

Mapping 2D input to 3D Immersive Spatial Augmented Reality

Michael R. Marner*

Ross T. Smith†

Bruce H. Thomas‡

University of South Australia

ABSTRACT

This poster presents Viewpoint Cursor, a technique for mapping 2D user input from devices such as mobile phones, trackballs, or computer mice, to 3D multi-projector spatial augmented reality systems. While the ubiquity of input devices such as these make them obvious choices for spatial augmented reality, their 2D nature makes them difficult to use. Existing VR techniques rely on a display in front of the user's eyes on which to place virtual information. Immersive spatial augmented reality systems allow users to experience and interact with projected virtual information from any angle, using arbitrary placement of projectors. Viewpoint Cursor addresses these issues by mapping 2D input to a plane in front of the user's view. Ray casting is then used to find the 3D location for the cursor in the scene, which is then projected using the projection system. The user's position is tracked, with the input remapped accordingly, resulting in 2D input that matches what the user expects, regardless of their location.

Keywords: Spatial Augmented Reality, User Interfaces.

Index Terms: H.5.2 [Information Interfaces and Presentation]: Graphical User Interfaces—Input Devices and Strategies; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques

1 INTRODUCTION

This poster presents Viewpoint Cursor, a new interaction technique for Spatial Augmented Reality (SAR) systems. Viewpoint Cursor maps 2D user input to a 3D SAR environment, allowing users to intuitively select items and manipulate virtual information using common 2D input devices. SAR presents unique challenges to researchers, including the lack of an image plane in the user's field of view and the need for all virtual information to appear on surfaces in the environment. The ubiquity of devices such as the mouse, hand-held trackballs, and mobile phones make them compelling choices for SAR. However, standard cursor mapping is unsuitable, as it was designed for monitors placed side by side directly in front of users. SAR often uses projectors facing one another in order to project onto all sides of an object. This causes significant difficulty when selecting objects, as the user must move the cursor to the correct projector. In addition, SAR allows the user to view and interact with the scene from any angle, making 2D input difficult for the user to understand. Viewpoint Cursor solves these problems by remapping 2D input based on the user's view of the scene. Moving left with the input device will always move the cursor left, from their point of view. Cursor placement is accomplished using a ray-casting algorithm independent of projector locations. This removes the need to move a cursor into the correct projector's frustum. The User controls the cursor using a touch-pad app on a mobile phone, as shown in Figure 1.

*e-mail: marnermr@cs.unisa.edu.au

†e-mail: ross.smith@unisa.edu.au

‡e-mail: bruce.thomas@unisa.edu.au

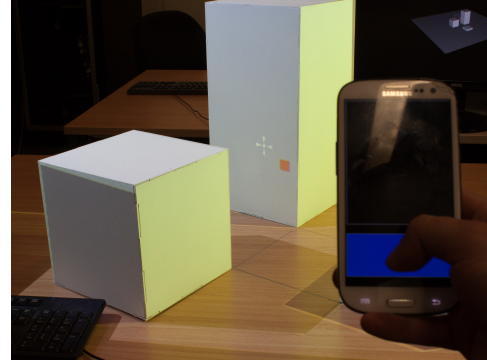


Figure 1: A user interacts with a SAR system using Viewpoint Cursor and a mobile phone.

2 BACKGROUND

Mapping 2D input to 3D Virtual Environments (VE) has been an active research area. Vincent et al. investigated several cursor based selection techniques for hand-held AR [7]. Grossman and Balakrishnan investigated pointing using volumetric displays [2]. Thomas [6] evaluated three 2D input devices for selection and annotation in AR. However, this approach required showing information on the image plane in front of the user. However, these techniques depend on either the ability to show virtual information in mid-air, or having a known image plane in front of the user (such as an HMD). SAR environments do not have these features.

Several SAR systems have used a tracked stylus as their primary input technique [1]. A key limitation of these approaches is the need for an accurate tracking system to track the position and orientation of the input devices. Laser pointers are an alternative approach for interaction with SAR systems. Kurz et al. [4] demonstrated how laser pointers can be tracked and used as input devices for SAR systems. Their approach required custom hardware for their tracking cameras. However, other research has shown the usability issues affecting laser pointer interaction [5], especially when interacting at a distance. Recently, Harrison et al. [3] have demonstrated using a depth sensor and shoulder worn projector to provide natural gesture based input.

3 VIEWPOINT CURSOR

Viewpoint Cursor works by remapping 2D input from input devices such as a touch-pad to the point of view of the user. The user's position must be tracked in order to accomplish this. Viewpoint Cursor maintains a 2D Device Cursor and a 3D Scene Cursor. The 3D Scene Cursor is what the user sees projected in the environment. The 2D Device Cursor is used to maintain the mapping from the device to the user's viewpoint.

3.1 Controlling Input

Input from the device is mapped to the Device Cursor on a *Viewpoint Plane* in front of the user. This mapping onto the Viewpoint Plane ensures input to match what the user expects, based on their point of view. The Device Cursor's initial position is in the centre

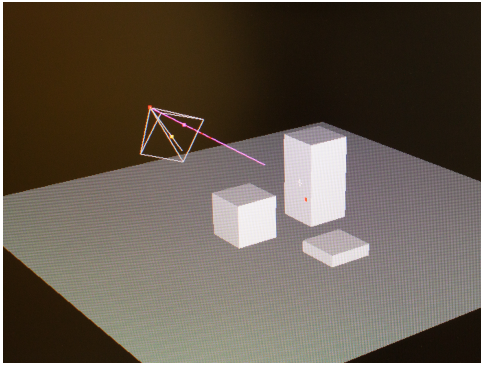


Figure 2: An overview of Viewpoint Cursor. The frustum indicates the user's position and orientation in the scene. The purple dot shows the 2D Device Cursor location mapped to the Viewpoint Plane. The purple line shows the ray cast into the scene to calculate the Scene Cursor location, with the Scene Cursor shown in red on the tall box.

of the user's view. However, the Scene Cursor does not appear until the user starts interacting with the system. The Scene Cursor is then placed in the scene using the approach described in Section 3.2.

Both the user's position and orientation need to be tracked in order to maintain the Viewpoint Plane's position in front of the user. The user's position is provided directly by the tracking system. We have provided two ways of calculating the user's orientation. In table-top SAR systems, it is safe to assume the user is always looking at the scene whilst interacting. For larger scenes where the user can truly be immersed in the environment, this assumption is unsuitable. Instead, we make use of the digital compass common in mobile phones. We assume the user will hold the phone like a remote control, pointing it in roughly the direction they are looking.

3.2 Scene Cursor Placement

The Scene Cursor is placed into the environment using a ray-casting algorithm. A ray is generated from the user's position, in the direction through the Device Cursor on the viewpoint plane and into the scene. Intersection tests are then performed against all objects in the scene, and the cursor is placed at the closest point of intersection. Placing the cursor in 3D allows Viewpoint Cursor to function independently of the projector configuration. There is no need to move a cursor between projectors, as there is when using the system cursor, because the Viewpoint Cursor's position is calculated based on the 3D SAR scene.

Additional work is needed when there are no intersections, such as scenes with multiple objects placed apart from one another. Otherwise, it would be impossible for the user to move the cursor between objects. Viewpoint Cursor provides two complimentary solutions to this problem. The first is to use a *floor plane* in addition to the objects of interest. Intersection tests are performed against the floor plane, allowing natural movement across the table surface between objects. The second approach involves an additional *back plane*, placed at a large distance in front of the user. This plane serves as an intersection of last resort. If none of the objects in the scene, or the floor plane, intersect with the ray, the back plane will. The back plane allows natural cursor movement between separate objects, using the Viewpoint Cursor mapping. The combination of these intersection cases allow Viewpoint Cursor to function regardless of the physical objects present in the scene.

3.3 Reverse Updates

For usability, the Scene Cursor's location is only updated when the user interacts with the input device. This allows the user to freely move in the environment whilst keeping the cursor at a constant

location. To prevent the cursor jumping to new locations, Viewpoint Cursor performs *Reverse Updates* as the user moves around in the environment and not interacting with the input device. Here, a ray is generated from the Scene Cursor's location, in the direction of the User. This ray is intersected against the Viewpoint Plane, and the Device Cursor's position is set to this new location.

4 IMPLEMENTATION DETAILS

Our demonstration system is running on a standard desktop PC with two NEC NP510W projectors, and a Microsoft Kinect is used to track the user's position. The Touchpad was implemented using Control¹, communicating with the host computer using the Open Sound Control [8] protocol and running on a Samsung Galaxy S3. Our implementation of Viewpoint Cursor can be found at <https://github.com/WearableComputerLab/ViewpointCursor>.

5 CONCLUSION

This poster has introduced Viewpoint Cursor, an interaction technique that maps 2D user input to 3D immersive SAR environments. Whilst we demonstrated Viewpoint Cursor using a touch pad implementation on a mobile phone, the technique has been designed to work with any 2D user input including computer mice. A key benefit of the technique is there is no need for expensive tracking equipment. The user can be tracked using a low cost device such as the Microsoft Kinect. This makes Viewpoint Cursor ideal for low cost, portable, SAR systems. In the future we will evaluate Viewpoint Cursor against other 2D input techniques.

ACKNOWLEDGEMENTS

This work was supported in part by a grant from the Australian Research Council - Discovery Grant DP120100248.

REFERENCES

- [1] D. Bandyopadhyay, R. Raskar, and H. Fuchs. Dynamic shader lamps: Painting on movable objects. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 207–216, 2001.
- [2] T. Grossman and R. Balakrishnan. Pointing at trivariate targets in 3D environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 447–454, Vienna, Austria, 2004. ACM.
- [3] C. Harrison, H. Benko, and A. D. Wilson. OmniTouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST '11, page 441.450, New York, NY, USA, 2011. ACM.
- [4] D. Kurz, F. Hantsch, M. Groe, A. Schiewe, and O. Bimber. Laser pointer tracking in projector-augmented architectural environments. In *International Symposium on Mixed and Augmented Reality*, pages 19–26, 2007.
- [5] B. A. Myers, R. Bhatnagar, J. Nichols, C. H. Peck, D. Kong, R. Miller, and A. C. Long. Interacting at a distance: measuring the performance of laser pointers and other devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pages 33–40, New York, NY, USA, 2002. ACM.
- [6] B. Thomas. Evaluation of three input techniques for selection and annotation of physical objects through an augmented reality view. In *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*, pages 33–36, 2006.
- [7] T. Vincent, L. Nigay, and T. Kurata. Handheld augmented reality: Effect of registration jitter on cursor-based pointing techniques. In *Proceedings of the 251èMe Conférence Francophone on L'Interaction Homme-Machine*, IHM '13, pages 1:1–1:6. ACM.
- [8] M. Wright and A. Freed. Open sound control: A new protocol for communicating with sound synthesizers. In *Proceedings of the 1997 International Computer Music Conference*, page 104.

¹<http://charlie-roberts.com/Control/>