won't change the green sphere's selection status. (Remember that the feedback ray is solely for visual feedback, with the selection ray defining the object being pointed at.)

So, the correct action to select the blue sphere is to rotate the hand downward (the bottom right of Figure 7). This problem appeared only in the occluded test model, because the first model's object layout avoided this situation. However, users could quickly overcome this behavior; the average selection times increased only 10 percent, in contrast with the 21 percent increment suffered by RC selection.

Therefore, we conclude that interobject occlusion affects both techniques, although the impact is much more noticeable with RC than with RCE. For RC, the loss of performance appears due to mismatch, whereas for RCE, it appears due to depth cue conflicts introduced by the visual feedback.

Regarding selection time, our approach clearly outperforms RC with both test models (see Figure 6a). With RCE, users required, on average, 33 percent less time to complete the trials with the unoccluded model and 15 percent less with the occluded one. Two-way ANOVA of selection time versus the test model and selection technique confirmed that the technique had a significant effect ($p < 0.001$). Indeed, a per-object study revealed that all objects were selected more quickly with RCE.

For the unoccluded test model (see Figure 8a), the selection time particularly improved for those objects with higher mismatch (objects 8, 9, and 16 in Figure 5a), owing to object self-occlusion. We also observed a slight improvement for the other objects. This is because the amount of rotation required to select an object, and hence its index of difficulty, depends on the selection technique (as we discussed in the section "Solid-Angle Mismatch"). Indeed, the objects' average index of difficulty was slightly higher for RC selection. (Accurate control of the index of difficulty is hard to achieve because the amount of hand rotation required to select a given object depends on the final hand position after the previous object in the sequence has been selected.)

Performance gains with the occluded model also applied to all objects (see Figure 8b). With these results, we can confirm than RCE clearly outperforms RC in presence of mismatch.

Figure 6b compares the number of focus changes. Two-way ANOVA showed that the technique ($p < 0.05$), the test model ($p < 0.01$), and the interaction of both ($p < 0.05$) significantly affected the number of focus changes. Again, RCE outperformed RC, requiring an average of seven percent fewer focus changes.

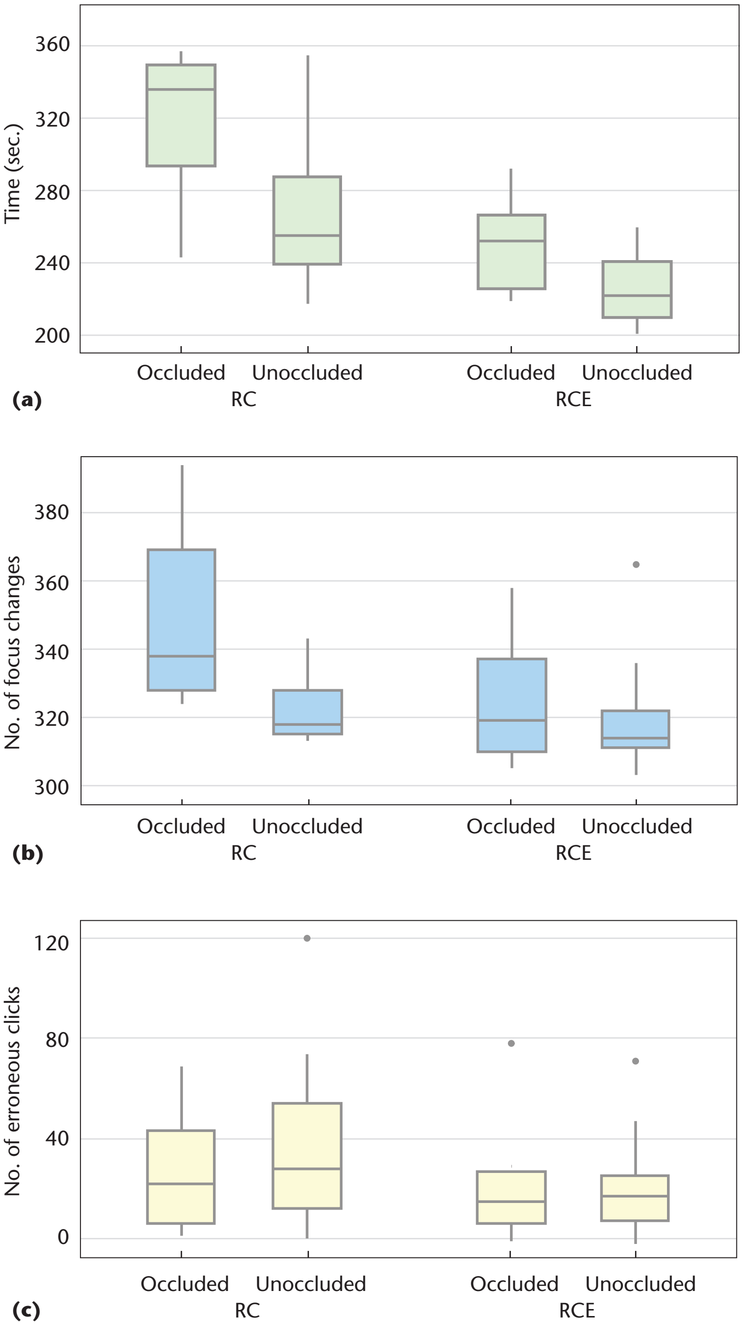

Figure 6. The performance of ray casting (RC) and Ray Casting with the Eye (RCE) on the occluded and unoccluded test models: (a) time, (b) focus changes (the number of times the target object changed its selection status before confirming its selection), and (c) erroneous clicks. The selection technique and the test model had a significant impact on time and focus changes, but not on erroneous clicks.

Figure 6c shows the results for the number of erroneous clicks. Two-way ANOVA showed no significant effect on this measure from the technique ($p = 0.129$), the test model ($p = 0.3$), or the interaction of both ($p =.0.5$). However, the standard

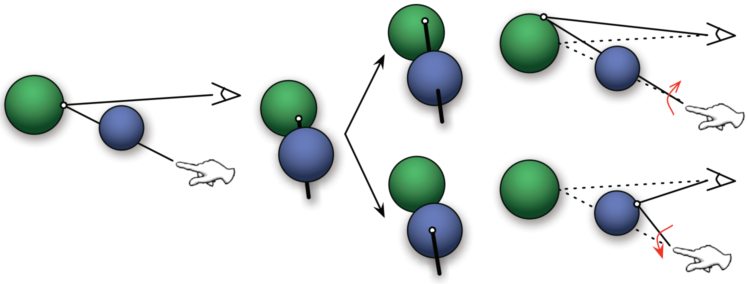

Figure 7. The worst-case scenario for RCE. Suppose the user wants to select the blue sphere, starting from the situation on the left. If the user relies only on the feedback ray to aim at the target (as in RC), the user will rotate his or her hand upward, moving the intersection point away from the target (top right). If the user relies only on the intersection point, the user will rotate his or her hand downward and will then be able to select the target (bottom right).

deviation of the number of errors was lower with RCE (σ = 12.1) than with RC (σ = 22.6).

Finally, we computed the Pearson correlation between the selection time and index of difficulty on a per-object basis (see Figure 9). We observed high correlation values with RCE for both the unoccluded (r = 0.972) and occluded (r = 0.902) models. This result provides further evidence of RCE's low sensitivity to mismatch. For RC, we found a correlation value of r = 0.898 for the unoccluded model. However, we found no correlation for the occluded model: as depicted in Figure 9d, a few objects (for example, objects 2, 3, 7, and 9 in Figure 5b) didn't adjust well to the expected positive correlation. We repeated the regression analysis ignoring these objects and then found a relatively high correlation (r = 0.755). Interestingly, when we contrasted these objects with those exhibiting high mismatch, we found a perfect match (see Figure 5b).

We derive two main conclusions from our regression analysis. First, RCE adapts well to Fitts' law (see the sidebar), being almost insensitive to mismatch, and an object's screen projection is a good measure of its effective width. Second, for RC selection, an object's screen projection isn't a good measure of its effective width because it doesn't account for mismatch.

A key feature of RCE is that the user's hand doesn't point exactly at the target; that is, vectors h and HQ aren't parallel, as Figure 4c shows. You could argue that this situation might interfere with the user's proprioception, potentially making pointing tasks more difficult. However, we found that users had no difficulty controlling the selection ray, even though we gave them no explicit explanation about the mapping during the experiment. Some users even didn't realize that they had used a different interaction technique in each block. RCE is comparable to RC when users align their hand with the pointing direction, as in Figure 4b, but users are free to place their hand in a more comfortable position, letting them, in effect, "shoot from the hip."

We could extend this research in several di-

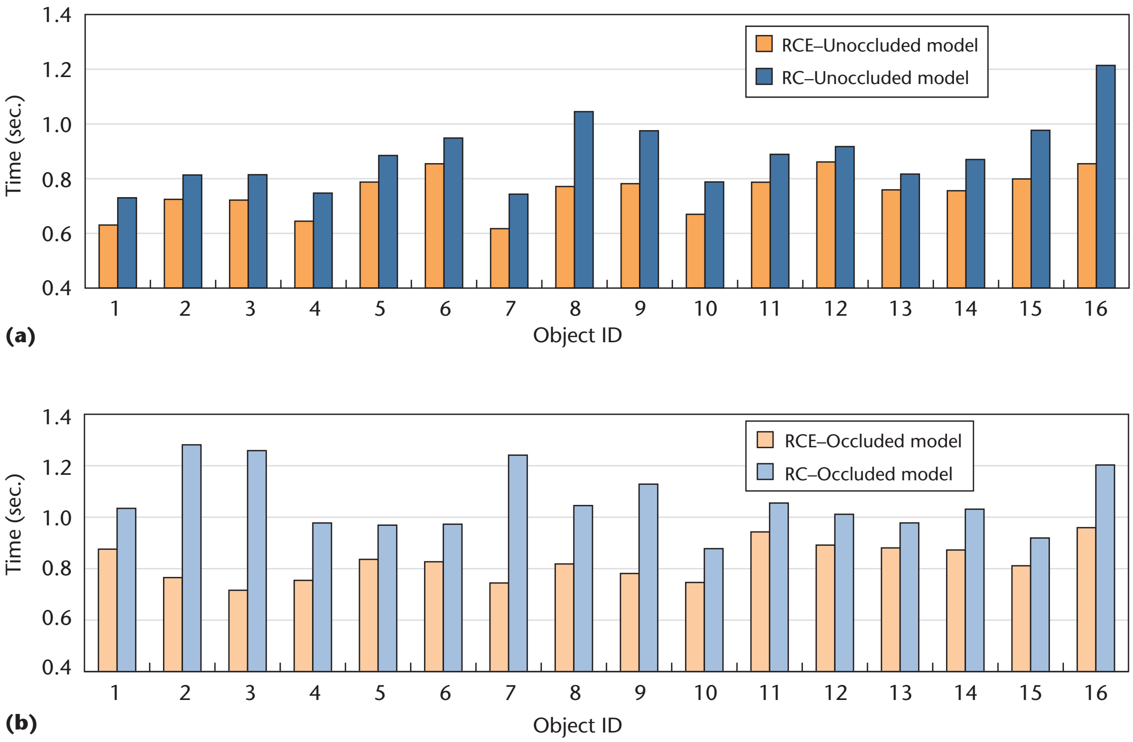

Figure 8. Mean selection time per object for the (a) unoccluded and (b) occluded models. Overall, selection time was lower for RCE; differences are more noticeable if the eye-hand visibility mismatch is higher (see objects 8, 9, and 16 in (a) and 2, 3, 7, and 9 in (b)).

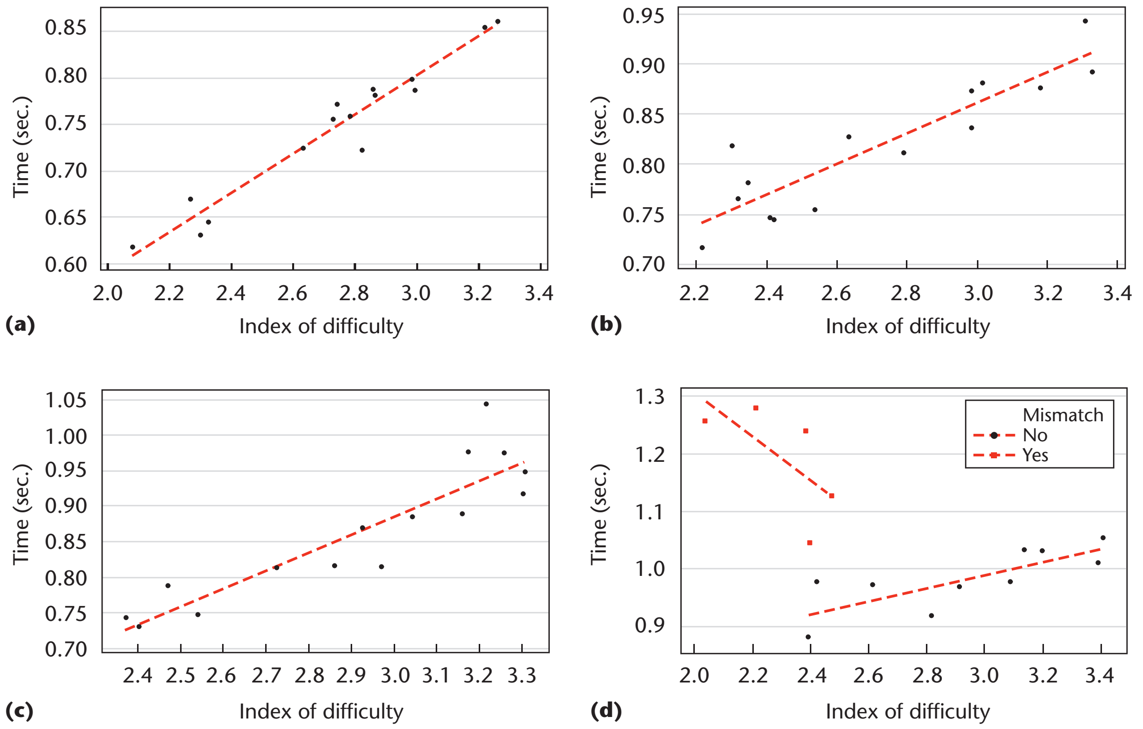

Figure 9. Scatterplots of the mean object selection time versus index of difficulty: (a) RCE, unoccluded model ($r = 0.972$); (b) RCE, occluded model ($r = 0.902$); (c) RC, unoccluded model ($r = 0.898$); and (d) RC, occluded model ($r = 0.755$). (We computed the correlation value for Figure 9d by ignoring spheres with high mismatch.) Whereas RCE adapts well to Fitts' law (see the sidebar) in both models, RC adapts well only in the absence of eye-hand visibility mismatch.

rections. We plan to explore how to use this new device-ray mapping in combination with manipulation tasks. We believe the best choice would be a hybrid approach such as Homer (Hand-Centered Object Manipulation Extending Ray Casting). For example, users can use RCE for selection and then switch to hand-centered manipulation. It might also be interesting to devise new visual-feedback techniques that minimize depth cue conflicts. We might accomplish this by placing the feedback ray's origin closer to the eye position, reducing the mismatch between the selection and feedback rays.

Acknowledgments

The Spanish Ministry of Science and Technology partially funded this work under grant TIN2007-67982-C02.

References

1. M. Mine, J. Frederick Brooks, and C. Sequin, "Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction," *Proc. Siggraph*, ACM Press, 1997, pp. 19–26.
2. J. Pierce et al., "Image Plane Interaction Techniques in 3D Immersive Environments," *Proc. 1997 Symp. Interactive 3D Graphics*, ACM Press, 1997, pp. 39–43.
3. A. Olwal and S. Feiner, "The Flexible Pointer: An Interaction Technique for Augmented and Virtual Reality," *Proc. 2003 ACM Symp. User Interface Software and Technology* (UIST 03), ACM Press, 2003, pp. 81–82.
4. L. Vanacken, T. Grossman, and K. Coninx, "Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments," *Proc. IEEE Symp. 3D User Interfaces*, IEEE Press, 2007, pp. 115–122.
5. F. Argelaguet, C. Andujar, and R. Trueba, "Overcoming Eye-Hand Visibility Mismatch in 3D Pointing Selection," *Proc. 2008 ACM Symp. Virtual Reality Software and Technology* (VRST 08), ACM Press, 2008, pp. 43–46.

Ferran Argelaguet *is a PhD student and member of the Universitat Politècnica de Catalunya's Modeling, Visualization, Interaction, and Virtual Reality Research Group. His research interests include 3D user interfaces, scientific visualization, and interactive rendering. Argelaguet has a BS in computer science from the Universitat Politècnica de Catalunya. Contact him at fargelag@lsi.upc.edu.*

Carlos Andujar *is an associate professor in the Universitat Politècnica de Catalunya's software department. He's also a member of the university's Modeling, Visualization, Interaction, and Virtual Reality Research Group. His research interests include 3D user interfaces, virtual reality, large-model visualization, and interactive rendering. Andujar has a PhD in software engineering from the Universitat Politècnica de Catalunya. Contact him at andujar@lsi.upc.edu.*