Three-Dimensional Modeling Interface for Augmented Realities

Thomas Jung

HTW Berlin
Wilhelminenhofstraße 75A
D-12459 Berlin

t.jung@htw-berlin.de

Abstract

Today’s geometric modeling tools are usually controlled by the mouse and keyboard. 3d-modeling operations are often constrained to planes or lines to make modeling easier. New augmented reality devices promise new classes of applications with natural 3d user interfaces. This paper deals with a new method that enables users to model 3d shapes directly with their fingers. It is implemented in Artist3D, which is a modeling tool that constraints modeling operations implicitly to planes and lines. The new method can also be integrated into commercial modeling tools.

1 Introduction

New augmented reality devices with better 3d registration between virtuality and reality (e.g., Microsoft HoloLens [Mic16a] or Leap Motion in combination with Oculus Rift [Lea16]) enable the development of new classes of applications with 3d user interfaces. For such applications, especially the registration of the user's hands as part of reality with the virtual scene is of great importance. As a result, users can directly interact with the scene by using their hands.

There exist at least three different types of 3d user interfaces: application control interfaces, navigation interfaces, and manipulation interfaces [H97] [JH13]. While manipulation interfaces facilitate rotating, translating or scaling of objects as a whole, 3d modeling interfaces are used to modify the shape of an object. Thereby modeling operations include the manipulation of object parts or control points.

Commercial geometric modeling systems (e.g., Maya, 3DS Max) enable the user to design three-dimensional scenes. The user interfaces are often complex and require a high learning effort. Users mostly interact by using classical devices like the mouse and keyboard. Since input devices with more than two degrees of freedom are currently rarely used, main interaction concepts in geometric modeling systems cannot be based upon such devices. These devices (e.g., Space mouse [Conn16]) are only an option for manipulation tasks or camera control.

Interactions in 3d space are difficult to perform. Users are not trained in 3d interactions in virtuality. 3d manipulations in reality often require haptic feedback (e.g., plug an ill-fitting electric plug into a socket [SC11]). Moreover, user interfaces in reality are often not full 3d (e.g., cutlery trays, doors, cockpits). Therefore, adding constraints to generic 3d interaction techniques can improve usability [SC11] [BKLP04, p. 180f].

Natural user interfaces where users interact directly by using hands or fingers promise easier modeling of three-dimensional scenes. Currently, natural user interfaces for 3d modeling tasks are subject of research (e.g., Microsoft HoloStudio system [Mic16b] or virtual pottery [RR15].)

2 Previous Work

There are several approaches to manipulate objects in three-dimensional space by using a *two-dimensional input device* (e.g., computer mouse). A method developed by Nielson and Olsen constrains object rotation and scaling into a plane that is defined by a selected object's local coordinate system [NO86]. Bier presents a more general approach. Users can attach a cursor coordinate system ("skitter") to an object's coordinate system ("jack"). Skitters, for example can be moved around on the surfaces of objects. Objects can be rotated around or translated along the axis of a coordinate system [Bie86]. Houde suggests different handles on the bounding box of an object to indicate how it can be rotated or translated [Hou92]. Conner et al. present several three-dimensional widgets that allow manipulating objects and modifying their shapes (e.g., bending, twisting, and tapering) [CSHRZv92]. Zeleznik et al., describe a toolkit that allows users developing their own widgets [ZHRHM93].

Clark enables users wearing a head-mounted display to design curves in 3d by using a *three-dimensional input device* ("wand") [Cla76]. Butterworth et al. presented an immersive modeling system with a menu ("toolbox") as part of a hierarchical 3d scene. There are tools for adding triangles, extruding polylines, and creating standard surface shapes. The system includes a snap-to grid and a snap-to plane mode [BDHMO92]. There are also solutions that support two-handed interactions in planes by using real props (e.g., [SG97]) or virtual props (e.g., [SES99]) ("pen-and-pad-metaphor") in an augmented reality environment.

Artist3D is a modeling tool that uses modeling planes and lines to constrain modeling operations [Jun06]. Users choose modeling operations by selecting vertices of objects with two-dimensional input devices; for example, different corners of a box are moved implicitly in different planes to scale and rotate the box. As another example, the vertex of a rotational solid can be moved in a plane spanned by the symmetric axis of the object and the chosen vertex in order to change the radius and the height of an object's ring. A multi-touch interface allows controlling the view by using two or three fingers to avoid mode changes caused by camera control [Jun09]. In an immersive scenario, Artist3D is controlled by three-dimensional finger positions [Jun15].

3 New 3d modeling method

Currently, two- and three-dimensional input devices are used in completely different scenarios. Because of the lack of *three-dimensional* input devices at the workplace of most users, main modeling concepts of commercial modeling tools cannot depend on such input devices. On the other hand, in order to ease modeling operations in virtual environments, 3d input often is constrained to *two dimensions*. In the following paragraphs, a method is presented that combines two- and three-dimensional input into a common modeling interface.

A simple scaling operation is shown in Fig. 1. The user selects the top right vertex of the box to choose a two-dimensional scaling operation and the appropriate modeling plane by pressing the left mouse button. The cursor is projected onto the modeling plane by using ray-casting. The user scales the box by dragging the box corner and finishes the modeling operation by releasing mouse button.

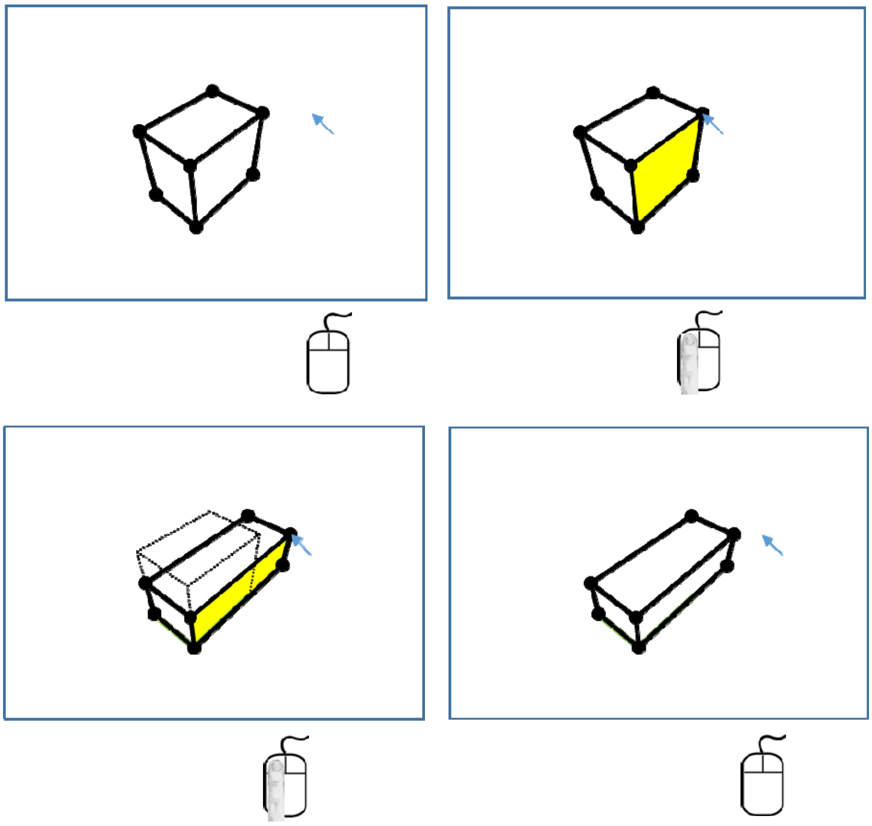

Fig. 1: Scaling a box with a mouse

This drag-and-drop operation can be divided into four phases:

1. Identifying a vertex by using the mouse pointer
2. Selecting the vertex by pressing a mouse button
3. Modifying the object by moving the mouse
4. Finishing the operation by releasing the mouse button

In Artist3D the modeling plane only depends on the chosen vertex. Therefore, these four phases can also be executed with a three-dimensional input device in a virtual environment:

1. Identifying a vertex that is nearest to the 3d cursor
2. Selecting the vertex, when the distance from the 3d cursor to the vertex falls below a threshold
3. Modifying the object by moving the 3d cursor in a 3d layer surrounding the modeling plane; thereby, the 3d cursor position is snapped to the plane
4. Finishing the operation when the 3d cursor leaves the 3d layer

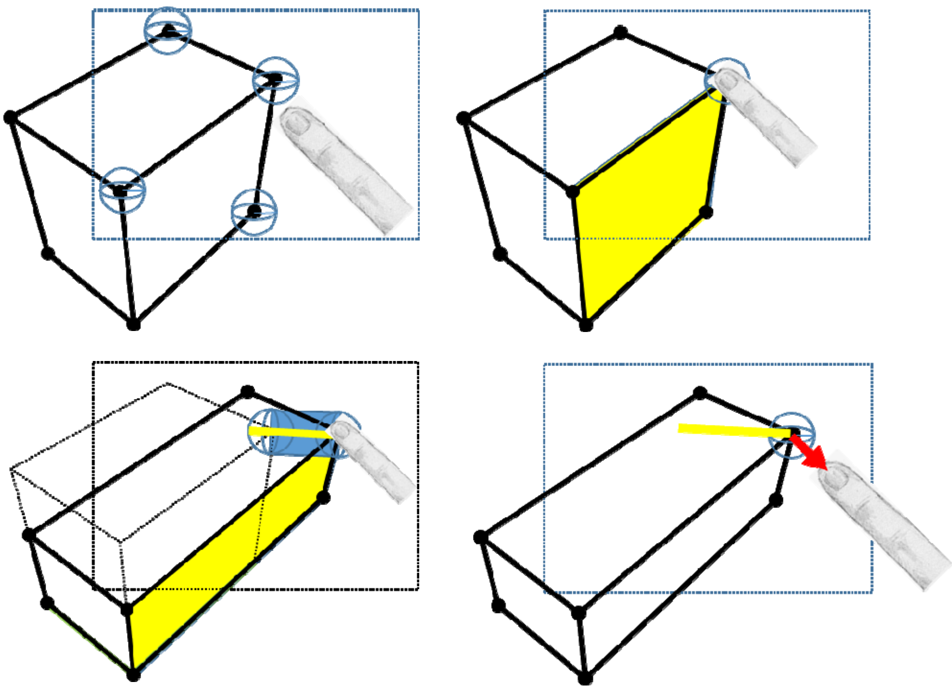

Fig. 2: Scaling a box with the new method by using a tracked finger

The simple scaling operation using this new method is shown in Fig. 2. The fingertip represents the 3d cursor. There is no need for explicit gestures to confirm selection. As a result, the modeling operations are controllable just by using the 3d cursor or fingertip, respectively. Since users wear a head-mounted display, they can easily recognize the distances between fingers and object surfaces.

This method is advantageous since there is no need for an explicit confirmation gesture (e.g., an “air tap” in Microsoft HoloStudio [Mic16b]) to avoid interference with the 3d cursor position. Therefore, modeling can be done much more efficiently and precisely.

Since the four phases of the new method correspond to the four phases of the traditional mouse-controlled interface, commercial modeling tools can directly benefit from the existence of a three-dimensional input device. The learning effort will be reduced by using the same modeling operations with either a 2d or a 3d input device

In a first step, the *three-dimensional widgets* of the modeling tool can be controlled by using this method. In further steps, all modeling operations have to be redefined to use implicit defined modeling planes or lines like in Artist3D that only depend on chosen vertices.

4 Results and outlook

A first prototype was developed that integrates the new method into the Artist3D modeling tool. The prototype uses a combination of Oculus Rift DK2 and Leap Motion [Lea16]. Modeling operations can be performed successfully with a single finger (see screenshots in Fig. 3 and 4). Since the vertices must be in reach of the fingers, a navigation interface has to be implemented. Following the approach of Artist3D this could be done by using multiple fingers to avoid mode changes, but currently the fingertip detection provided by Leap Motion is neither stable nor precise enough. Further steps will be to perform usability tests.

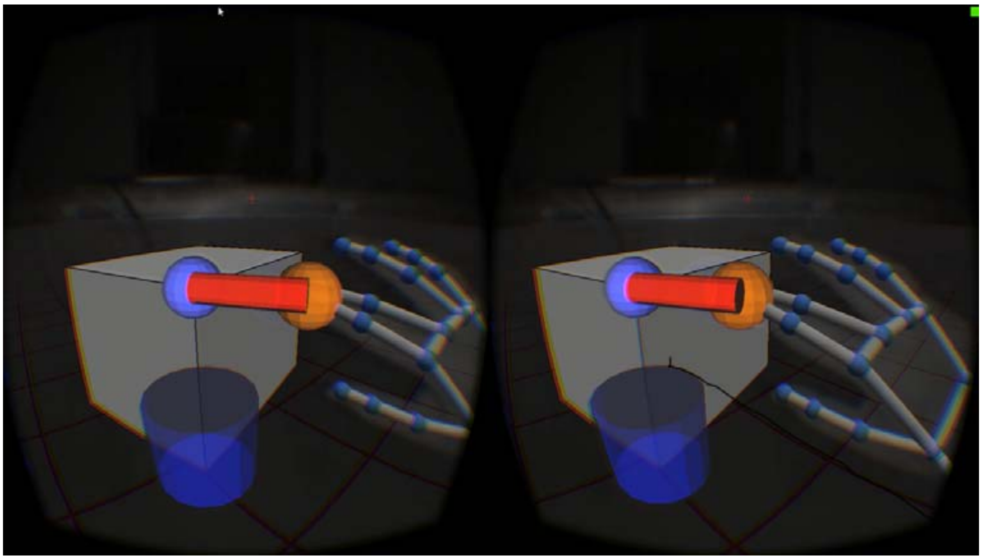

Fig. 3: New method: spheres show vertex to choose and fingertip position, the length of the line must fall below threshold

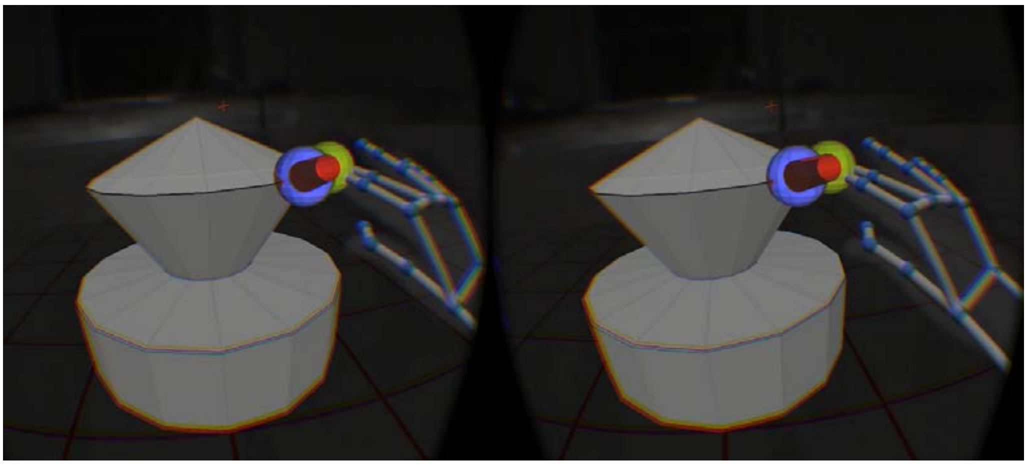

Fig. 4: Modifying the height and radius of a ring as part of a rotational solid

Literature

[Mic16a] https://www.microsoft.com/microsoft-hololens/en-us, retrieved Mar 8th 2016

[Lea16] https://www.leapmotion.com/product/vr, retrieved Mar 8th 2016

[H97] C. Hand. 1997. A Survey of 3D Interaction Techniques. *Computer Graphics Forum,* Vol. 16 (1997): pp. 269–281.

[JH13] J. Jankowski, M. Hachet. 2013. A survey of interaction techniques for interactive 3d environments, in: *Eurographics State of the Art Reports*, 2013.

[Con16] http://www.3dconnexion.de/, retrieved Mar 8th 2016

[SC11] Wolfgang Stuerzlinger, Chadwick A. Wingrave. 2011. The Value of Constraints for 3D User Interfaces, *Dagstuhl-Seminar* 2008, Springer 2011

[BKLP04] Doug A. Bowman, Ernst Kruijff, Joseph L. LaViola, Ivan Poupyrev. 2004. *3D User Interfaces Theory and Practice*, Addison Wesley 2004

[Mic16b] https://www.youtube.com/watch?v=b6sL_5Wgvrg&t=8m22s, retrieved Mar 8th 2016

[RR15] Vinayak and Karthik Ramani. 2015. A Gesture-free Geometric Approach for Mid-Air Expression of Design Intent, *Computer-Aided Design,* Vol. 69, December 2015, pp. 11–24

[NO86] Gregory M. Nielson and Dan R. Olsen, Jr. 1986. Direct manipulation techniques for 3D objects using 2D locator devices. In *Proceedings of the 1986 workshop on Interactive 3D graphics* (I3D '86), Frank Crow and Stephen M. Pizer (Eds.). ACM, New York, NY, USA, pp. 175-182.

[Bie86] Eric Allan Bier. 1986. Skitters and jacks: interactive 3D positioning tools. In *Proceedings of the 1986 workshop on Interactive 3D graphics* (I3D '86), Frank Crow and Stephen M. Pizer (Eds.). ACM, New York, NY, USA, pp. 183-196.

[Hou92] Stephanie Houde. 1992. Iterative design of an interface for easy 3-D direct manipulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '92), Penny Bauersfeld, John Bennett, and Gene Lynch (Eds.). ACM, New York, NY, USA, pp. 135-142.

[CSHRZv92] Brookshire D. Conner, Scott S. Snibbe, Kenneth P. Herndon, Daniel C. Robbins, Robert C. Zeleznik, and Andries van Dam. 1992. Three-dimensional widgets. In *Proceedings of the 1992 symposium on Interactive 3D graphics* (I3D '92). ACM, New York, NY, USA, pp. 183-188.

[ZHRHM93] Robert C. Zeleznik, Kenneth P. Herndon, Daniel C. Robbins, Nate Huang, Tom Meyer, Noah Parker, and John F. Hughes. 1993. An interactive 3D toolkit for constructing 3D widgets. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques* (SIGGRAPH '93).

[Cla76] James H. Clark. 1976. Designing surfaces in 3-D. *Commun. ACM* 19, 8 (August 1976), pp. 454-460.

[BDHMO92] Jeff Butterworth, Andrew Davidson, Stephen Hench, and Marc. T. Olano. 1992. 3DM: a three dimensional modeler using a head-mounted display. In *Proceedings of the 1992 symposium on Interactive 3D graphics* (I3D '92). ACM, New York, NY, USA, pp. 135-138.

[SG97] Zs. Szalavári and M. Gervautz. 1997. The Personal Interaction Panel – A Two Handed Interface for Augmented Reality, *Computer Graphics Forum (Proceedings of EUROGRAPHICS'97),* Volume 16 (3), pp. 335-346.

[SES99] Dieter Schmalstieg, L. Miguel Encarnação, and Zsolt Szalavári. 1999. Using transparent props for interaction with the virtual table. In *Proceedings of the 1999 symposium on Interactive 3D graphics* (I3D '99). ACM, New York, NY, USA, pp. 147-153.

[Jun06] Thomas Jung. 2006. Artist3D - Ein einfach bedienbares Werkzeug zur 3D-Modellierung von Oberflächen, In Proc. Workshop 3D-NordOst 2006, Berlin, Germany, Dezember 2006, pp. 13-20.

[Jung09] Thomas Jung. 2009. Artist3D - Ein einfach zu bedienendes Multitouch-fähiges User Interface für die 3D-Modellierung, In Proceedings of 04. Kongres Multimediatechnik, Wismar, Germany, October 2009

[Jun15] Thomas Jung. 2015. Geräteübergreifende effiziente 3D-Modelliertechnik für 3D-Interaktionsräume,Touch-Interfaces und WIMP-Systeme basierend auf implizit bestimmten Modellierebenen oder -geraden, In Proc. Workshop 3D-NordOst 2015, Berlin, Germany, December 2015, pp. 155-164.