EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling (2004)
John F. Hughes and Joaquim A. Jorge (Editors)

From raw 3D-Sketches to exact CAD product models – Concept for an assistant-system

Dipl. Ing. Holger Diehl, Dipl. Ing. Franz Müller, Prof. Udo Lindemann
Lehrstuhl für Produktentwicklung, TU München
Boltzmannstr. 15, 85748 Garching
{diehl,mueller,lindem}@pe.mw.tum.de

ABSTRACT

As we try to develop new 3D-Sketching devices we come across the problem of how to transfer 3D-Sketches into exact product models to gain an integrated design system. There are some promising approaches for the retrieving of 3D-geometries out of 2D-Sketches, but nearly none supports the concretization of 3D-Sketches. We are developing an intelligent assistant that supports the designer in the task of concretization. Our higher level goal is a system that uses a cad-kernel and supports the designer from the early stages of sketching up to the modelling of exact product representations.

I.2.10 [Vision and Scene Understanding]: Shape, Design, 3D/Stereo scene analysis. I.3.5 [Computational Geometry and Object Modeling]: Curve, surface, solid, and object representations; J6 Computer Aided Engeneering: Computer aided Design;

1. Introduction

CAD-Systems have established themselves as a standard tool in the product development process long time ago. Over the years they have become very powerful, but also very complex and are therefore often seen as a science of their own. For every designer it is a long learning process to get to know all the CAD-modeling functions and to handle them correctly. Even if he has reached that point the CAD-system is still not very intuitive in use.

Especially in the early phases of product-design CAD-systems do not meet the designer's needs. CAD-Systems are designed to model exact representations of the future product, but not for the fast and intuitive generation of raw sketches, which are often not only created during the conceptual stage of design. Pache made a survey among designers and asked them how often they use sketches on the one hand for preparing their work with CAD and on the other hand during their work at a CAD-system. Over fifty percent pointed out that they use sketches, before turning their ideas into CAD. Furthermore, about 35% percent also use sketches during work with CAD-systems. [PL*03]

The main reason for these results is the fact that the process of sketching is very intuitive and fast. Sketches provide the possibility to record abstract information and therefore they come very close to the designer's mental model of the product.[Hac*02]

The problem is that the sketched information has to be manually transferred into the CAD-System at some point of the development process. The designer is forced to invest a lot of time and thinking to concretize the raw sketch. In this phase it would be very helpful to have an assistant-system that supports the generation of exact digital geometries out of the sketches.

If the designer wants to sketch a solid figure, like e.g. a cube, on a sheet of paper he has to foreshorten it. To enable the designer to not only create a perspective drawing but a three-dimensional cube we developed a 3D-Sketching-System which allows generating 3D-Sketches in a very intuitive way (see Figure 1).

The image displayed on a CRT-monitor is being projected on a semitransparent mirror. Beneath the mirror you can see the 6DOF-input device Phantom Desktop. We use an active stereo display technique with shutter glasses. Through this setup we are able to merge interaction- and display room. This makes sketching in 3D very intuitive.

Due to the limitation of the interaction room of the Phantom Desktop we also developed a 3D-Skteching-System for a data glove (see Figure 2). We use a CyberGlove with 22 Sensors in combination with the electromagnetic tracking system flock of birds. For interaction, we defined four static gestures: A pointing gesture for sketching lines into 3D-Space, a fist for grasping and moving objects, a flat hand for confirming certain actions and a pointing gesture with the index and middle finger stretched to select distant objects with a virtual extension of the middle finger. 3D-Buttons which are located on the left and right side of the window are used to call extra functions.

Figure 1: *Phantom based 3D-Sketching system*

Because sketching of objects which have surface with no edges, like spheres or only with a few edges, like cones, just with lines is a little bit difficult, we implemented a function to create basic geometries like spheres or cylinders. Beside this the user is able to erase single or multi objects and to resize objects.

Figure 2: *Gesture based 3D-Sketching system*

We developed the described digital Sketching tools, which are quite promising, but to gain an integrated system we have to solve the problem of transferring raw sketches into exact product-models. In this paper we want to discuss a concept for a concretization assistant which generates vector-based 3D-geometries out of voxel-based 3D-sketches.

2. State of the art

Because CAD-Systems are not very intuitive and don't support the early phases efficient enough a lot of research has been done to develop new specialised modelling tools. All these tools have in common, that they use new input devices.

One of these systems is AR-Sketcher [FdAS*97]. It uses a tracked pen together with a tablet as input device. Different VR output devices as well as lookthrough glasses are being supported as output device. The system has various functions to model freeform surfaces, lines and primitives. What is special about this system is that it uses a CAD-kernel. All created models can directly be used with CAD-Systems. SketchAR doesn't support the concretization of the raw models.

A system that seems to be very similar to SketchAR is being described by Wesche [Wes*00]. It also uses a pen based input in combination with a tablet.

3D-Draw [SRS*91]is one of the first systems which supports 3D modelling functions. Here we can also find a tracked pen in combination with a tracked tablet as input device.

SKETCH is a 2D-system supporting a 3D-Button-Mouse. Different Strokes are defined to model geometries [ZHH*96].

A system that is only supporting 2D input is described by Igarshi [IgHu*01]. Geometries consisting of straight lines can be modelled with the help of an assistant. Suggestions are visualized in a separate window. Ambiguity is dealt by the possibility of multisuggestions.

ARCADE is a System similar to SketchAR but has its focus on the intuitive modelling of exact product representations and on collaborative design. A system that uses multimodal Input is COVIRDS [DCG*97]. This system is also focused on exact modelling. It has an CAD-interface.

GIDeS [PJB*00] is a system that has sketching-functions and functions for reconstruction of geometries out of sketches. Through a list of defined gestures certain user intentions can be retrieved. Suggestions are being displayed which can be then accepted through the user. A pen without buttons in combination with a tablet is used as input device. GIDES++ is based on GIDeS but has a multimodal input. It is possible to export and Import STEP and IGES-Files. Objects and edge dimensions can be change through handwriting or spoken commands.

Igarshi [IMT*99] developed a System for modelling 3D-Freeform surfaces with a 2D input device.

3. The manual concretization process

If we want to analyse the manual concretization process we have to take a closer look at the differences between sketches and CAD-models.

In sketches you can find different information. There are variants of the product's geometry represented through different lines of varying thicknesses. But lines can also represent kinematics. If a part of the product is very complex or this part is not in the focus of the first modelling step, designers often use symbols or notations to abstract this part.

Not all information can be transformed into the CAD-model. This shows that we also have a loss of information when going from sketches to CAD-models.

Sketches allow an intuitive modelling without defined rules or functions. This stands in contrast to the very strict way a product has to be modelled in CAD-Systems. Sometimes this leads to a counterproductive way of modeling. Designers then try to design their products in a way that can be modeled with the given CAD-functions. But to be forced to ask yourself how to model the product with the given functions can also have the positive effect of reflecting the first idea. Through this, weak points are being recognized and can then be redesigned. Areas that are too abstract e.g. symbols or notations have to be concretized. Therefore designers again use sketches. So concretization can bee seen as an iterative process. [Hac*02].

4. Concept for an semiautomatic concretization process

To make the concretization process faster an assistant would be very helpful. This assistant is a computational assistant that works with digital data, which is generated by our 3D-sketching-systems. Because we use 3D-Data we need an assistant that is able to "think" in 3D.

The assistant should be very intuitive in its use. When designing this assistant we have to think about some important aspects. One of these aspects is the point of time at which the concretization assistant acts. Another aspect is how the suggestions are being represented without hindering the designer in his creative process?

There are three principal ways an assistant can be designed in matters of the point of time it is acting. One way could be assisting the designer already during sketching. As soon as a geometric object is being recognized by the assistant it would be visualized in some way that is being discussed later. Another option could be a concretization at a user defined point. We will discuss different ways how this could be realized. A very complex option is a concretization after sketching.

One main motivation for the development of an assistant are results from user-tests with our 3D-sketching-systems. Users who had to sketch certain objects with our 3D-Sketching system pointed out that it was very difficult to sketch 3D-geometries which have surfaces without edges. We could solve this problem by a button-based menu, but this would lead to a design very similar to common CAD-systems. Too many buttons will make our system complex and not intuitive. That's why we think a concretization assistant could help to solve this problem.

4.1 Visualization of suggestions

A concretization assistant should recognize the user's intention as early as possible to save time and to reduce the complexity. The suggestion of the assistant should be visualized in a way that doesn't hinder the designer. Also the user should not be forced to accept or withdraw the suggestion. If he wants to accept a suggestion he should be able to do so, but he also should be able to go on with sketching without having to withdraw the suggestion.

One possible option could be visualizing the suggestion in form of a transparent geometry at the place of the sketched object. The advantage of this visualization technique is that the user can immediately see if the suggestion is right or wrong. Figure 4 shows how such a concretization could look like for a cube.

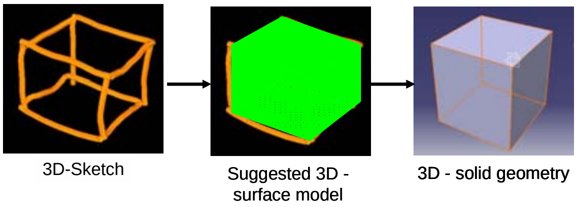

Figure 3: *Concretization of a cube*

A difficulty that arises when using transparent suggestions is ambiguity. For example if a user draws two rectangles which are orthogonal to each other the assistant might assume that the user intends to draw a cube. But maybe the user only wants to draw two rectangles. So the assistant should suggest both possible intentions. Because two suggestions can not be visualized as transparent geometries at the same place, at the same time it would be better to visualize them in a separate window like Igarshi proposes it [IH*01]. (similar with gides)

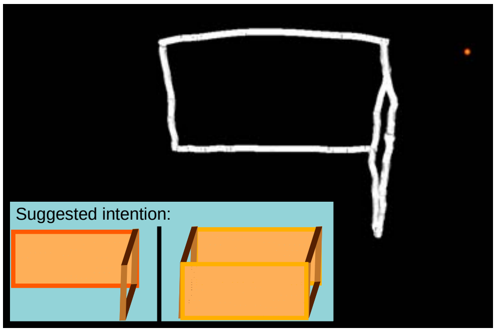

Figure 4: *Suggested intentions visualized in a separate window*

The next chapter discusses how we want to implement our geometry recognition.

4.2 From the Recognition of basic geometries to intuitive gestures

For the implementation of the geometry recognition we can use classical algebra only if all users sketch in a more or less similar way. Our goal is to develop a stable and user independent system. People shouldn't have to put too much effort in learning how to sketch. That's why an artificial intelligence approach seems to be more promising. Humans are able to recognize geometries which are sketched in different ways without any difficulty. For example humans usually don't have any problems to cognize that the three pictures in Figure 5 are different representations of a cone.

Humans learn through experiences how to interpret objects. Young babies aren't able to interpret objects as adults do because they didn't have enough time to interact with their environment to gain the necessary experience.

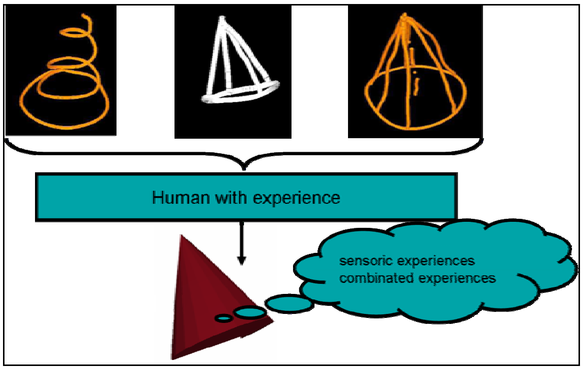

Figure 5: *Interpeting different representations of a cone with human intelligence.*

If we want to create an assistant which uses artificial intelligence this assistant has to be treated with a lot of examples to learn how to interpret 3D-Objects. First step will be the recognition of basic geometries like spheres, cylinders and cubes, which are sketched into 3D-Space. Therefore we will ask subjects to perform given modeling tasks, like: "model a sphere with the given input device". Besides using the inputs of the subjects to train the concretization assistant we will use them to get a hint on how we have to define intuitive gestures. If the number of subjects is big enough we will see how most of the subjects move the input device when performing the given modeling task. With this information we will be able to define dynamic gestures for the generation of basic geometries like spheres, cones or cylinders. This would also reduce time and complexity for the interpretation of 3D-geometries.

Integrating human intelligence in the concretization process will also reduce complexity. If you take the example of the three representations of a cone the user could give a hint for the interpretation of the object in form of speech in combination with a gesture. Compared to an alternative input over a number of buttons speech commands are much more intuitive.

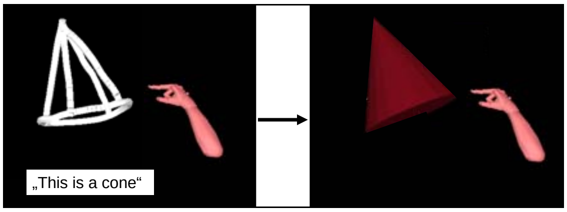

Figure 6: *Integrating the user in the concretezation process*

At the moment we are trying to integrate a speech recognition system in our VR application for the creation and concretization of basic geometries.

4.3 Recognition of advanced information

The recognition of basic geometries is a first step towards an automated concretization process. But what happens with text or symbol based information? Does it make sense to help the designer with recognizing also this kind of information?

Very interesting is the recognition of standard parts like screws or bearings. This could help to gain a very fast modeling process. A user could then write down the description of a standard part or alternatively sketch a symbol for the standard part. As soon as the text or symbol is being recognized by the assistant and the equivalent standard part has been found in the database it could be replaced through the geometric representation. Different concretization steps would make sense. First a user may only need a representation of the category of that standard part without having to define specific parameters. But some parameters like diameter, length or thickness of the geometric representation should be adaptable through the user at runtime in an intuitive way. If we use a data glove this could be realized with gestures imitating the natural interaction with this object. For example to shorten the diameter of a bearing one would press with both hands from the outside toward the inside.

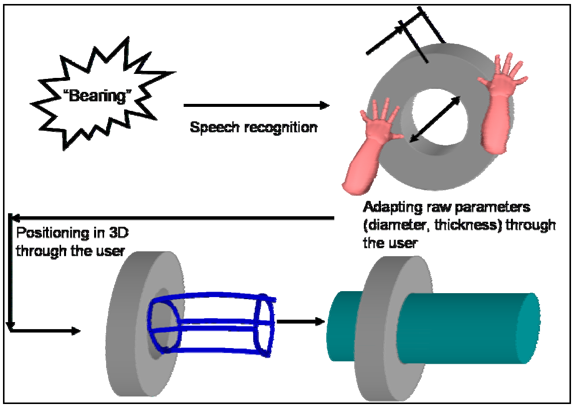

Figure 7: *Creating and adapting a standard part*

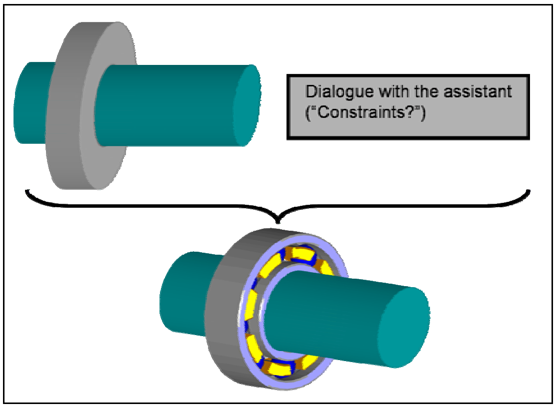

Figure 8: *Concretization of a standard part*

At a user defined point an exact representation can be created. Therefore the user would define constraints in dialogue with the assistant.

The idea of recognizing standard parts comes very close to the principal of feature based modeling. That's why it would make sense to link this function with the CAD-database to prevent a reinvention of the feature based modeling idea. Through the integration into a VR- environment the handling of these features would be much more intuitive. New interaction techniques would make it possible to intuitively adapt parameters.

In respect to the complexity of the implementation of a 3D-character-recognition it makes more sense to use speech recognition to describe standard parts.

Lines that represent kinematics are also important information that can be found in sketches and might be very useful for a new generation of modeling systems. Most of the times you want to realize one or more certain functions with the modeling of your product. A lot of these functions at least in the area of mechanical engineering do not work without kinematics. It would be very helpful and resolving if we could use the lines which represent kinematics to simulate the dynamic behavior of our product in a very early stage.

Conclusion

We developed two systems for 3D-Sketching which are very promising, but also have some restrictions. To solve these restrictions and to realise an integrated product development process we think it is necessary to develop a concretization assistant. Not only geometric information should be concretised. Also the recognition of representations of kinematics in combination with their simulation has a high potential for much better support of the early phases of product development. The development of algorithms for the recognition of 3D-geometries is the main work in our first step.

References

[PL*03] Pache, M.; Lindemann, Sketching in 3D. Human Behaviour in Design, Springer Berlin 2003

[Hac*02] Hacker, W.; "Denken in der Produktentwicklung", vdf, Hochschulverl. an der ETH Zürich, 2002

[Wes*00] Conceptual Free-Form Styling on the Responsive Workbench. G. Wesche and M. Droske. *VRST 2000 Proceedings.* Seoul, Korea, October 2000.

[FdAS*97] Fiorentino Michele, De Amicis R., Stork A., Monno G., "Spacedesign: A Mixed Reality Workspace for Aesthetic Industrial Design", Proc. of ISMAR 2002 IEEE and ACM INTERNATIONAL SYMPOSIUM ON MIXED AND AUGMENTED REALITY, pp. 86-94, Sept. 30 - Oct. 1, 2002, Darmstadt, Germany. ISBN 0-7695-1781-1.

[SRS*91] Sachs, E., Roberts, A., Stoops, D.: "3Draw: A Tool for Designing 3D Shapes", IEEE Computer Graphics and Applications, 11, 1991, pp 18-26

[SSdA*00] Stork A., Schimpke O., De Amicis R., "Sketching Free-Forms in semi-immersive Virtual Enviroments", proceedings of ASME 2000 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Baltimore, Maryland, 2000.

[DCG*97] Dani T.H., Chu PCh., Gadh R., "CORVIDS: shape modeling in a virtual reality environment", proceedings of 1997 ASME Design Engineering Technical Conferences, 1997, Sacramento, California.

[IMT*99] Igarashi, T.; Matsuoka, S.; Tanaka, H.: Teddy: a sketching interface for 3D freeform design; International Conference on Computer Graphics and Interactive Techniques archive Proceedings of the 26th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co. New York, NY, USA; 1999

[IH*01] Takeo Igarashi, John F. Hughes, "A Suggestive Interface for 3D Drawing ", 14th Annual Symposium on User Interface Software and Technology, ACM UIST'01, Orlando, Florida, November 11-14, 2001, pp.173-181.

[PJB*00] Pereira J., Jorge J., Branco V., Nunes F. Towards calligraphic interfaces: sketching 3D scenes with gestures and context icons. WSCG'2000. Conference Proceedings, Skala V. Ed., 2000.

[ZHH*96] Zeleznik R.C., Herndon K.P., Hughes J.F. SKETCH: An interface for sketching 3D scenes. SIGGRAPH'96, 163-170, 1996.