

# The 3D Tractus: A Three-Dimensional Drawing Board

Paul Lapidès  
paul.lapides@gmail.com

Ehud Sharlin

Mario Costa Sousa  
{ehud, mario, streitl}@cpsc.ucalgary.ca

Lisa Streit

Department of Computer Science, University of Calgary  
2500 University Drive NW, Calgary, Alberta, Canada, T2N 1N4  
+1.403.220.6015

## Abstract

*We present the 3D Tractus: a simple and inexpensive system for interaction and exploration of three-dimensional (3D) data. The device is based on a traditional drawing board-like mechanical structure that can be easily moved up and down while its surface height is being tracked using a simple sensor. Users interact with a tablet or tablet PC that rests on the surface while simultaneously changing its height. The result is direct mapping of virtual and physical spaces allowing intuitive 3D interaction and data exploration. The 3D Tractus allows us to investigate novel 3D interaction techniques based on sketching and drawing as well as intuitive visual indicators and GUI layouts. The 3D Tractus' simple design concept can be easily adapted to other tabletop systems and the simple nature of the physical interaction allows the design of various exciting applications. We detail here the design and development of the 3D Tractus hardware and software as well as preliminary evaluation of a 3D drawing and sketching application realized using the new tabletop interface.*



**Figure 1: Interaction with the 3D Tractus.**

## 1. Introduction

The 3D Tractus (Figure 1) is an intuitive three-dimensional (3D) interface based on the familiar drawing board as its interaction metaphor. The Tractus enhances the concept of a standard two-dimensional (2D) tabletop by allowing the user to easily adjust the interaction surface height. Having the ability to control the drawing surface height allows direct mapping between physical and virtual spaces, and enables the user to intuitively input and explore 3D data.

The Tractus is designed to afford direct interaction and full functionality using only a single pointing device, namely a stylus. Our current implementation supports drawing of 3D curves and shapes in real time and allows simple presentation, exploration and editing of the 3D content using an intuitive graphical user interface. The user is using both hands for interaction: one hand is used to adjust the Tractus height and the other hand for drawing or writing on the interaction surface.

Most current interfaces used today are based on the windows, icons, menus, and pointers (WIMP), and the common mouse and keyboard and while they are good for interacting with 2D data, they are cumbersome for 3D manipulation and interaction. Computer-aided design (CAD) programs attempt to preserve the ease of 2D interaction and context by forcing the user to do their work from only one side, instead of using a 3D context.

We are pursuing several applications for the Tractus. First we will present a 3D drawing application that allows artists to draw line art directly in 3D. With the ability to draw 3D curves, we plan to expand this application to allow users to create trees and plants. Other ideas include programs that explore 3D MRI data and allow physicians to interact and annotate their patients' scans, as well as tools that support other spatial interaction and control tasks.

The concept of the Tractus can also be seen as an inexpensive way to enhance other tabletop interfaces allowing them to explore, analyze and interact directly with 3D data.

## 2. Related Work

Traditional 2D input devices like mouse or tablet are still the principal interfaces for 3D interaction. Following, current methods of inputting 3D curves and lines using CAD software tools (like Maya®) are based on 2D concepts and require the use of top, side, front, and back views (viewports) of the object being modeled. This does not lend to an intuitive interface and forces the user to learn 3D visualization skills for proper interaction. 3D interfaces that can potentially offer intuitive solutions to these challenges changed little over the last decade [7] and are still expensive and rare outside of research labs.

### 2.1. 3D Curve Input

A new approach for inputting 3D curves while preserving a single 2D viewport was proposed by Cohen et al. [2]. The user draws the curve, as it would appear to him/her from their viewpoint, as well as the curve's shadow on the floor plane. The system determines the shape of the 3D curve based on the projection of the user drawn curve to a shadow from a virtual light source. The authors outlined two main problems: determining what a shadow should look like for complex 3D curves is not intuitive, and can be difficult even for users with artistic training.

Interaction of layered data may be done using a tray metaphor, as described by McGuffin et al. [11]. The tray is not tangible but instead a virtual object controlled by a mouse and a GUI. The user selects a slice of their 3D data and pulls it out of the stack so they can view and change it.

Kallio et al. [10] describes a tool that allows the user to deform and translate a 3D virtual surface. Curves are sketched on the screen using a mouse and projected onto the surface allowing the user to draw smooth lines. The surface can be transformed through a series of keystrokes, which may not be intuitive for simultaneous drawing and transformation.

Ijiri et al. [9] presents a tool for generating 3D curves used in plant creation with the goal of minimal effort. The user draws a single 2D curve with the mouse and the system attempts to generate a 3D plant stem that looks the same from any angle (around the vertical axis). For example, if a 2D sinusoid curve is drawn, the resulting 3D stem will look like a sinusoid from any angle. The end result is a 3D curve generated from a single 2D stroke that resembles an even and balanced plant stem. This type of interaction is not intended for 3D drawings since the system assumes that the curve is to be used as a plant stem, and not a general 3D curve.

### 2.2. 3D Drawing Using Physical Interfaces

3D space interaction dates to Sach's et al. [14] 3-Draw system proposed in the early 90s. Other free space devices used for virtual reality use six degrees of freedom (DOF) tracking of the stylus and drawing surface [1, 8, 13]. Users can view their 3D drawing with an immersive virtual reality environment. The requirement of a complete virtual reality environment, commonly including head mounted display (HMD), stereo shutter glasses, tethered trackers for the input stylus and user's head, etc., can be seen as a disadvantage due to the complexity of the equipment necessary. Arguably free space interaction may not be an intuitive drawing approach for most users and designers who are used to drawing on physical surfaces.

The CAT and Interaction Table ([5, 6], respectively) give the user 6 DOF using a touch sensitive surface that can swivel around its center. Both interfaces were designed for isotonic orientation control and isometric translation control, so physical pressure instead of actual movement is used for translation in the virtual world.

ArtNova and inTouch ([3, 4], respectively) use a haptic interface that allows the user to directly interact with virtual reality environments for drawing applications. Both tools use a PHANTOM haptic device [15] that provides limited force feedback to the user through a non-planar physical interface.

The Boom Chameleon [21] incorporates a touch sensitive display attached to an arm that is sensitive to position and movement. The user can move the display around to change the viewpoint, and to annotate and interact with the 3D scene. This arrangement gives direct mapping of virtual and physical spaces, but as far as we know it is not simple and inexpensive and hasn't been used for 3D drawing.

## 3. Designing the 3D Tractus

When designing the Tractus we had two distinct goals. First, to design a tool that allows the user to directly map virtual and physical spaces. This design feature would eliminate many counterintuitive design principles like zooming and moving around in virtual space. Secondly, the preservation of the traditional drawing board principle was the key to intuitive interaction. This is the hardware and software that we developed with both goals in mind.

The Tractus is designed as a physical table with adjustable height. To enable this, we designed a simple metal base with four legs and a table surface that also has four legs. The two sets of legs are aligned next to one another and attached together with plates that allow the legs to slide. The

prototype has a minimum height of 73cm and a maximum height of 113cm. A simple potentiometer-based sensor is used for measuring the top-surface height. This simple design can be adapted to tables of various sizes and heights.

For simplicity of control, the software to be used with the table relies on a single pointer. The user is presented with toolbar buttons that control common features to avoid menus.

The entire Tractus cost about \$300 CAD, excluding the tablet and tablet PC.

#### 4. Hardware Implementation

The most critical hardware requirement is that the user may freely move the surface up and down with only one hand. For this reason, the surface must be light and the tracks on which the surface moves on must be as frictionless as possible. The tracks must still resist torques that will be the result of lifting the surface by one edge, instead of by the center. To make the surface seem weightless and easy to move up and down, we designed a simple counterweight system. Because of this, the user doesn't need to pull up or push down with a lot of force, so there is less torque on the tracks.

##### 4.1. Frame and Counterweight

The first iteration of the hardware consisted of four vertical aluminum bars [19] that remain on the ground as the base of the 3D Tractus. Four more bars were added to the base for support, creating a square prism. A table surface made from plastic was attached to four other bars such that both

sets of legs would fit closely together and could be attached together with plates. The plates that held the surface to the base provided enough friction for the user to lightly support the surface while they interacted with the peripheral on top. At first the interactive surface we used was a Wacom Intuis 2 [21] tablet, which is very light. This forced separation of action and perception spaces since the user was inputting through the tablet while looking at the interaction results at a PC monitor [17]. This was improved by placing a Toshiba Portege M200 Tablet PC [20] on top of the 3D Tractus (Figure 2). This enabled merging of perception and action interaction spaces supporting intuitive interaction [17]. However, the weight of the tablet PC in addition to the surface became too heavy for the user to constantly support, let alone move up and down easily. In order to support the physical height adjustment action a simple counterweight system similar to an elevator was added to the 3D Tractus.

Typically an elevator will have the car and weight both hanging below a pulley, but applying this idea to the Tractus would require an elaborate support structure coming off the back. This would not be aesthetically pleasing, and would require a lot of materials. Instead, we attached the surface's support cable to the bottom of the surface's legs (Figure 3). The entire counterweight system is housed underneath the surface and hidden out of the way. Additionally, no extra support structures need to be built, since the pulley can be attached to the top of the base's frame.

We use two weight combinations: one for the lighter Intuis and another additional weight for the heavier Portege Tablet PC.



Figure 2: The 3D Tractus.



Figure 3: The counterweight system (3D Tractus in the upper position).

## 4.2. Height Sensor

A string potentiometer (POT) was used as a simple, inexpensive and precise sensor for tracking the height of the surface. A rotary potentiometer is attached to a string creating a variable voltage divider controlled by the distance the string is withdrawn. A 16-bit analog/digital converter [12] is connected to a computer via USB. The converter powers the potentiometer with +5VDC and samples the output voltage in a range of +0VDC and +5VDC. The length of the string is 75cm, but in the current Tractus setting only about 40cm will ever be withdrawn due to the maximum height the surface can be lifted to.

## 5. Software Implementation

The software that we have developed allows the user to draw sketches in 3D space using a tablet or tablet PC. We are currently using the Toshiba Portege Tablet PC to run the Tractus software. Here we will discuss general and specific concepts that relate to the Tractus and to the drawing application.

### 5.1. Pen Based Input

The Tractus is operated with two hands; one hand controlling the height of the interaction surface, the other hand drawing and interacting with the virtual content. Following, the software should be intuitive to use and handle with a single pointing device. The devices we use for the Tractus are a Wacom Intuis 2 [21] tablet and a Toshiba Portege Tablet PC [19], both of which are pen input devices. Other input devices such as the SmartTable [18] may be used, as long as the user can effectively interact with them using a single hand. Following other Tablet PC software, the drawing tools we have developed (Figure 4) requires no need for keyboard interaction and is intuitive to use, providing large buttons and easy to understand visual and contextual cues. Due to the variety of tablets and pens (as well as touch sensitive displays) available on the market today, we can only be guaranteed that the interface hardware will support at least one pointing interaction, based on at least one button in addition with sensing the stylus location (using the common mouse analogy, we are guaranteed the mouse pointing location as well as at least the left mouse click).

### 5.2. Drawing Widgets

Due to the expensive price of large tablet and touch sensitive displays, effective use of screen real estate was a key design consideration. The information presented must be large enough to be clearly seen and to accurately interact with. We designed the 3D Tractus drawing application so the user will always have visual access to three main

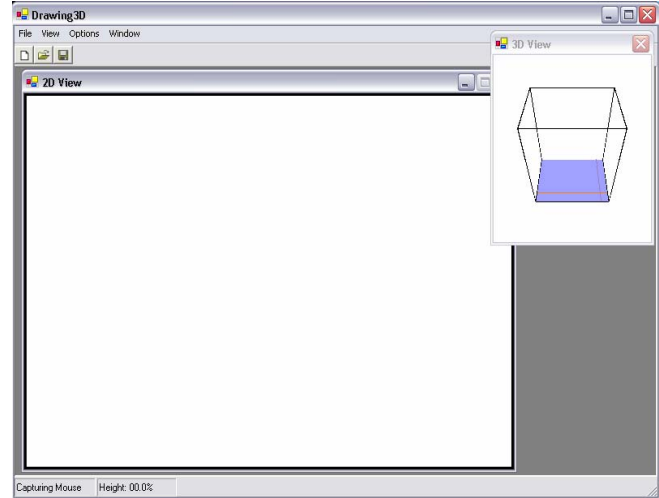


Figure 4: The 3D drawing software.

components: the sketch overview in 3D, the drawing pad area to input strokes, and a menu bar to access less frequently used features.

First, the users should always be able to see what they are drawing as if their sketch is laid directly in front of them. We propose a small overview window that may be resized and moved around the screen (Figure 4). The overview window always remains on top of all other windows since the user will frequently consult this window for a glance at what their sketch looks like. If the user finds that this window is interfering with the drawing process it may be minimized and maximized later. Often the user wishes to see what their 3D sketch looks like from different perspectives. A drag-based rotation is supported, where the user drags the stylus across the overview window and the perspective of the sketch changes. Dragging horizontally will rotate the sketch along the Y-axis, while dragging vertically will rotate it along the X-axis. This is a fast and simple interaction method that gives the user full viewing control over the sketch.

Second, the user must have an area dedicated to input the strokes that compose the sketch, like a drawing pad (Figure 4). Usually this drawing pad will make up the larger portion of the screen, although it can also be resized and minimized. This drawing pad window displays what the users would see if they were looking directly down on their sketch. We see the drawing pad as a window that enables the user to see into the virtual world of their sketch. Since the 3D Tractus does not use stereovision, the drawing pad supports the illusion of depth by implementing various visual and projection cues that attempt to intuitively convey the 3D information to the user. These visual and projection cues will be discussed in Section 5.3.

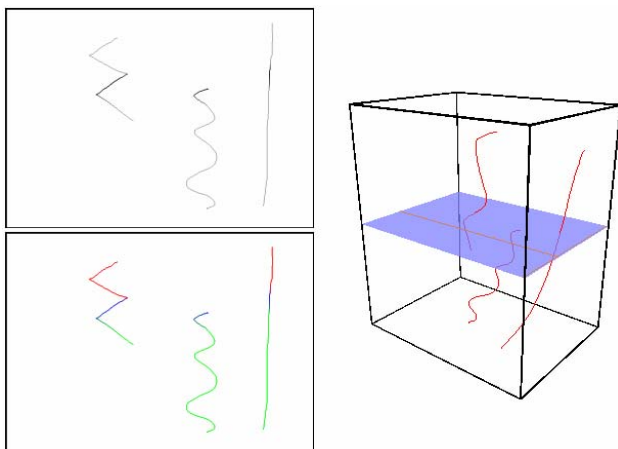
Lastly, the user should have control of Tractus general functions like system calibration, saving and loading of sketches, and exiting the application. A windows-type menu is displayed at the top concealing these features from the user while still allowing easy pointer based access.

### 5.3. Visual Cues

When using the drawing pad (Section 5.2), users can have a difficult time perceiving the direct spatial mapping of their work if they are unable to distinguish which strokes are far below them and which are close to their view. Our goal was to develop a simple depth cue that would quickly allow the user to perceive their work in 3D but would also be minimal and avoid cluttering the drawing pad display.

Integrating depth cues to the drawing pad becomes more challenging when the surface is in the middle position and parts of the sketch are above the interaction surface and others are below it. Since the user is physically looking down onto the 3D Tractus display, it makes sense to include in the drawing pad everything that is drawn below their viewpoint. Ideally the drawing pad should also integrate information of the sketch that is above the interaction surface and the user’s viewpoint.

We began by experimenting with the gray scale intensity of the lines that are part of the sketch (Figure 5). Lines that are far away from the surface would be displayed in a very light gray color, while lines that are in close proximity to the surface would be very dark gray or even black. We found that it was difficult to distinguish lines that were above the surface with lines that were below the surface. Further, we could not determine quickly the relative distance of the lines from the interaction surface. Due to lack of contrast between the black portions and the darker gray portions of the lines, it was difficult to accurately assess at which point the 2D interaction surface was intersecting the sketched 3D lines.



**Figure 5: Gray scale (top) and color (bottom) depth cues, and the overall sketch (right).**

Instead of a pure gray scale approach, we also added color (Figure 5). Parts of the line that were above the surface would be red, parts that were below would be green, and the region of the line in close proximity to the surface would be blue. These colors would still fade as the line got further away from the surface. While information was presented to show which lines were above and below, it was not a natural or intuitive cue. The user would constantly have to recall that red meant “above” and green meant “below”, taking their focus away from their work.

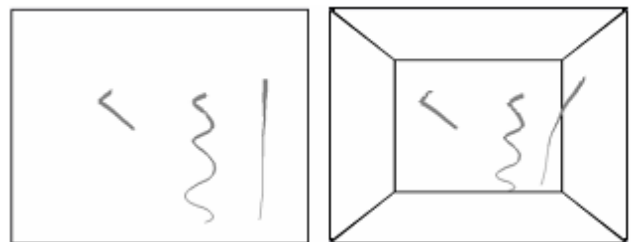
In both cases, we observed that color intensity did not have the necessary contrast to present distance and proximity accurately or intuitively.

Our drawing pad is currently implemented using dynamic line width for the depth cues, where lines that are far are thin and lines that are close are thick (Figure 6). This follows a simple vision concept where near objects are bigger than far objects. We were also inspired by a traditional illustration technique that modulates line width [23]. In traditional cases, the width of the line is a function of the pressure of the pen, but in our application it depends on the vertical distance from the drawing surface.

The line width depth cue solution was much more effective in presenting proximity than the color fading techniques, but it still falls short of showing the lines above the surface without distracting the user. Currently our implementation simply shows nothing above the interaction surface, enhancing the clarity of the information being displayed. This approach neglects important information that can be viewed only using the overview window, or by raising the Tractus to include more volume in the current drawing pad display. Being guaranteed that any lines shown are below the surface means that the user does not have to consciously make the distinction between lines that are above and below.

### 5.4. Projection Cues

For the drawing pad, we considered and implemented both an orthographic projection and a perspective projection to present the sketch to the user, eventually selecting the perspective projection (Figure 6).



**Figure 6: Orthographic projection (left) and perspective projection (right) cues.**

We implemented the orthographic projection in the drawing pad to try and provide a different view of the 3D drawing, since the user is already given a perspective projection view of the sketch (in the overview window; Section 5.2). Also, the users may wish to align their sketching or have a purely top-down view of their sketch, both of which are benefits of an orthographic projection. The problem with this approach is that the user does not have an intuitive feel of the volume of the 3D sketch. The orthographic projection does not display any context information for height, so the user is forced to try and visualize how high the surface is and the bounds of the volume that he/she must work in. Switching to our current drawing pad approach, a perspective projection (i.e. with a vanishing point), the user immediately sees the bottom, sides and edges of his/her work volume. Also, the perspective projection simulates an actual viewpoint that the user looks at the sketch from. An orthographic projection has benefits that may be applied to other applications where there is a real need for alignment, but a perspective projection currently seems to us to be the best solution for a 3D drawing application that will be used mostly by artists.

### 5.5. Deletion

Occasionally users make mistakes while drawing and need to erase some of the strokes they have drawn. The concept of erasing is widely used and different implementations have been developed and used in 2D drawing tools. The

most popular erase tool (as in Adobe PhotoShop®) behaves like a thick pen, removing any ink underneath it. Another approach will be to emulate a real eraser by allowing the user to scribble overtop of lines they would like to remove. For both erasing tools, when used with a 2D image, it is easy to see what is being erased because intersecting the erase tool with the target is straightforward using a 2D interface. This may not be the case for a 3D eraser tool, especially if the user is erasing lines, since being forced to trace along 3D line may be difficult to do repeatedly. Instead of a 3D eraser, we implemented a traditional 2D eraser and treated the drawing pad as a 2D image while erasing. Furthermore, we allow the user to erase entire strokes instead of small portions. Since strokes are intended to be a rough visualization and not a precise curve, the user can reproduce variations of deleted strokes quickly.

### 5.6. Calibration

The Tractus software should be aware of three things for accurate measurement of the surface height: minimum, maximum, and the current voltage readings being supplied from the converter. The calibration tool should allow the user to capture minimum and maximum height readings from the string POT. Since the calibration values rarely change over time, the software stores these values after calibration, and the process is not repeated every time the application is launched.

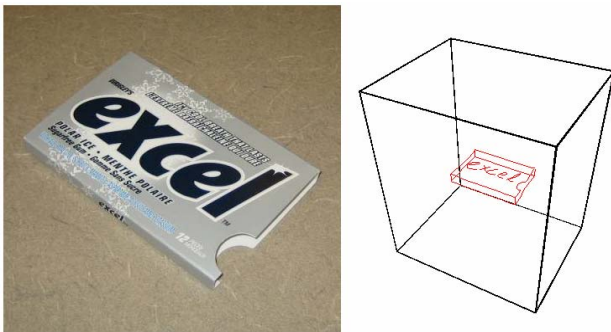


Figure 7: User created gum package.

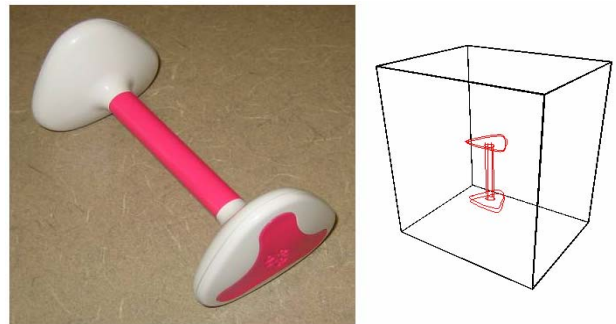


Figure 8: User created Aibo® Bone.

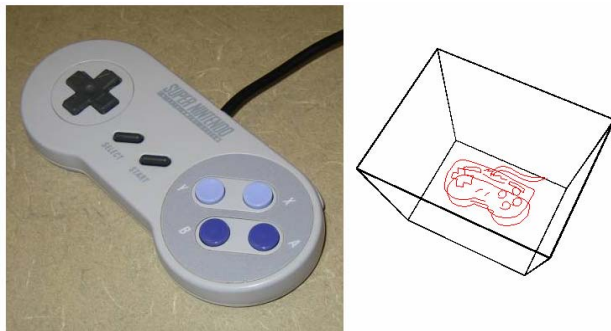


Figure 9: User created Nintendo® game controller.

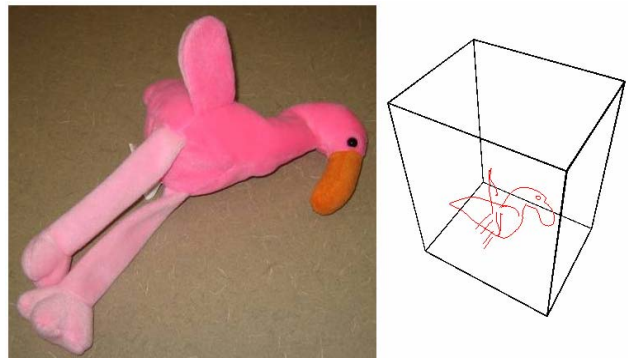


Figure 10: User created stuffed animal.



**Figure 11: Enlarged user created strokes (refer to Figures 7 - 10).**

## 6. Evaluation

We performed a small-scale evaluation where we asked three students with art backgrounds to use the Tractus for about 30 minutes each to draw simple shapes (Figures 7, 8, 9, and 10) and to give their opinions of the interface.

Users quickly grasped the concept of direct mapping between the virtual and physical spaces and the idea of moving the table up and down while drawing. A common remark was that it was easier to move the table down while drawing instead of moving it up. When users were moving the Tractus down, both hands were pushing down instead of one hand pulling the Tractus up and one hand pushing down the stylus. Users said they enjoyed being able to view their sketch from different angles using the overview window but wanted to be able to tilt the surface in a similar fashion.

When drawing objects with aligned characteristics like boxes, users said that they had a difficult time making the top and bottom of the box (in this example) the same size. This lends a perfect use for the already developed orthographic projection (Section 5.3). Allowing users to quickly switch between orthographic and perspective modes would enable easy aligning only when required, with the majority of the time spent using the perspective mode.

## 7. Conclusion

We present the 3D Tractus, a simple and intuitive evolution of the traditional drawing board that allows direct spatial mapping of virtual and physical spaces in three dimensions. The 3D Tractus affords interaction and exploration of 3D data using classic drawing and sketching techniques. We have developed hardware that is inexpensive, simple to replicate and easy to use and software that allows the user to draw in 3D. The user handles the 3D Tractus with both hands, one hand adjusts the interaction surface height and the other holds a single pointing device for input. The 3D Tractus software has an easy to use interface, is customizable, and provides intuitive visual cues.

The Tractus is a design concept that can be applied to any tabletop interface. All that is required is a mechanism to allow vertical movement of the table and a sensor. Vertical interaction can add richness to already developed table applications.

## 8. Future Work

While our design restricts the tilting of the physical surface, allowing the user to tilt the virtual surface is an important feature. We have found that drawing lines at a very low gradient with respect to the physical horizontal plane is difficult to do due to the minuscule changes necessary in the height of the surface, which is hard to control with just one hand. Instead of forcing users to create their curves this way, we propose a mechanism that allows the user to rotate the entire virtual space, changing the gradient the curve is drawn in to a steeper and easier to control one. Once the user completes the drawing, they would either rotate the virtual space back to normal or rely on the computer to perfectly align it for them.

We are planning to further study and improve the ergonomics of the Tractus. We will also evaluate the usability of different interaction techniques in a comparative study. We will allow artists to choose between other input and visualization techniques to discover which are preferred when drawing with the Tractus. Such differences include having a separate input window for more complicated and cluttered sketches and different depth cue visualization.

There are many applications the Tractus is suited for. We would like to see the Tractus support various object representations, such as polygon meshes and volume data. The Tractus would also be well suited for exploration of spatial data. Due to the vertical slice nature of MRI pictures, physicians can view and move around inside of the MRI scans of their patients. Physicians would also be able to highlight regions and organs that need treatment and to add their notes and diagnoses so colleagues or students can easily share information. Other spatial interaction applications would allow the user to control objects in space by selecting and assigning wayfinding landmarks or instructions.

## Acknowledgments

We thank the anonymous reviewers for their valuable comments and suggestions. We would also like to thank Brian Scowcroft, Ryan Schmidt, Annie Tat and Joseph Cherlin from the Computer Science Department at the University of Calgary for their support and help. This work was partially funded by the Canadian Natural Sciences and Engineering Research Council (NSERC) and a University of Calgary startup grant.

## References

1. M. Billinghurst, S. Baldis, L. Matheson, and M. Philips, "3D palette: a virtual reality content creation tool". *Proc. Virtual Reality Software and Technology*, ACM Press, Lausanne, Switzerland, 1997: pp 155-156.
2. J.M. Cohen, L. Markosian, R.C. Zeleznik, J.F. Hughes, and R. Barzel, "An Interface for Sketching 3D Curves", *Proc. Symposium on Interactive 3D Graphics*, ACM Press, Atlanta, Georgia, United States, 1999: pp 17-21.
3. M. Foskey, M.A. Otaduy, and M.C. Lin, "ArtNova: Touch-Enabled 3D Model Design", *Proc. IEEE Virtual Reality Conference 2002*, IEEE Press, Orlando, Florida, United States, 2002, pp 119-126.
4. A.D. Gregory, S.A. Ehmman, and M.C. Lin, "inTouch: Interactive Multiresolution Modeling and 3D Painting with a Haptic Interface", *Proc. IEEE Virtual Reality Conference 2000*, IEEE Press, New Brunswick, New Jersey, United States, 2000, pp 45-54.
5. M. Hachet, P. Guitton, P. Reuter, and F. Tyndiuk, "The CAT for Efficient 2D and 3D Interaction as an Alternative to Mouse Adaptations", *Proc. Virtual Reality Software and Technology*, ACM Press, Osaka, Japan, 2003, pp 205-212.
6. M. Hachet, and P. Guitton, "The Interaction Table - a New Input Device Designed for Interaction in Immersive Large Display Environments", *Proc. Eighth Eurographics Workshop on Virtual Environments*, Blackwell Publishers, Saarbrucken, Germany, 2002, pp 189-196.
7. K. Herndon, A.V. Dam, and M. Gleicher, "The Challenges of 3D Interaction", *ACM SIGCHI Bulletin*, ACM Press, 1994, pp 36-43.
8. K. Hinckley, R. Pausch, J. Goble, and N. Kassell, "Passive Real-World Interface Props for Neurosurgical Visualization", *Proc. CHI 1994*, ACM Press, Boston, Massachusetts, United States, 1994, 452-458.
9. T. Ijiri, M. Okabe, S. Owada, and T. Igarashi, "Floral diagrams and inflorescences: Interactive flower modeling using botanical structural constraints", *Proc. ACM SIGGRAPH 2005*, ACM Press, Los Angeles, California, United States, 2005, pp 720-726.
10. K. Kallio, "3D6B Editor: Projective 3D Sketching with Line-Based Rendering", *2nd Eurographics Workshop on Sketch-Based Interfaces and Modeling*, Blackwell Publishers, Dublin, Ireland, 2005, pp 73-80.
11. M.J. McGuffin, L. Tancau, and R. Balakrishnan, "Using Deformations for Browsing Volumetric Data", *Proc. IEEE Visualization 2003*, IEEE Press, Seattle, Washington, United States, 2003, pp 401-408.
12. ADU-100®, *Ontrak*, [www.ontrak.net](http://www.ontrak.net)
13. L. Poupyrev, N. Tomokazu, and S. Weghorst, "Virtual Notepad: Handwriting in Immersive VR", *Proc. Virtual Reality Annual International Symposium '98*, IEEE Press, Atlanta, Georgia, United States, 1998, pp 126-132.
14. E. Sachs, A. Roberts, and D. Stoops, "3-Draw: A tool for designing 3D shapes", *Proc. IEEE Computer Graphics and Applications*, IEEE Computer Society Press, 1991, pp 18-26.
15. PHANTOM Arm®, *SensAble*, [www.sensable.com](http://www.sensable.com)
16. E. Sharlin, M.C. Sousa, "Drawing in Space using the 3D Tractus", *Proc. CHI 2005 Extended Abstracts*, ACM Press, Portland, Oregon, United States, 2005, pp 2215.
17. E. Sharlin, B.A. Watson, Y. Kitamura, F. Kishino, and Y. Itoh, "On Tangible User Interfaces, Humans and Spatiality", *Personal and Ubiquitous Computing*, Springer-Verlag, 2004, pp 338-346.
18. SmartTable®, *Smart Technologies*, [www.smarttech.com/](http://www.smarttech.com/)
19. The Industrial Erector Set®, *80/20*, [www.8020.net/](http://www.8020.net/)
20. Toshiba Portege®, *Toshiba*, [www.toshiba.com](http://www.toshiba.com)
21. M. Tsang, G.W. Fitzmaurice, G. Kurtenbach, A. Khan, and B. Buxton, "Boom Chameleon: Simultaneous capture of 3D viewpoint, voice and gesture annotations on a spatially-aware display". *Proc. Symposium on User Interface Software and Technology*, ACM Press, Paris, France, 2002, pp 111-120.
22. Intuos®, *Wacom*, [www.wacom.com](http://www.wacom.com)
23. W. Crane, *Line and Form*, George Bell Sons, London, 1900.