

Tangible Navigation and Object Manipulation in Virtual Environments

Andy Wu¹, Derek Reilly², Anthony Tang¹, Ali Mazalek¹

GVU Center¹

Georgia Institute of Technology
85 5th St. NW, Atlanta, GA 30318, USA
{ andywu, tonyt, mazalek }@gatech.edu

OCAD University²

100 McCaul St.,
Toronto Ontario, M5T 1W1, Canada
dreilly@faculty.ocad.ca

ABSTRACT

In this paper, we introduce approaches to navigating and manipulating objects in a Collaborative Virtual Environment (CVE) that engage tangible objects and an interactive table interface. We also identify three design concerns that are common to physical-virtual connectivity for interaction with CVE systems. At last, we propose solutions to these issues within the context of CVEs.

Author Keywords

Tangible User Interface, TUI, HCI, Tabletop, 3D interaction, virtual world.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Design

INTRODUCTION

Collaborative virtual environments (CVEs) were designed with the promise of supporting collaboration in a virtual, three-dimensional world: participants would be able to navigate through the world, and interact with virtual objects just as they would in the real world. CVEs have seen adoption in primarily two areas: gaming (e.g. first-person shooters, and massively multi-player online role playing games—MMORPGs), and in massively multi-user virtual worlds that are primarily social/recreational (e.g. SecondLife). While there has been interest in using CVEs for other purposes (meetings and collaborative work, for example), adoption remains low.

One of the core obstacles to more widespread adoption of CVEs is the absence of easy-to-use interaction mechanisms: first, to navigate through the virtual world, and second, to manipulate the virtual objects.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

TEI'11, January 22–26, 2011, Funchal, Portugal.

Copyright 2011 ACM 978-1-4503-0478-8/11/01...\$10.00.

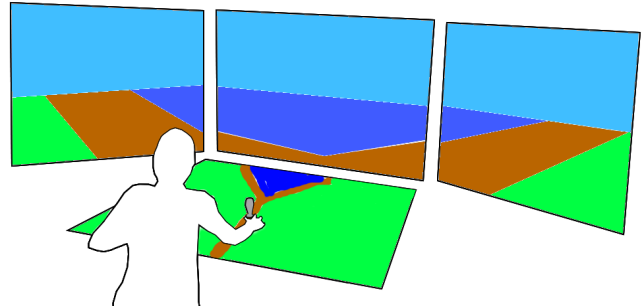


Figure 1. The user operates a grey tangible avatar on the tabletop that navigates through the virtual world. The coordinated view of the world on the upright displays shows that the avatar is at the fork in the road.

While most CVEs adopt a conventional keyboard+mouse scheme that works well for planar navigation, designers have been hard-pressed to find comparably easy-to-use mechanisms for full six-degree-of-freedom navigation (x, y, z position, and yaw, pitch, roll).

CVE systems typically only provide very crude methods of interacting with and manipulating objects in the virtual world: very often a large subset of objects cannot be moved, and those that can may only be moved along each plane (independently) in the virtual world. Changing the orientation of these objects (yaw, pitch, roll) is even more challenging. In general, this problem has been considered as a mismatch between the perceptual structure of the interaction task and the control space of the input device (i.e. that the desired degrees of freedom does not match the interaction device's available degrees of freedom) [12].

To support navigation, many CVE interfaces provide small inset maps. Because these maps are simplified, they may not provide enough detail to make navigation decisions relevant to collaboration: where a certain document is located, who is around, etc.

Top-down perspectives are common when configuring virtual environments, and anytime multiple virtual objects need to be arranged. A tangible+tabletop configuration offers interesting interactive possibilities for configuring virtual environments [20], both before and during collaboration.

In this work, we present approaches for navigating through and manipulating objects in a virtual world that employ tangible objects and a tabletop interface. As illustrated in Figure 1, tangible objects (tracked via a combination of cameras mounted above and under the table surface) are paired with objects in the virtual world: manipulating the tangible results in a corresponding action in the virtual world. The primary benefit of this mapping is that it allows users to fluidly manipulate virtual objects with a full six-degrees-of-freedom afforded by the physical object in the real world. Another benefit of this approach is that the tabletop provides a concrete perceptual frame of reference, absolutely positioning actions according to the view on the table. Thus, our tabletop interface provides a more concrete form of direct interaction when compared to typical 3D mouse/input devices, which only provide an indirect, relative form of interaction. Further, by devoting the tabletop display to presenting a live, interactive and view-controllable overview of the virtual environment, we provide a much richer information and interaction resource than the typical inset map.

While this approach has been foreshadowed and realized in prior research (e.g. [3]), the problems of the approach have rarely been addressed. From the context of supporting multi-user interaction in a CVE, we articulate three problems that arise from this approach: (1) potential for inconsistency between physical object and virtual object; (2) managing physical and digital constraints, and (3) controlling granularity. We illustrate several solutions that address and mitigate these problems within the context of CVEs, and begin the process of articulating a more general solution space.

This work makes three main contributions: first, we provide a system that supports navigation (and object manipulation) in a CVE using tangible objects; second, we articulate three problems that are common to systems that provide tangible control of virtual objects, and finally, we propose solutions to these problems within the context of CVEs.

RELATED WORK

Our general approach applies Tangible User Interfaces (TUIs) as a way to better connect users with virtual worlds by creating physical controls for digital information. By coupling digital information with everyday objects, people can apply the skills and practices used in their daily life when interacting with virtual worlds [5, 11, 23].

Some of the earliest work using tangible interfaces for 3D information focused on using physical objects to aid in the creation of 3D digital geometry [6, 7]. Triangles [9] is a canonical example in this space: users manipulate digital information using a physical construction kit. By tiling objects in the physical world, users could construct both 2D and 3D constructions in the physical world. This topography would be reflected through a connected computer, and updated in real-time based on the manipulation of the physical pieces.

Tangible Surrogates for Virtual Objects

The general approach of linking physical objects with objects in a virtual world has been explored in a variety of ways. Buxton [2] provides a sketch of a system where tangible props can be physically manipulated and placed on a stage. These props are linked to a 3D visualization of a car (in a virtual world). Gillet et al. [8] apply this approach to the domain of molecular visualization, using a 3D printer to create a set of molecules as props. The generated visualizations are then applied to an Augmented Reality (AR) scene with the virtual molecule. We call this approach for manipulating virtual objects *tangible handles* (phidgets in [11]), where physical objects represent a virtual object. Kruszyński et al. further extend this approach by allowing users to employ a physical stylus as a probe to explore both the tangible prop (in their case a coral structure) and the digital visualization of the prop [15].

Using a similar approach, Metoyer et al. developed a tangible tabletop system to coach the movements of animated football characters at a high-level [18]. Their system employs a miniature football field with small tangible players on the tabletop. Users control the movements of the tangible characters during the simulation of moves in a football game. Their work focuses on building timing of players' routes through the virtual world (i.e. the football field) in relation to one another. The system makes use of a projected tabletop as a concrete positional frame of reference, in a manner similar to our own use of the tabletop display.

Other approaches have provided fine-grained, articulated control of a virtual object and scene. In TUI3D, Mazalek et al. instrument a marionette puppet to control a virtual character [16] as a means to address the production and performative challenges involved in creating machinima (employing real-time 3D CGI for performance).

Tabletops in Collaborative Virtual Environments

The potential benefits of tabletop interfaces for collocated collaboration have been articulated and explored elsewhere [21]. A tabletop plus vertical display configuration promises some of the benefits of face-to-face interaction when working with remote collaborators. This has been explored using video streams by Tang and Minneman [22], Yarosh et al. [24] and others, and also for collaboration using a CVE by Hansson et al. [10] and Regenbrecht et al. [19]. By default, CVEs support group configurations in which all collaborators are remote. Tabletop CVE interfaces may also facilitate mixed-presence configurations, where some collaborators are collocated and share a single tabletop interface.

Navigating Virtual Worlds

Navigation is a fundamental activity in interactive virtual environments. Several researchers have explored how virtual worlds can be navigated using tangible approaches.

Kim et al. developed a TUI system for Cave Automatic Virtual Environment (CAVE) applications [14]. Their

system used an AR technology and a regular tabletop surface for space navigation in a virtual environment (VE). They attached fiducial markers on physical objects that serve different purposes, e.g. proxies of virtual objects, controls of navigation in VE, paddles for simulating mouse clicks. Their AR navigation tool is very similar to a joystick, which controls the speed of the virtual object while our tangibles control the location of the object (or avatar). We build on this work by allowing the user to dynamically change the zoom level to gain more precise maneuverability and manipulation of virtual objects.

Finger Walking in Place is a planar navigational technique for virtual environments (presented on an upright display) using a tabletop surface [13]. A user employs finger gestures that are then mapped to moving commands like forward, backward and rotation in a walking area on the tabletop.

Bowman et al. [1] conducted a series of quantitative studies comparing several types of 3D navigation techniques. They discovered that “pointing” techniques are faster relative to “gaze-directed” steering techniques. Pointing techniques also have the advantage of decoupling ‘look direction’ and direction of movement. Our tangible+tabletop configuration permits a similar kind of decoupling of attention/gaze from movement.

Everyday Navigation in 3D Worlds

The problem of navigating 3D worlds was first appreciated in popular culture with the rise of first person shooter (FPS) games. Early control schemes in planar FPSs mapped to forward/backward and horizontal rotation movement. Within a short period of time, many of these users began using the mouse for rotation combined with left/right keyboard keys for strafing—a combination that is difficult for novices to learn, but has become the de facto method for navigating in FPS and MMORPGs.

At the same time, there have been many commercial attempts to bring more “natural” and “easy to use” interaction mechanisms for 3D environments. Such devices (e.g. 3D mice) have not achieved much popularity beyond computer-aided design applications.

APPROACH

To provide a system that supports navigation and object manipulation in a CVE, we believe building tangible controls integrated with a top-down view helps users to understand the 3D virtual world and provides more natural interactions (see Figure 2).

An interactive table has a large surface that is natural for a group of people to work together around, and is an ideal platform for 2D planar navigation in virtual space without requiring frequent context switching. Our interactive tabletop display combining TUI with AR techniques expands the physical working space from the 2D tabletop to include 3D space above the table surface.

Prototyping Tabletop Interfaces with TwinSpace

Our tabletop CVE interfaces were built using the TwinSpace architecture [4] for collaborative cross-reality. TwinSpace was originally designed to couple activities in a virtual world with activities in a physical space in order to support mixed-presence (collocated and remote) collaborative work. TwinSpace uses Open Wonderland as an MMO collaborative virtual world platform and the EventHeap service to communicate across diverse devices and services in the physical and virtual.

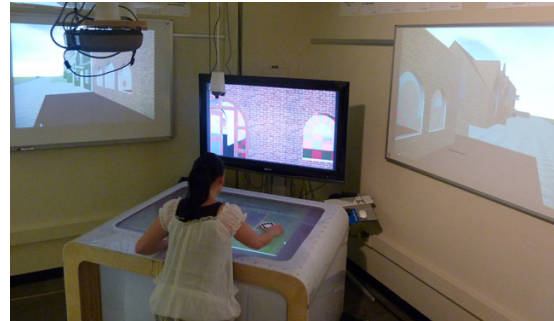


Figure 2. The user navigates through, and interacts with objects in the virtual world using the tangibles on the tabletop. A coordinated view of the world (as would be seen by the avatar—also a tangible) is presented on the upright displays.

A range of tabletop technologies has been integrated into TwinSpace-based projects so far. One uses a top-down IR camera and IR pens to interact with digital content projected on the tabletop from a top-down projector. Another is the Tangible Tracking Table (TTT) [17], an interactive tabletop display that tracks multiple objects and finger touches on its surface. The TTT has been used in a number of ways as a virtual world interface: to permit camera control, object manipulation and virtual world navigation. Top-down, 3D fiducial tracking has also been used on different projects with a horizontal LCD display, the TTT, the IR pen table, and on a static horizontal surface. We base our assessment of the design challenges of tabletop CVE interfaces (and our proposed solutions) on the collected experience of design and evaluation across these multiple projects.

Most of these projects involved some form of tangible+tabletop interaction. These interactions fall under two broad categories: virtual object manipulation, and avatar navigation. Our experiences with each are discussed below.

Object Manipulation with Tangible Props

A user is configuring a room for a 3D online gallery. She walks to an interactive table, which presents a floor map of the virtual gallery. Each category of virtual object in the gallery (e.g. tables, seating, 3D artwork, 2D artwork) has its own physical prop. Specific virtual objects (e.g. a specific piece of 3D artwork) are added to a default location in the gallery by selection using the table’s touch-based menu controls. After adding a specific 3D artwork

she then picks up the physical prop representing a 3D work, hovers over the newly instantiated object until it becomes active, and positions the prop on the tabletop where she wishes the specific art piece to be located. She releases the virtual object by hiding the fiducial on the prop. She finishes her job after placing all pieces on the tabletop to their positions.

The first environment built using TwinSpace was a physical collaborative team room. One of the key elements of this space is a tangible+tabletop interface. Using a tabletop interface with physical objects was an effective way to manipulate objects on a plane in the 3D virtual world. As we move the physical objects, the corresponding virtual objects change simultaneously. This simple interaction was found to be helpful in several applications: deploying furniture in a room, collaborative discussion over slides, moving groceries in a virtual supermarket, and positioning portals in a game space. In a traditional Wonderland client, a user has to select an object, launch the edit window and control the coordinate values using keyboard and mouse, providing high accuracy but more difficult manipulation.

The 2D object manipulation technique works well for objects that lay on a flat surface (such as furniture and some documents) or reside on the same plane. In several of our prototypes we considered extending the interaction to include 3D positioning. This can be achieved by extending the interaction space to include the region above the table (i.e. by lifting the tangibles above the table surface), and/or by allowing the view on the table to zoom or change vertical position or orientation. The first technique has been explored for adding and removing documents from a shared workspace, and for controlling a peripheral camera view of the tabletop region. We are exploring the latter technique in our game prototype. A combination of these techniques has been explored for navigation control, discussed next.

Navigation using tangible avatars

A visitor comes to our interactive table to explore our 3D virtual campus. On the table, there is an action figure, which serves as a tangible avatar. The tangible avatar controls a virtual avatar in the virtual world. Behind the interactive table stand three large vertical displays. The three vertical displays present the virtual avatar's perspective. As he glides and rotates the action figure on the tabletop, the vertical displays respond by showing the avatar's updated perspective. .

In addition to using an interactive tabletop display as a main working and viewing area, we use it as an interface to control avatars in the 3D virtual world. Most 3D computer games use keyboards, mice, joysticks or keypads to navigate the 3D virtual world. Usually, these games use first person views or third person perspective views for users to see the 3D world. Our interactive tabletop provides a top-down view of the virtual world. To move an avatar in the 3D world, instead of using a traditional controller, we use a tangible avatar (an action figure) to move the avatar in

the virtual world. When a user moves the action figure forward, the avatar in the virtual world moves forward; when the action figure rotates, the avatar rotates in the same way. Figure 3 shows the actions performed on a tangible object on the tabletop – moving and rotating.

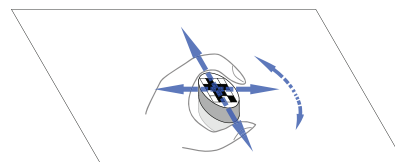


Figure 3. A tangible avatar can be moved and rotated on the tabletop.

However, moving an object on a tabletop (a two dimensional movement) does not provide a 3D navigation experience, unless elevation changes occur automatically, e.g. as a result of walking on stairs. The TwinSpace framework permits the simultaneous updating of views on multiple displays. In our prototypes we have used this to couple top-down and first-person views. The configuration in this section uses three vertical displays to provide a wide egocentric view of the 3D space (see Figures 1, 2). A user stands in front of the TTT and moves the action figure on the tabletop to navigate the 3D world. The interactive tabletop has become a TUI that provides visual feedback and direct manipulation of a physical avatar.

Changing level of details and navigating with a physical tangible object

A tabletop display shows only part of the virtual campus. When the visitor wants to see the whole virtual world, he has two options. He can use the tangible avatar and hand gestures one the tabletop to explore the map, or he can lift the tangible avatar. As the tangible avatar moves further away from the table surface, a larger unseen region of the map is revealed. Therefore, he can pickup the action figure, navigate the campus from different levels in the air, and place it at a different place.

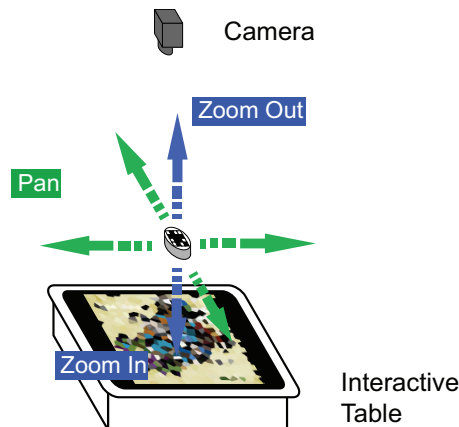


Figure 4. 3D navigation on GoogleMap.

Navigation with an object on an interactive tabletop gives direct visual and proprioceptive feedback of the location of an action figure. In 3D environments that only allow ground

navigation, this implementation may be sufficient. However, many 3D games permit navigation in the vertical direction, and content-centric CVEs often exploit the vertical dimension. Therefore, we extended our tabletop interface to permit 3D navigation.

Our concept is to navigate the ground using the tangible tabletop interface, and navigate the 3D space when a user lifts the object in the air (see Figure 4). To implement this, we attached a second camera on top of the interactive tabletop to detect the 3D position of objects on the tabletop. We use NyARToolkit to detect a fiducial marker attached to the tangible object, and translate the physical location and orientation of the object into a 3D virtual world transformation matrix. This matrix is sent out through the TwinSpace infrastructure to update the virtual avatar's location and the views on the vertical displays.

We tested a number of related interaction concepts using Google Maps. In this case, when we raise the object, the map zooms out; when we put the object down, the map zooms in. We can also move the object in the horizontal to pan the map at any specific level.

LINKING PHYSICAL & VIRTUAL OBJECTS: DESIGN CHALLENGES

In designing these interactions, we encountered several design challenges that would likely also be encountered when linking virtual entities with tangible objects in other contexts. These are: potential inconsistency between tangible object state/location with virtual object state/location; potential mismatch between physical/virtual constraints, and the need for managing granularity of control.

Physical/Virtual Object Inconsistency

When the physical object is the sole input mechanism for manipulating the virtual object, then position and state information is unlikely to be a problem. There are two cases when inconsistency can become a problem: first, when there is another method to manipulate the state/location of the object (common in a CVE, where another user can change the location of an object), and second, if the state/location is represented via a *relationship*, and that relationship can be changed by another method. We return to this latter case more explicitly later when we describe how granularity of control is achieved.

Physical/Virtual Constraints

An inherent problem in linking physical and virtual objects is that our interactions with these objects in their different manifestations are subject to different constraints. For example, in the physical world, we might place a letter-sized digital document on a meeting table and pass it around to share it; in the virtual world, we might dispense with a meeting table altogether, and scale the document so that the content is legible from a distance.

Virtual objects have artificial constraints in the virtual world. For example, using arrow keys to move an avatar allows only linear movement, the boundary of the 3D

virtual world restricts the objects to move further, and the physics in the virtual environment limit the way objects move.

Similarly, physical objects in the real world have intrinsic constraints (their weight, size, shape). An interactive tabletop provides a physical affordance (the tabletop surface) to support objects moving on its surface, but it constrains the object's freedom of movement in the vertical axis (and certainly, we cannot leave an object in midair!). There are also technical constraints in the physical world. For example, the view field of the lens of our tracking cameras create an invisible boundary for these physical objects and the discriminability of the optical sensor limits the minimum size of the marker on an object, and hence the size of the object itself.

Another constraint is the difficulty of manipulating physical objects from a virtual environment. For example, while we can manipulate the location of a virtual object when its physical surrogate is moved in the real world, when a user moves a digital document in the virtual world, the physical document does not move.

Since the constraints in these two worlds are different, we have to adopt transformations or mappings to manage them. In some cases, these mappings are straightforward (e.g. to solve the constraints of size mismatch between the tabletop and the virtual world is to scale the image of virtual world down to the size of the tabletop); however, in many cases, the "correct" mapping is a matter of interpretation.

Managing Granularity of Control

One constraint we employ in our tabletop+tangible prototype is the physical frame of the tabletop. As stated earlier, it provides a concrete frame of reference for users' interactions: their interactions on this frame of reference are thus absolute rather than relative.

Yet, if the user moves the tangible object 1" on the tabletop, how much should that move in the virtual world? How much of the world should the user see on the tabletop? Likely, the answer depends on the context: sometimes, it may be desirable for fine-grained control and positioning, while in other cases, the user may need to move long distances across the world. How can the user change this granularity of control dynamically?

Mental models of panning issues

In demos to visitors of our research center, we found differing expectations of how panning should work. There are two types panning on a map. When a user moves an object from location A to location B on the tabletop, this has two interpretations. In Figure 5 (a), a tangible object is about to slide to the right. Figure 5 (b) shows the navigation method that is commonly used for Google Map navigation on the iPhone. Figure 5 (c) shows the navigation method we are used to when we navigate a physical paper map or slide a scroll bar in a GUI. These two navigation strategies correspond with different mental models of navigation:

users either employed a "subject-centric" (5b), or an "object-centric" (5c) perspective.

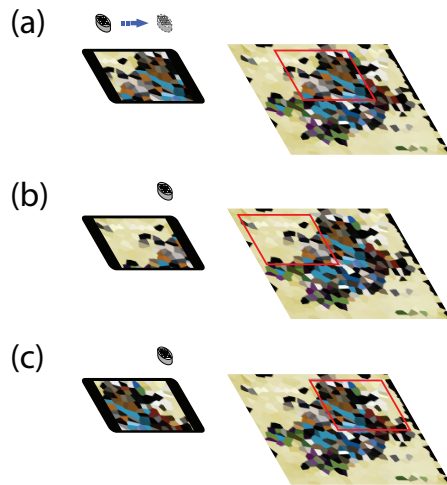


Figure 5. The movement of a tangible object and the two possible navigations on an interactive tabletop display.

ADDRESSING THE CHALLENGES FOR CVES

In this section, we outline how we addressed the design challenges identified in the previous section in our prototypes. We believe our approaches have wide applicability for object manipulation and avatar navigation in CVEs.

Inconsistency: Notify, Copy, Lock Virtual

We encounter the inconsistency problem in a cross-reality environment when a user manipulates an object in the CVE that is associated with a tangible handle. For example, a user may move or rotate a virtual document. We designed two solutions to this problem: notify and copy

Notify (a.k.a. Halo). In this design solution, we simply notify the users of the inconsistent state using a halo or colored aura drawn around both the virtual object (in the virtual world) and on the tabletop around the physical object. This halo also shows the link between the two objects to allow users to resolve the inconsistency by (for example) moving the one of the objects to the other object.

Copy. Creating a duplicate of the inconsistent object can satisfy every individual trying to control it. This is useful when several groups of people from different locations want to read one document.

Lock Virtual. A third possible solution that we did not implement was simply to lock virtual objects that have been associated with a tangible handle. In such a solution, users in the CVE simply are not allowed to manipulate the properties/state of such virtual objects.

Physical and Virtual Constraints: Clutch, Proxy, Elastic Link

Clutch. Permitting vertical movement by raising or lowering a tangible object partially overcomes the physical constraint of the 2D tabletop surface for 3D navigation. However it is difficult to move an object horizontally in

mid air for extended periods, and impossible to leave it there. Whether the tangible controls the elevation of the tabletop view or the 3D position of an object relative to that view, we require a mechanism for turning elevation control on or off. A clutch can be implemented on the tangible (e.g. through a fiducial-based control) or on the tabletop interface.

Proxy. We have prototyped applications in which each virtual object in the virtual world has a physical representative in the real world, however this is problematic when there are many manipulable objects, or when the set of objects can change dynamically. In such cases a fixed set of tangible proxies can be assigned to virtual objects dynamically. Proxies are representative of the virtual object, rather than a physical counterpart to it. Proxies require a mechanism for attaching and removing their association with specific virtual objects.

Elastic Link. When there are three or more tangible objects on the tabletop, the objects and their underlying digital representation have become difficult to manage. Our tentative solution is not to create digital avatars right under the associated physical objects. Instead, displaying the digital object next to the physical object with a visual link between them. One reason for an elastic link is so the physical object will not block dynamic digital information and permit interactions beyond movement in space (as is the case with editable documents, for example). Another reason is when the map zooms, or pans, digital objects can move accordingly, and a call-out would appear that visually links the physical location of the tangible object with its virtual object's location.

Granularity: Coupled and Decoupled Zooming

The 3D tangible object manipulation method mentioned before provides zooming naturally. One can see the detail by moving the tangible object toward the tabletop and see the whole map by moving the object away from the tabletop. This is known as a coupled method that links zooming with the tangible.

Another navigation method is to use a tangible object with hand gestures when the tangible object is placed on the tabletop—where zooming is decoupled from the tangible object itself. In our design, moving a tangible object on the tabletop means panning the map, while rotating this object means changing the orientation of the avatar. What if we want to rotate the map or zooming into an area and keep the action figure on the tabletop? There are many reasons to do this. First, raising an avatar up and down can cause fatigue to users. Second, in 3D worlds where most activities occur on the ground, users may want to quickly zoom into the object. If we can achieve this with tangible objects staying on the table, this can save users effort.

The new system provides interactions using tangible object and finger gestures. In Figure 6, the tangible object is held still. (a) and (b) use one finger to rotate and zoom the map. These two interactions can be performed together, so a user

can draw to rotate and zoom the map at the same time. (c) uses two fingers to pan the map. As a result, a user can manipulate her avatar on a 2D plane by moving or rotating the physical object. She can zoom out to have an overview of the map with one figure gesture. She can also rotate the map to have a front view of the avatar's face. If she does not want to change the granularity, she can use two fingers to pan the map as well.

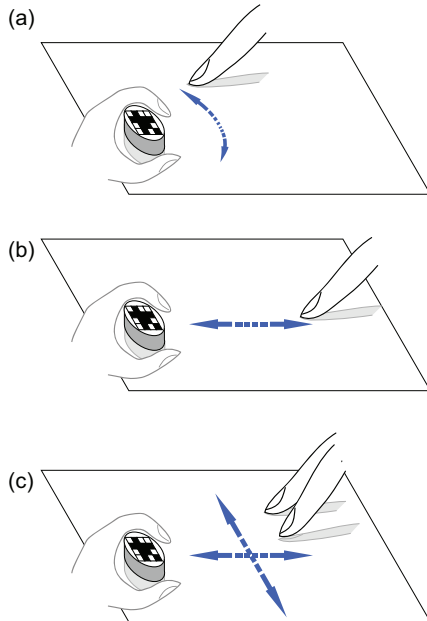


Figure 6. Tangible interaction with finger gestures: (a) gesture for rotating; (b) gesture for zooming; (c) gesture for panning.

Our 3D GoogleMap navigation concept began from this observation that one usually has to zoom out before panning. Therefore, creating a system that performed this transformation automatically was our goal. The 3D GoogleMap navigation was based on this scenario:

A map navigator stands at our table. His current location on the map is at Atlanta. He wants to move the center of the map to New York. When he raises the physical object from Atlanta, the map starts to zoom out. At some point, he is able to see New York at the edge of the map. He moves the object toward New York on the map. Slowly has New York moved toward the object. Finally, he places the object on top on New York City on the map.

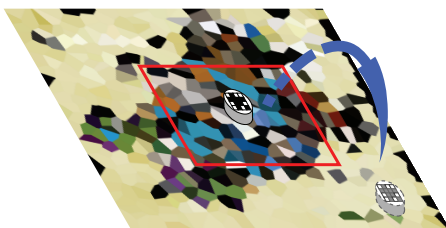


Figure 7. The concept of navigation using a tangible object. The destination is out of current tabletop's view, which is bounded with red lines.

In other words, a user simply pickups the tangible prop, and put it onto a destination location as shown in Figure 7. Figure 8 illustrates the decomposed actions in four steps.

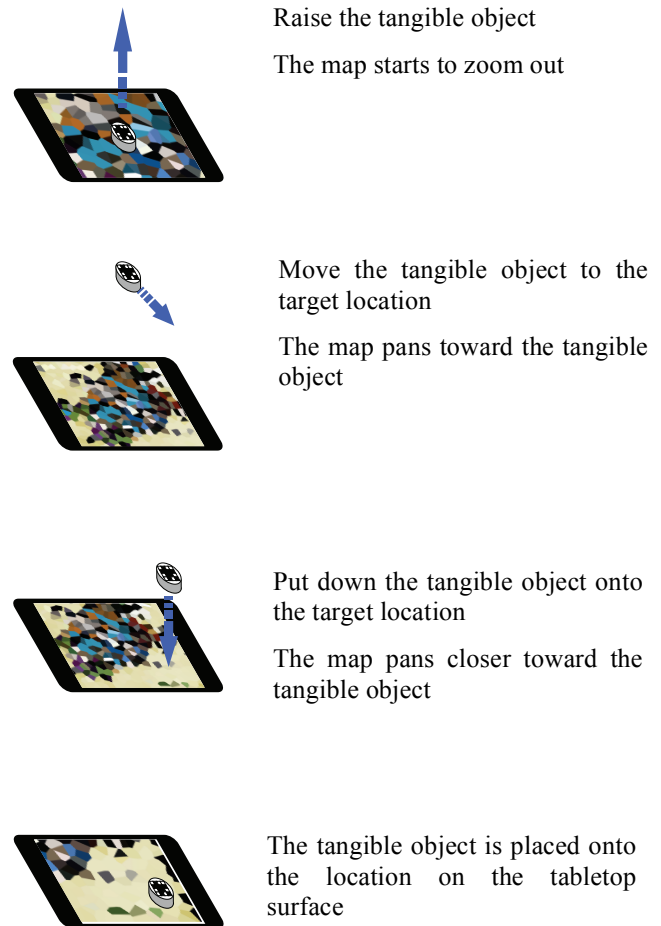


Figure 8. Moving a tangible object from one location to another.

To achieve this, the panning mechanism we chose was the one illustrated in Figure 5 (c). When the scenario described above happens, the navigator has to move the tangible object to the target location. However, this target location keeps moving toward the object's projected location. This navigation requires some practice to master. Yet the distance between the tangible object and the target location is provided visually and their relative movement is predictable.

CONCLUSION

We have presented results from our prototyping efforts involving object manipulation and avatar navigation in collaborative virtual worlds using tangible+tabletop interfaces. In particular, we present two new types of CVE navigation methods; one combines a tangible with finger gestures and the other builds on the action of relocating an object on a table. We summarized three issues that are common to using tangibles for object manipulation and

navigation in CVEs and finally, we proposed solutions to these problems within the context of CVEs.

One possible extension to this work is to apply the Google Maps 3D navigation concept inside 3D games. So that the zoom level of the radar map reflects the avatar's vertical position. We can also adjust the three vertical displays' views from fixed-angle views to the view from the virtual avatar's eye to provide more degrees of freedom in navigation. We can achieve this by making an avatar's head a trackable camera using a fiducial marker.

Finally, we would like to get more control of a virtual avatar using TUIs. The future tangible avatar should not be just a physical model, but a figure with flexible joints that can perform more complex tasks.

ACKNOWLEDGEMENTS

We thank Andy Echenique, Jonathan Massey, Shashank Chamoli, Szu-Chia Lu and the Synaesthetic Media Lab for helping shape our ideas. This work is supported by NSF-IIS grant #0705569 and a research collaboration grant from Steelcase Inc.

REFERENCES

1. Bowman, D. A., Koller, D. and Hodges, L. F., Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques, *Proc. VRAIS '97*, IEEE Computer Society (1997), 45-52.
2. Buxton, B., A Props-Based UI for Automotive Visualization on Large Format Powerwalls. Alias|Wavefront Research, 1999.
3. Buxton, W., Fitzmaurice, G., Balakrishnan, R. and Kurtenbach, G. Large Displays in Automotive Design. *IEEE Comput. Graph. Appl.*, 20, 4 (2000), 68-75.
4. Reilly, D., Volda, S., McKeon, M., Dantec, D. L., Bunde-Pedersen, J., Forslund, C., Verma, P., Edwards, W. K., Mynatt, E. D., Mazalek, A., Want, R., Space Matters: Physical-Digital and Physical-Virtual Codesign in inSpace, *IEEE Pervasive Computing vol. 9 no. 3*, (2010), 54-63.
5. Fitzmaurice, G. W., Ishii, H. and Buxton, W. A. S., Bricks: laying the foundations for graspable user interfaces, *Proc. CHI 95'*, ACM (1995), 442-449.
6. Frazer, J. H. *Three-Dimensional Data Input Devices*. National Academy of Sciences, 1982.
7. Frazer, J. H., Use of Simplified Three-Dimensional Computer Input Devices to Encourage Public Participation in Design, *Proc. of CAD '82*, (1982), 143-151.
8. Gillet, A., Sanner, M., Stoffler, D., Goodsell, D. and Olson, A., Augmented Reality with Tangible Auto-Fabricated Models for Molecular Biology Applications, *Proc. Vis. '04*, IEEE Computer Society (2004), 235-242.
9. Gorbet, M. G., Orth, M. and Ishii, H., Triangles: tangible interface for manipulation and exploration of digital information topography, *Proc CHI 98'*, ACM (1998), 49-56.
10. Hansson, P. R., Wallberg, A. and Simsarian, K., Techniques for "natural" interaction in multi-user CAVE-like environments, *ECSCW Poster*, (1997).
11. Ishii, H. and Ullmer, B., Tangible bits: towards seamless interfaces between people, bits and atoms, *Proc CHI 97'*, ACM (1997), 234-241.
12. Jacob, R. J. K. and Sibert, L. E., The perceptual structure of multidimensional input device selection, *Proc. CHI 92'*, ACM (1992), 211-218.
13. Kim, J.-S., Gračanin, D., Matković, K. and Quek, F., Finger Walking in Place (FWIP): A Traveling Technique in Virtual Environments, *Proc. ISSG 08'*, Springer-Verlag (2008), 58-69.
14. Kim, J.-S., Gračanin, D., Singh, H. L., Matković, K. and Juric, J., A Tangible User Interface System for CAVE Applicat, *Proc VR 06'*, IEEE Computer Society (2006), 261-264.
15. Kruszyński, K. J. and Liere, R. v. Tangible props for scientific visualization: concept, requirements, application. *Virtual Real.*, 13, 4 (2009), 235-244.
16. Mazalek, A. and Nitsche, M., Tangible interfaces for real-time 3D virtual environments, *Proc. ACE 07'*, ACM (2007), 155-162.
17. Mazalek, A., Winegarden, C., Al-Haddad, T., Robinson, S. J. and Wu, C.-S., Architalas: physical/digital co-design of an interactive story table, *Proc. TEI 09'*, ACM (2009), 241-248.
18. Metoyer, R. A., Xu, L. and Srinivasan, M., A Tangible Interface for High-Level Direction of Multiple Animated Characters, *Proc. Graphics Interface 03'*, (2003), 167-176.
19. Regenbrecht, H., Haller, M., Hauber, J. and Billingham, M. Carpeno: interfacing remote collaborative virtual environments with table-top interaction. *Virtual Real.*, 10, 2 (2006), 95-107.
20. Rekimoto, J. and Saitoh, M., Augmented surfaces: a spatially continuous work space for hybrid computing environments, *Proc. CHI 99'*, ACM (1999), 378-385.
21. Stacey, D. S. *Guest Editors' Introduction: Interacting with Digital Tabletops*. 2006.
22. Tang, J. C. and Minneman, S. L. Videodraw: a video interface for collaborative drawing. *ACM Trans. Inf. Syst.*, 9, 2 (1991), 170-184.
23. Ullmer, B. and Ishii, H. Emerging frameworks for tangible user interfaces. *IBM Syst. J.*, 39, 3-4 (2000), 915-931.
24. Yarosh, S., Cuzzort, S., Müller, H. and Abowd, G. D., Developing a media space for remote synchronous parent-child interaction, *Proc. IDC 2009*, ACM (2009), 97-105.