Multi-Fingered Interface for Sculpting of Generalized Cylinders

Tomer Moscovich
tm@cs.brown.edu

Olga A. Karpenko
koa@cs.brown.edu

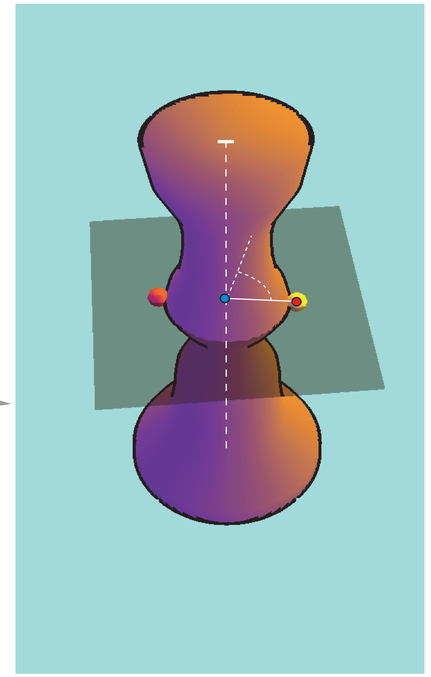
BROWN

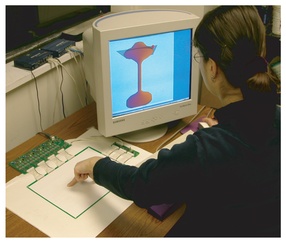
Goal

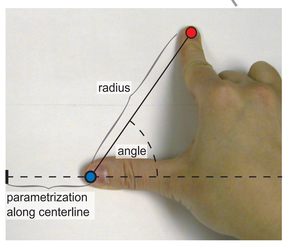
The advent of multi-touch input technology [1] provides inspiration for more expressive user interfaces. Our aim is to explore multi-fingered interfaces for 3D free-form modeling. With several fingers a user can control many shape parameters simultaneously, which allows for a clay-like sculpting interface. This project is our initial attempt to create such an interface for a limited class of 3D shapes - generalized cylinders.

Interaction

A shape in our system is defined by a set of cross-sections perpendicular to a 3D curve. A cross-section at position *t* along this curve is defined by a local coordinate frame and a list of points specified in polar coordinates within this frame. As shown below, one finger controls the parameter *t* to specify the modified cross-section, while the other controls the angle along which the radius should change, as well as that radius. A semi-transparent plane and two cursors are displayed on the screen to show the selected cross-section and the positions of the user's fingers.

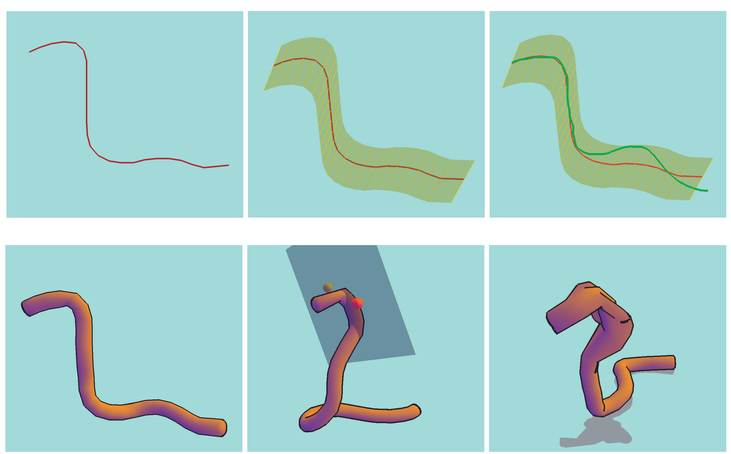


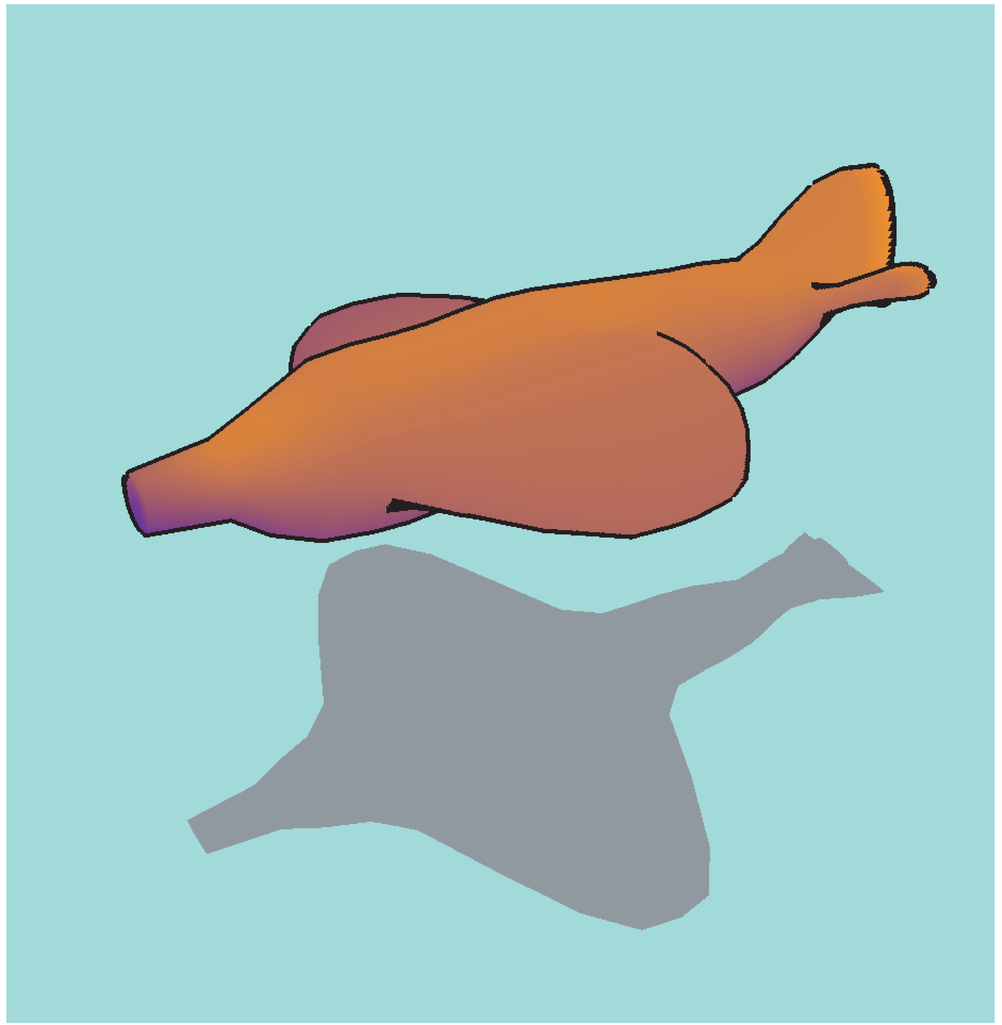




The user specifies the backbone curve of the shape using multi-view 3D curve sketching with epipolar constraints [2]. The user first draws the curve from one view, then moves the camera and draws the curve from a new view. These two projections define the curve in 3D space.

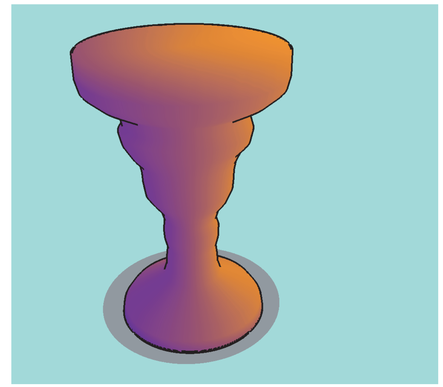
The system creates a generalized cylinder of uniform radius around the backbone curve. The user then begins to deform the shape by adjusting cross-sections as described above.

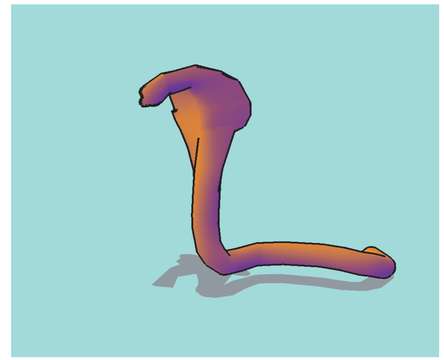




Results

The vase is created using the surface of revolution mode. The snake is created using the symmetry mode. The model of the plane is created with all three techniques, including single radius modification (plane's tail).





Implementation

When the user changes the radius at a specified angle of a cross-section, we must interpolate the radii of the adjacent angles to create a smooth fall-off within the cross-section. We also propagate the change to the neighboring cross-sections to ensure a smooth surface. We use a quadratic B-spline basis to approximate the values at the neighbors; this basis has the nice property of summing to unity.

Our system supports three modes: the first updates the radius at the specified theta, the second is a symmetry mode that changes the radii along both the angle and its conjugate, and the third allows the creation of surfaces of revolution.

References

[1] Rekimoto, J., *SmartSkin: an infrastructure for freehand manipulation on interactive surfaces.* SIGCHI 2002.

[2] Karpenko, O., Hughes, J. F., and Raskar, R. *Epipolar methods for multi-view sketching.* In Eurographics Workshop on Sketch-Based Interfaces 2004.

[3] Grimm, C., *Implicit generalized cylinders using profile curves.* In Implicit Surfaces '99.

[4] Igarashi, T., Matsuoka, S., and Tanaka, H. *Teddy: a sketching interface for 3d freeform design*. SIGGRAPH '99.