

# 3D Bridge Construction

Author: Yuan Xie

Mentor: Dr. Peter Schröder and Steven Schkolne  
*California Institute of Technology*

## Abstract

In the field of architecture, tangible virtual reality interfaces allow architects to design and construct large complex structures in a three-dimensional space, and interact with the 3D models using the most natural means of computer human interaction: the two-handed system. The main goal of this project is to design and create an interactive, tangible, virtual reality interface for constructing various kinds of bridges. This project involves constructing basic elements of a bridge and integrating these elements with the help of the existing tangible tools so that the user can conveniently interact with the building elements and design a broad variety of bridge structures by using the provided primitives. The user is provided with some physical tools that are often seen in our life to interact with the digital objects. He can use a tong to grab and therefore select any bridge component and move it, a handle to place a copy of a selected component in space, and a ray gun to create cables in a bridge or change the shape of the cables. All of the bridge elements and functions of the tools are built using the principle of object-oriented programming.

## Introduction

Computers have been major tools for designers and engineers from all kinds of industries to construct models of complex objects and interact with these models to predict the quality of their designs. As chip designers in Silicon Valley are making painstaking efforts to develop faster but cheaper chips to maximize the utility of computers, traditional computer interfaces can no longer satisfy the great demand for a more convenient user-interactive interface mechanism. With the 30 years old Keyboard-Mouse-Monitor interface and the

Window-Icon-Menu-Pointer interaction metaphor, the designers have little freedom to play with these objects in different angles therefore creativity is seriously limited [8]. What the users really need for object modeling is an interface with which they can freely construct objects in space and interact with these constructed visual objects in the same way as they interact with real objects. Instead, tangible virtual reality interfaces for creating three-dimensional shapes offer a natural means of human computer interaction and provide the users a far more intuitive way than the traditional two-dimensional interfaces in designing and assembling complex objects. A tangible interface is one that allows the user to use some physical tools to grasp and manipulate digital information. Via the use of physical tools such as tongs and ray gun, the motions of the user's two hands become the input of the computer program [2]. The idea of tangible user interfaces first originated from Hiroshi Ishii from the Tangible Media Group at the MIT Media Lab. He and his team established their own "vision of Human Computer Interaction (HCI): 'Tangible Bits'" and are seeking to "change the 'painted bits' of graphical user interface into 'tangible bits' by giving physical form to digital information." [2]. Tangible user interfaces have many applications in various industries. In their paper "Evolution of the talking dinosaur: the (not so) natural history of a new interface for children", Kristin Alexander and Erik Strommen from Microsoft present an example of tangible interactive tools as toys for children, ActiMates Barney: a 13" animated doll that can interact with children via touch sensors in its hands and feet and a light sensor in its left eye [1]. In the field of education, the invention of AlgoBlock allows students to compose computer programs without understanding the complex

programming language syntax, instead using some physical blocks connected with computers as programming input [3]. In Biology, tangible interfaces can be used to navigate a 3D model of the human body to serve research purposes.



Figure 1: Various examples of tangible user interface.  
 Top: the interface doll ActiMates Barney that can function as a playmate for children [1].  
 Bottom: the new learning tool called ArgoBlock which enables the students to learn how to construct computer programs without knowing any programming language [3].

Dr. Schröder and his research group in the Computer Science Department at Caltech have recently deployed a semi-immersive head-tracked stereoscopic display, which can be used as a virtual reality interface for spatial construction. In this virtual interactive environment, the physical motions of the user are mapped to interface commands through tangible props, and objects can be constructed and manipulated directly in 3D space [5]. The physical tools built in the existing interface in the lab, indicated in Figure 2, include a tong, a ray gun, a light saber and a glove.

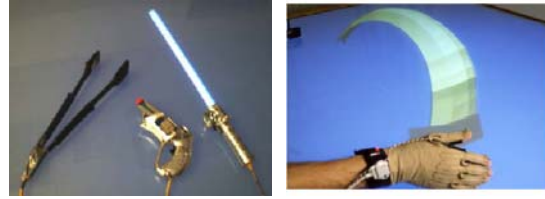


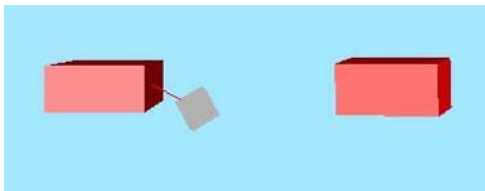
Figure 2: The tools used in our interface, including a tong, a ray gun and a light saber, are illustrated. The glove is a part of the existing interface in the lab that can be used to draw any free shape in space. [7]

- Left: the black tool on the left most is a tong, which is used for grabbing and moving objects freely in space. The ray gun is the one in the middle, which is mainly used for binding objects together. And the light saber on the right most is used for breaking these bonds [7].
- Right: illustrates the use of a glove to enable the user to draw any shape of his choice in space [7]

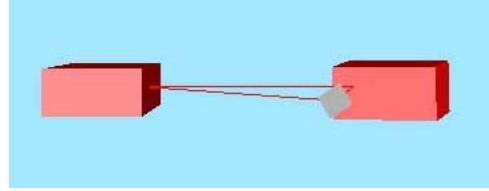
The current work in the lab has been focused on developing a 3D interface for spatial construction for DNA molecules and complexes. The idea of applying the tangible virtual reality interfaces to architectures has just been developed recently and no such interface has been developed before in this lab. The scope of my project was to design and create a similar tangible virtual reality 3D interface for constructing complex architectural structures, focusing on various kinds of bridges. The interface is called 3D Bridge Construction Interface and is based on the existing interface in the lab. The main objective of this project was to construct basic elements of a bridge and integrate the tools and objects, so that the user can conveniently interact with them and design a broad variety of complex bridge structures by using the provided primitives. The final interface enables the user to create mainly three kinds of bridge: cable-stayed bridges, beam bridges, and suspension bridges. The user can easily perform various types of adjustments and modifications on the resulting bridge structure and view the bridge from any angle to have a much better understanding of their design. It provides the architecture designers a much more intuitive way to interact with the bridge models and more flexibility to make changes in their bridge models than the traditional 2D interface.

### The Interface

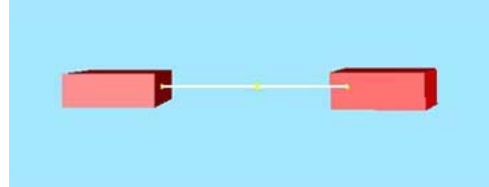
The 3D Bridge Construction Interface provides the user a variety of common primitives of bridges, including pier, beam, anchor, cable, main tower, etc. The user is able to use the tangible tools to select their desired components to design and build bridges. These tangible tools used in the 3D Bridge Construction Interface are those described above: two tongs, a glue gun and a light saber. However, additional functions are added to these physical tools for a better adaptation to their use in the construction of bridges. Most of the solid bridge components, such as beam, pier, anchor, main tower, can be placed in space by allowing the user to drop copies of these components with the light saber. As a result, the light saber becomes a bridge dropper handle. The bridge dropper handle, when triggered by pressing a menu button on the handle, displays a menu of various bridge components and allows the user to select their desired component. After a selection is made, although not physically shown, the handle is then attached to the selected bridge component. Pressing the main button on the handle enables the user to place a copy of that component in space. Other string type components, such as the main cable and supporting cables of a cable-stayed bridge or suspension bridge, can be drawn in space with a ray gun. When the user presses the main button of the gun, a ray comes out of the gun. The user can drag the tip of the ray between two other components of a bridge, such as two anchors, therefore draw the cable from its start position to its ending position. Figure 3 shows how a straight cable is formed.



Step 1: The user uses the tip of the ray to specify the starting position of the cable.



Step 2: The user drags the tip of the ray to the desired ending position of the cable



Step 3: A cable is then built with its specified starting and ending positions.

Figure 3: A cable between two anchors is formed by dragging the tip of a ray from its starting position to its ending position.

After a component is placed in space, the user can use the available tools to edit the properties of the component. Tongs enable the user to move a single component or a group of components in space, or change the dimensions of each component. The trigger button on each tong displays its various functions which the user wants to use. If the user selects the mover function, using a tong to grab any component close to the tong attaches that component to the tong, then moving the tong to any desired position moves the attached component to the same position as well. If a number of components are grouped together, using the tong in the same way as moving a single component results in moving the whole group of components. The following figure shows how an object is moved using the Tong.



Figure 4: Use a tong to grab an object and move it in space [7].

If the user selects the stretcher function, using

the tong to grab its nearest bridge component in space results in grabbing the part of a component to which the tong is closest, and moving the tong changes the corresponding dimension of that component. For example, placing the tong on the top of a pier, pressing the tong and moving it upward increases the height of the pier. The following graph illustrates how the stretcher works.

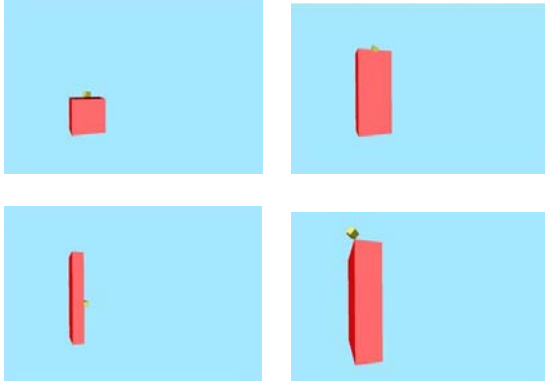


Figure 5: How a brick is stretched to become taller, thinner and wider

The tong can also be used to change the shape of a cable. Using the tong to grab a point on the cable and moving the tong downwards results in sagging of the cable, therefore illustrating the elastic bendy nature of the cable. Figure 6 shows the process of sagging.

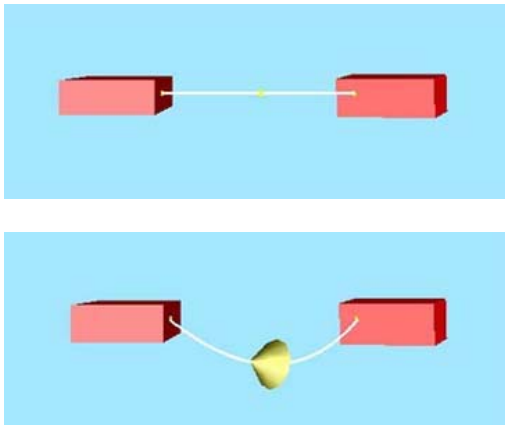


Figure 6: The ray gun can grab the center point of the cable and sag it.

In order to make sure some components, such as the beam and its supporting anchors, are orthogonal to each other as they always

are in the real world, certain constraints are enforced while the user places different bridge components in space. While the user is placing the components, despite the way she puts it, the beam will automatically be adjusted to horizontal orientation and the anchors will be adjusted to vertical orientation. Enforcing these constraints makes the final bridge structure more like real-world model.

Once all of the components are placed at the right positions with respect to each other and all of the properties are modified as the user desires, a bridge is completed. Figure 7 shows two examples of bridge models constructed using the 3D Bridge Construction Interface.

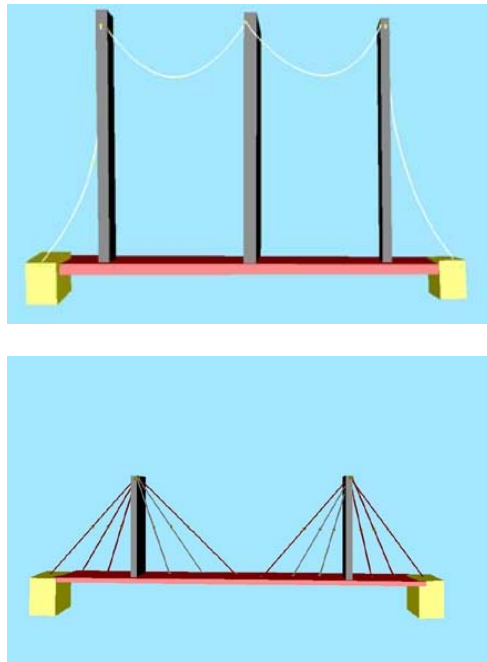


Figure 7: Examples of visual bridge models constructed using 3D Bridge Construction Interface. The top one is a suspension bridge, the bottom one is a cable-stayed bridge.

The user can use the tong to move the whole bridge into any angle or direction. She can bring the bridge into a closer or farther view, view it from the top or bottom, or investigate it from different sides. Based on a more detailed examination from various angles, the users can make further modifications and perfect her design.

### Benefits of the Interface

The main benefit of the 3D Bridge Construction Interface is that it provides more flexibility to make changes than the traditional 2D interface. Apparently making a change to a 3D structure if all you have is a 2D drawing is much harder than working with a change on the 3D structure in 3D itself. Moreover, since the interface provides such an intuitive way for the designers to interact with the object, they can just grab a bridge component using a tong, or shoot a cable using a ray gun, and therefore do not need to deal with all the complicated procedures of learning and using 2D graphics software. Thus, spatial construction allows the designers to face fewer constraints in the process of designing and provides more room for creativity while also minimizing the cost.

Our 3D Bridge Construction Interface explores the application of tangible virtual reality interfaces in architecture. In the future, one possible extension of this project is to build the forces between the internal parts of the bridge based on the physical properties and constraints of each component. We can plant it on the “ground” and bring it into the interaction with all kinds of outside environments, such as air and water. That is, based on its physical relation with its environments, we can add the forces between the bridge and the ground, the air, the water and other possible objects that might be interacting with this bridge. Another possible extension, which is also what we are currently working on right now, is to create a 3D interface for general polygon modeling. The interface is mainly used for the user to model any free-form object by starting from a simple polygon.

## **Methods**

*Interactive Programming:* a model of computation that involves frequent interaction between the user and the program in a dynamic environment. In our interface, the user gives commands to the program via motions of hands and available tools. The interface immediately maps these motions into digital form and calculation is automatically performed using the received digital information. Then corresponding

geometry and objects are displayed in the computed position in space. We use the principle of interactive programming to implement our interface.

*Object-Oriented Programming and Polymorphism:* Except for the physical tools described above, all other tools and objects are in digital form, and therefore need to be programmed into the computer. All of the objects are implemented in an object-oriented way since this makes adding more components and tools for constructing the bridge in the future much easier. All of the bridge components inherit from one abstract base class called BridgeComponent. Polymorphism is also utilized in creation of these various bridge components since it allows us to maximize the usage of the codes and allow the users to design and implement their own components on top of the primitives we provide. All of the software tools for drawing and editing the components, inherit from the already existing class Softool, which is the base class for all functions of all physical tools.

*Matrix Computation and Quaternion:* Another important method used in the implementation of our interface is matrix computation. Since all of our objects are composed of points, our program requires large amount of manipulations with the positions of points. Matrix is used to transform the position of points in 3D, represented in form of a vector  $(x, y, z)$ . There are three ways a matrix can transform a point in space, correspondingly there are three different kinds of transformation matrices, each of which cover a specific transformation. Translation matrix moves a point a certain distance relative to its origin. Scaling matrix scales a point by scaling its distance to the origin of the world coordinate. Rotation matrix rotates a set of points within a coordinate system.

Quaternion is a special 4x1 matrix defined using four floating point values  $(qx, qy, qz, qw)$ . Quaternion allows the programmer to rotate an object through a single arbitrary rotation axis. This solves the problems caused by rotating an object through a series of

successive rotations, which is the way a rotation matrix rotates an object, since the order of these sequences of rotations is usually not specified and different orders of rotations have different effects. Quaternion is an important method in our implementation since most of the rotations made while the users interact with objects in our interface are along some arbitrary axes. Given the rotation axis and angle, the following algorithm may be used to generate a quaternion:

```

sin_a=sin(angle/2);  cos_a = cos(angle/2)
qx = axis -> x * sin_a
qy = axis -> y * sin_a
qz = axis -> z * sin_a
qw = cos_a

```

Reversely, a quaternion can also be converted back to a rotation axis and angle. Moreover, a quaternion and a rotation matrix can be converted to each other as well. Quaternion is a very convenient method and concept in dealing with object rotations in 3D space.

*Lagrange Polynomials:* An algorithm utilized to bend the cable by changing the position of a point always interpolated by the cable. Lagrange polynomials give the updated positions of all points on a curve while the position of one point is changed [4].

## References

- 1 Alexander K., Strommen E. 1998. "Evolution of the Talking Dinosaur: the (not so) Natural History of a New Interface for Children". ACM ISBN 1-58113-028-7
- 2 Ishii H., Ullmer B. 1997. "Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms". ACM Press, pp. 234-241.
- 3 Kahn K. 1996. "Drawings On Napkins, Video-Game Animation, and Other Ways to Program Computers". Communication of the ACM, Vol. 39, No. 8
- 4 Rockwood A., Chambers P., 1996. Interactive Curves and Surfaces <<http://www.people.nnov.ru/fractal/Splines/Lagrange.htm>> Accessed September 27, 2003

- 5 Schkolne S., Schroder P. 2002. "Assembling and Rearranging Digital Objects in Physical Space with Tongs, a Gluegun, and a Lightsaber". <<http://www.cs.caltech.edu/~ss/pubs/schkolnelightsaber.pdf>> Accessed September 27, 2003.
- 6 Schkolne S., Schroder P. 2001. "Tracing the Line of Thought on the Responsive Workbench," <<http://www.cs.caltech.edu/~ss/pubs/schkolnecast01.pdf>> Accessed September 27, 2003.
- 7 Schkolne S., Pruett M. and Schröder P. 2001. "Surface Drawing: Creating Organic 3D Shapes with the Hand and Tangible Tools" <<http://www.cs.caltech.edu/~ss/pubs/schkolnechi01.pdf>> Accessed September 27, 2003.
- 8 Sutphen S., Sharlin E., Watson B., Frazier J. 2000. "Reviving a tangible interface affording 3D spatial interaction" <<http://www.cs.northwestern.edu/~watsonb/school/docs/ski00.segal.pdf>> Accessed September 27, 2003

## Acknowledgements

I thank Professor Peter Schröder for his mentorship, guidance and support, Steven Schkolne for constant guidance and assistance, and the Caltech SURF program for granting me the opportunity to participate in this research.