3D Warp Brush: Interactive Free-Form Modeling on the Responsive Workbench

Yong Joo Kil* UC Davis
Pietro Renzulli† Politecnico di Bari
Oliver Kreylos* UC Davis
Bernd Hamann* UC Davis
Giuseppe Monno‡ Politecnico di Bari
Oliver Staadt* UC Davis

Abstract

We introduce the 3D warp brush, a method for interactive shape modeling in a immersive virtual reality environment. 3D warp brushes are implicitly-defined tools that operate on triangle meshes. We combine the efficiency of explicit mesh representations with implicit modeling operators. The area of influence of a 3D warp brush can be of arbitrary shape since it has an associated distance field. We define different warp functions including drag, explode, and whittle. A unique feature of our framework is the ability to convert meshes into 3D warp brushes at run time. The use of a Responsive Workbench and two-handed interaction allows the user to exploit the full potential of the modeling system by intuitive and easy modification of a base surface into a desired shape. We present models, which have been created and modified using 3D warp brushes, to demonstrate the usefulness of our framework.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, Surface, Solid, and Object Representations

Keywords: free-form modeling, human-computer interaction

1 Introduction

Traditional 3D-surface modeling methods – including parametric surface representations such as Bézier and NURBS surfaces – achieve high-quality results, but require manipulation of control meshes to influence surface shape. This indirect manipulation metaphor requires the designer to change control vertices using either a mouse or a tablet. Even though input devices with higher degrees of freedom have become more widely available, it is unclear how they can be successfully utilized in such settings.

We propose a hybrid free-form modeling framework that uses a VR interface and combines implicitly-defined tools – 3D warp brushes – that operate on adaptively-refined triangle meshes. We present data structures and algorithms that support 3D warp brushes with arbitrary shape and a variety of associated warp functions that determines how the shape of the model is changed. The result is a modeling system that in conjunction with the 3D VR interface, makes conceptual free-form shape modeling simple and intuitive. In particular, this system provides the user with a means of quickly generating, modifying and visualizing conceptual shapes in an iterative manner until the desired result is achieved. This result may then be used as a starting point for the aesthetic, ergonomic and functional evaluations typical of the later stages of the product design process.

*e-mail: {kil, kreylos, hamann, staadt}@cs.ucdavis.edu
†e-mail: rufusgufus@libero.it
‡e-mail: gmonno@poliba.it

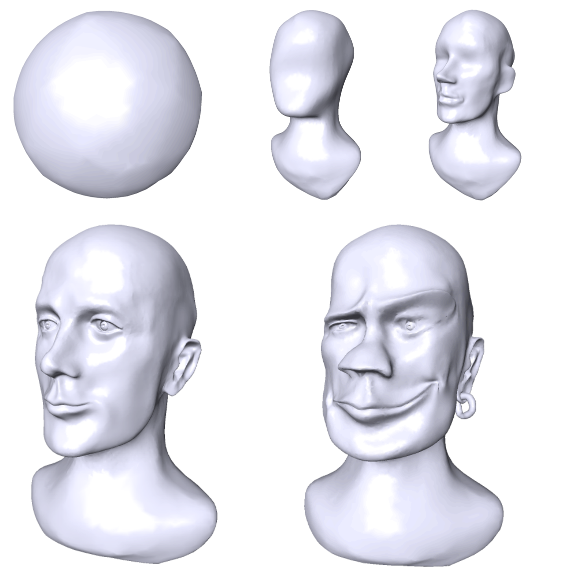

Figure 1: Sequence of snapshots from an interactive modeling session. An initial surface (a sphere) is warped into a complex bust using several brushes and warp operations. Modeling from the sphere to the bottom left bust took less than two hours. The modeling from the bottom left bust to the bottom right bust took ten minutes.

The capabilities of our modeling framework are illustrated in Figure 1. It depicts a sequence of objects created using 3D warp brushes, starting from a spherical base mesh.

2 System Overview

We use a similar setup to the one described in [2] with a Responsive Workbench display that utilizes active stereo shutter glasses and Fakespace Pinch Gloves and one Polhemus stylus with relative 6DOF sensors for two-handed interaction.

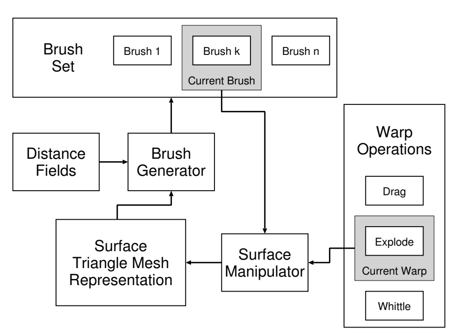

Figure 2: Overview of the system architecture.

A high-level overview of our modeling system's software architecture is depicted in Figure 2. At the core of our system is a self-adapting triangle mesh [1] (see Section 4) representing the surface of the object currently being modelled. Modeling is performed by direct free-form manipulation of the represented surface, using one or more surface manipulators associated with the 6DOF input device(s) present in the virtual environment. A surface manipulator combines the effects of a *brush* and a *warp operation* (see Section 3). Brushes define the spatial shape of a manipulator, and are defined as distance fields [3]. Warp operations define how a manipulator modifies a surface's vertices, and are defined as vector fields.

An important contribution of our modeling system is the ability to extend the set of available brush shapes from within the system. The brush generator can "clone" a part of the current surface and convert it into a distance field to be added to the set of available brushes. This allows starting from a set of pre-defined brushes loaded as existing distance fields or generated from analytical representations, e. g., spheres and ellipsoids, and generating successively more complex brushes which are finally used to model a desired shape.

3 THE 3D WARP BRUSH

Our modeling tool, the 3D warp brush, employs a brush-like paradigm with a limited area of influence in a local region of space, defined by a distance field $d(x)$ and several user-defined parameters such as influence region size and "pressure." As a warp brush moves through space, a vector field is defined locally around the brush. This is realized through warp operators $f\colon \mathbf{R}^3 \to \mathbf{R}^3$. As the brush comes into contact with the model, the warp operator is modified by weight functions calculated from the brush's distance field and applied to the vertices of the model.

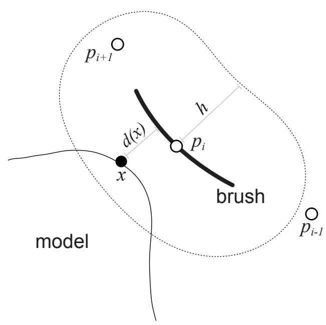

Figure 3: A brush at position p_i interacts with a model. Vertex x on the model's surface is within the region of influence h of the brush. As the brush moves through space, the warp function transforms the surface vertices within the region.

Warp operators are defined as vector fields that impose movements on surface vertices inside the influence region of a brush. At each frame, the position x of a surface vertex v is adjusted by multiplying the weight $w(x)$ with the warp operator value $f(x)$, and adding the result to the current value $x : x' = x + w(x) \cdot f(x)$.

Warp operators can be arbitrary vector fields, and they can depend on a vertex' position, on the brush position, on the brush motion, on the brush orientation, or on any combination of these. Our current modeling system contains three different warp operators: *drag*, *explode and collapse*, and *whittle*.

Drag is a simple yet powerful way of manipulating the model by allowing pushing, pulling, and twisting of regions of its surface. This operator's vector field is based on brush motion only.

Explode and collapse has the effect of moving the model's surface towards or away from the brush's surface, depending on the gradient of the distance field. The effect can be thought of as blowing up, or sucking the air out of, a balloon containing the brush, or as stamping the brush into the model's surface.

The whittle operator is used to locally smooth the surface of a model. Our system uses Laplacian smoothing to modify the positions of surface vertices inside the brush's influence region.

As described above, our system represents brush shapes as distance fields $d(x)$. These distance fields can be generated in several ways. They can be given analytically, e. g., for spherical or ellipsoidal distance fields, or they can be loaded as pre-computed regular grids. This generates distance fields for brushes directly from the surface of our models, which allows intuitive creation of complex brush shapes using the same modeling tools that are used for surface modeling.

4 SURFACE REPRESENTATION

The main goal in developing our surface representation was to strike a compromise between the ease of rendering of polygon meshes, and the ease of manipulation of point set surfaces, especially when inserting/removing points. The result was a triangle mesh data structure, with built-in operations for point insertion and point removal. The surface is rendered just like a triangle mesh, but it behaves like a point set under manipulation. When the mesh is stretched, and local point density becomes too low, new points are inserted automatically; when the mesh is condensed, and local point density becomes too high, points are removed automatically. Additionally, the point insertion/removal operations are implemented such that the shape of triangles in the mesh is as uniform as possible, preferring isosceles triangles over long and skinny ones. This is achieved by *recursive longest-edge bisection* and *shortest-edge collapse*.

5 CONCLUSIONS

We developed a new framework for free-form modeling of triangle meshes using 3D warp brushes. Defining the shape of warp brushes using signed distance fields along with different warp function allows us to create new brushes in an efficient and flexible manner. The use of this framework in immersive virtual reality environments with 6DOF interaction devices supports intuitive and direct manipulation of surface shapes. Our results demonstrate the capabilities of our system with several example models.

6 ACKNOWLEDGMENTS

This work was supported by NSF under contract ACI 9624034, the LSSDSV program under contract ACI 9982251, through NPACI, and a large ITR grant. We gratefully acknowledge the support of the W.M. Keck Foundation, and thank the members of IDAV at UC Davis.

REFERENCES

[1] J. R. Bill and S Lodha. Computer sculpting of polygonal models using virtual tools. In *Tech. Rep. UCSC-CRL-94-27*, Baskin Center for Computer Engineering and Information Sciences, University of California, Santa Cruz, US, July 1994.

[2] Lawrence D. Cutler, Bernd Fröhlich, and Pat Hanrahan. Two-handed direct manipulation on the responsive workbench. In *Symposium on Interactive 3D Graphics*, pages 107–114, 191, 1997.

[3] R Perry and S Frisken. Kizamu: A system for sculpting digital characters. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 47–56, Los Angeles, CA, August 2001.