

iSphere: A Free-Hand 3D Modeling Interface

Chia-Hsun Jackie Lee, Yuchang Hu, and Ted Selker
Massachusetts Institute of Technology
Context-Aware Computing Group, The Media Laboratory
20 Ames ST., E15-324, Cambridge, MA 02139
+1 617 253 4564
{jackylee, yhu, selker}@media.mit.edu

Abstract

Making 3D models should be an easy and intuitive task like Free-hand sketching. This paper presents iSphere, a 24 degree of freedom 3D input device. iSphere is a dodecahedron embedded with 12 capacitive sensors for pulling-out and pressing-in manipulation on 12 control points of 3D geometries. iSphere exhibits the top-down 3D modeling approach for saving mental loads of low-level machineries. Using analog inputs of 3D manipulation, designers are able to have high-level modeling concepts like *push* or *pull* the 3D surfaces. Our experiment shows that iSphere saved steps of selecting control points and going through menus and make subjects more focus on what they want to build instead of how they can build. Novices saved significant time for learning 3D manipulation and making conceptual models, but lacking of fidelity is an issue of analog input device.

1. INTRODUCTION

Freehand sketching is the most common way for designers to express ideas directly and interactively. Modern CAD systems provide computational platforms for automating drawing processes and novel representations. However, the input interfaces of most systems usually consist of machinery of low-level commands and trivial mode-switching manipulations. Usually, it costs months to become an expert to perform designs in those CAD systems. This paper is to introduce a realistic way of 3D manipulation interfaces by which even novice users can manipulate models intuitively with their own hands.

Making 3D models isn't an easy task for 3D designers. In modern CAD systems, designers take bottom-up approaches to 3D modeling. They have to set the goal state of what to build in mind and then decompose it into a series of smaller pieces before turning it into a set of modeling commands. However, it is not easy for designers to think and perform modeling commands at the same time. Dealing with trivial, disruptive and low-level steps of modeling commands should not take too much mental load from designers while they spend efforts in designing. There is a strong need to quickly connect modeling concepts to shapes.

Typical 3D modeling systems, like Rhino, 3D Studio MAX or Alias|Wavefront Maya, usually consist of sets of abstract commands. Users are always performing the mediating actions between high-level modeling concepts and low-level manipulation commands. Although the 3D modeling functionality was mature in most 3D systems, the gap between realistic interaction and low-level commands is still left unsolved. To manipulate 3D environments efficiently may be the result of simplifying cognitive behavior to perform mappings and powerful commands intuitively. Designing a high-level modeling system can possibly reduce users' cognitive load.

We argue that an input device which can use a spatial metaphor to map hand actions into modeling commands can improve the processes of 3D modeling. Mapping hand actions into analog inputs on a parametric model can eliminate a series of viewing and editing commands. In other words, designers can use natural gestures to manipulate 3D objects. Understanding hand actions can save tasks from the software interface. 3D input systems should understand user's behavior to provide interaction directly and meaningfully. Direct mapping of realistic modeling concepts, such as push, pull and twist actions should be easy to learn and remember.

We present iSphere dodecahedron with two analog sensing modes per face, as in Figure 1. Hand-position aware mechanism has been equipped into the physical reference of 3D modeling. A study was conducted to compare the performance using standard command-based interfaces and iSphere.

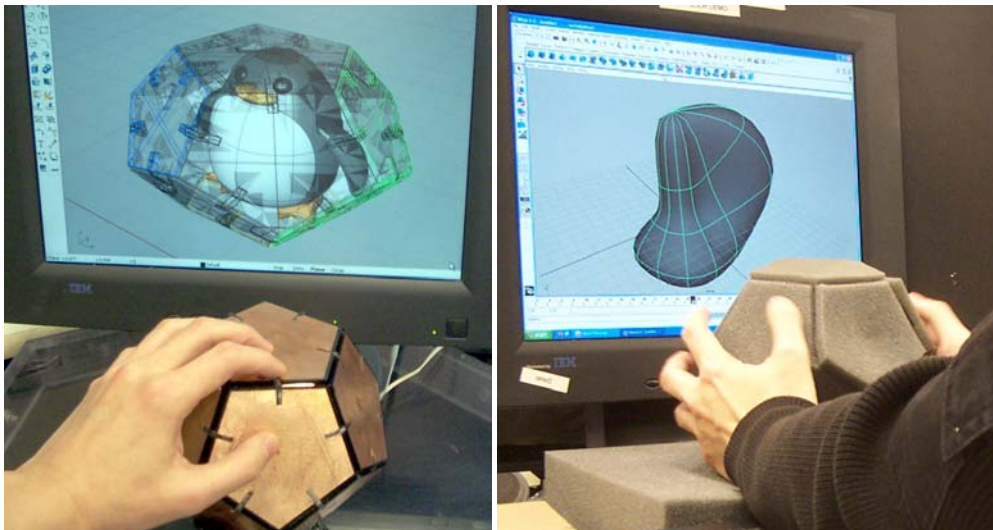


Figure 1: iSphere is a dodecahedron with capacitive sensors to interpret hand positions into high-level 3D modeling commands.

2. RELATED WORK

User interface designers have dealt with 3D input problems for decades. Aish claimed that 3D input systems should be able to create and modify 3D geometry intuitively in order to interpret and evaluate the spatial qualities of a design directly [1]. But in most 3D modeling systems, command-based input and Graphical User Interfaces (GUIs) still dominate 3D Computer-Aided Design systems and have been optimized for 3D modeling. Keyboards and mice are also essential for users to type in or select commands. 3D manipulations are usually sequential and abstract. Users have to aggregate a serial of simple and abstract commands into a bigger modeling concept. It partially occupied mental resources so that designers are limited to act and think differently. There are always trivial steps before inspecting and editing 3D models that makes 3D modeling complex.

Ishii suggested a new concept to design interfaces integrating both physical and digital systems [2]. Designing Tangible User Interfaces (TUIs) is to create seamless interaction across physical interfaces and digital information. Interacting with TUIs can be more meaningful and intuitive than using traditional GUIs. iSphere also extends the concept of TUI with understanding user's hand behavior in order to provide relevant modeling functions at the right time. The orienting approach of a 3D view port into a 3D world-view has been a conceptually important idea since people started creating 3D computer graphics [3].

A desirable controller for a 3D environment might be a 6 degree of freedom device like a SpaceBall [4]. The space ball allows pressure forward aft side to side and up and down and rotation in X, Y, Z to control modeling. SpaceBall provides an intuitive 3D navigation experience with rotating a physical ball. But it also requires significant work with keyboard and mouse to map it into the control points and other desired function. It still takes time and steps to use physical navigation tool with mouse in order to complete a task like getting the right viewpoint and pulling the surface for 10 units along certain axis.

DataGlove usually works with 3D stereo glasses and positioning sensors. Users have to wear sensors and learn to map hand actions into manipulation commands. The advanced versions of DataGlove can provide 6DOF control and force feedback for users to model 3D under a rich immersive 3D environment. However, lacking physical references is easy to make users get lost. Working with stereo glass and wearable sensors for a long period of time may not yet be a good way. In [5], Zhai concluded that none of the existing 6DOF devices fulfills all aspects of usability requirement for 3D manipulation. When speed and short learning is a

primary concern, free moving devices are most suitable. When fatigue, control trajectory quality and coordination are more important, isometric or elastic rate control devices should be selected. In [6], Zhai suggested that designing the affordance of input device (i.e. shape and size) should consider finger actions. A 3D volume control system using foam resistance sensing techniques was demonstrated in [7]. With cubical input channels and pressure-based deformation, it could provide intuitive visual feedback for deforming shapes based on a physical cube. In [8], gesture modeling provided a novel way of interacting with virtual model directly, but the limitation of gesture language, fatigue problem, fidelity, and lacking of physical reference remained unsolved.

In [9], SmartSkin introduced a new way of bimanual interaction on the desktop. By using capacitive sensing and embedded sensors, users can naturally control digital information projected on the table by hands. In [10], Twister was presented as a tool of 3D input device using two 6DOF magnetic trackers in both hands to deform a sphere into any shape.

Learn from past experience in order to minimize the complexity of 3D modeling processes, this paper suggests that a physical modeling reference and the capability of using realistic hand interaction will enhance the intuitive experience of 3D modeling. Low-level operations of commands are time-consuming and costing extra efforts to complete a task in a 3D environment. A user does not have direct feedbacks from command-based manipulation and has to break concepts into trivial steps. The fragmented metal views and visual representation should be coupled in order to give designer expressive ways to model intuitively. iSphere is able to simplify the mappings between low-level manipulation commands and modeling concepts, such as pushing and pulling 3D geometries and viewpoints..

3. INTERACTIVE TECHNIQUE

iSphere is a physical reference for controlling the virtual model. This interface also enables the bi-manual interaction to shape 3D models through bare hands. The input interface is a hand-held dodecahedron which has 12 facets. Each facet is designed as a capacitive electrode that detects the distance from the human body. This dodecahedron takes inputs from 12 capacitive sensors and maps them spatially into a meta-model which controls any 3D model as the starting shape. This meta-model is the data structure of the dodecahedron which stores the deformation data from the sensors. This meta-model also stores interaction history into a time dimension for playing back the modeling history. As shown in Figure 2, when the hands

are away from the sensors from 1 to 6 inches, it considers this action is 8 degrees of 'pull'. When the hands are very close (< 1 inch) or pressing, it considers the user is denting the model.

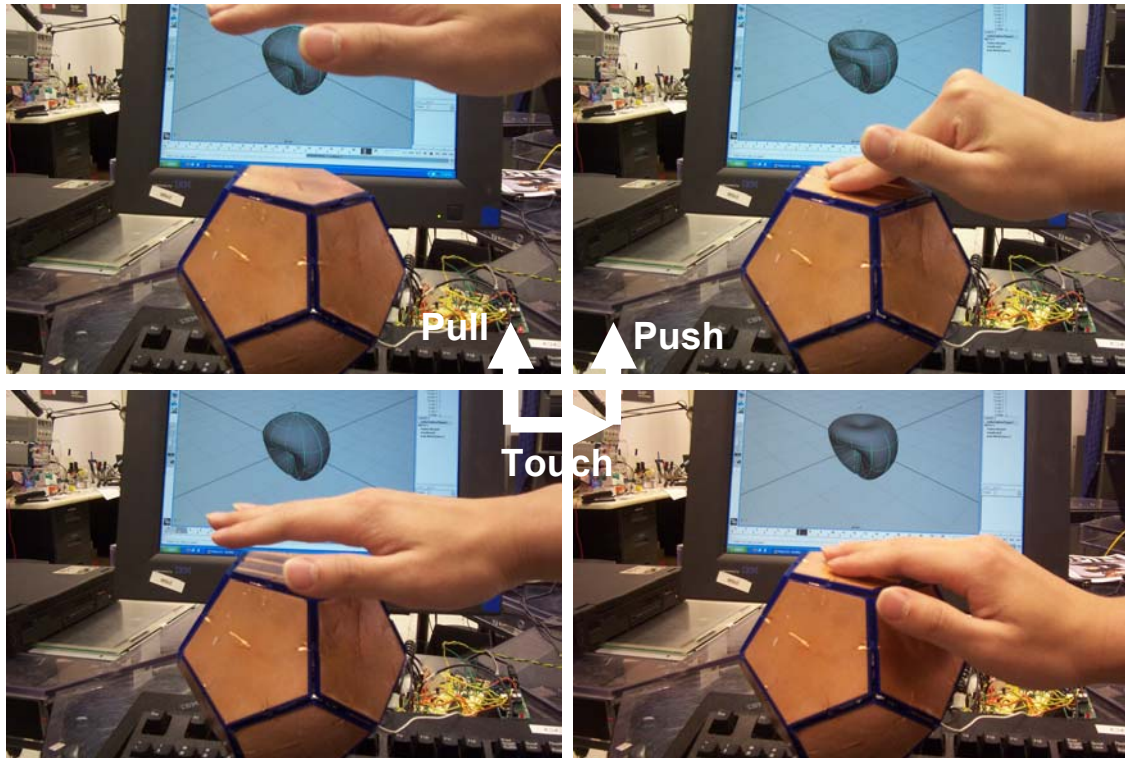


Figure 2: High-level 3D modeling commands, like pull, push and touch.

3D users should be expected to consume more cognitive load on designing rather than modeling. We propose iSphere acting as a hand sensor knowing about levels of actions, like hand positions, touching, pushing and twisting actions. In most 3D modeling systems, keyboards and mice are good for command-executing and mode-switching. However, it still can't allow us to perform an editing command by a single and intuitive action. We claim that making the interaction more realistic can enhance the experience of 3D modeling.

3.1. Realistic Interaction

iSphere has been used with an editing mode for modeling 3D geometries and an inspecting mode for navigating 3D scenes. The natural mapping for an enclosed object is to map the dodecahedron to pulling on and pushing in on the surfaces as though it were clay. Natural hand actions are mapped to the modeling commands, such as pushing multiple facets to squeeze the 3D model on that direction, as shown in Figure 2(a,b,c,d). Visual feedback is provided in 3D software responding the 3D warp effect like playing with

virtual clay when a user's hand is attempting to stretch the 3D object. In the inspecting mode, it acts as a proximity sensor which can detect the hand positions around the device. It is connected to the 3D software that rotates the corresponding camera viewpoint when a hand approaches the surface, as shown in Figure 3 (e,f,g). The 3D model can automatically get oriented when a user touches anyone of the surfaces. To switch the editing and inspecting mode, a functional button was installed on the desktop which allows users to switch between them by touching it or leaving it.

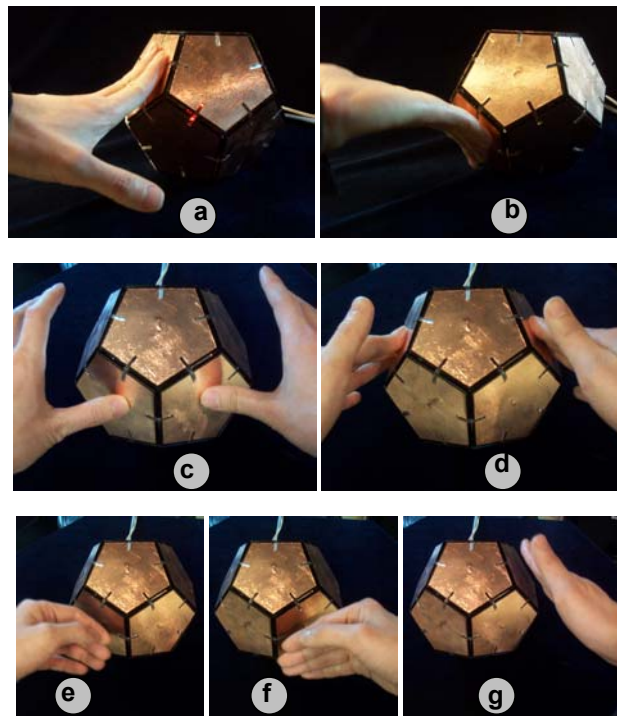


Figure 3: Hand movements as metaphors for editing and inspecting 3D scenes as realistic interaction

3.2. Play and Build

We propose a top-down 3D modeling approach that allows designers to play and build 3D models and develop their concept directly in order to reduce the cognitive load of fragmented design mode and modeling mode. Making a 3D model requires visualizing, re-visualizing and acting. It involves more mental activities of planning how to do instead of what to do. Having a design goal and shaping it into 3D objects involves a series of mode-switching activities. The processes are usually trivial, disruptive, and have little relation to design. Designers designed a 3D shape and then switched to the modeling mode. Obviously, designing and decomposing shapes into sequential machinery commands are two totally

different cognitive behaviors. This bottom-up approach limits the diversity of design outcomes and the expressiveness of 3D representation during the early design stage.

4. IMPLEMENTATION

iSphere is a physical interface for 3D input and also capable to perform 3D manipulation in modern CAD systems. iSphere hardware consists of physical materials, capacitive sensors and a low-cost circuit board. Modern CAD systems can be extended by their APIs to make iSphere available for an external input device.

4.1. Physical Interface

iSphere dodecahedron is made by acrylic. To create this device, a laser cutter was employed to make a foldable pentagonal surface. This was assembled with fitting pieces that snap it together. A circuit board which is incorporated a PIC16F88 microcontroller and a RS232 serial interface was embedded in the device. A microcontroller is used for getting the digital inputs from the sensor and output the signals to the serial port to a PC. Each side of the dodecahedron is a plastic acrylic piece, designed, with a copper backing and foam. Each of them is capable of sensing eight degrees within six inches above the surface of the pentagon when a hand is placed over it. Push command will be triggered if hands are closer to the surfaces. Pull commands will be triggered if hands are away from the surfaces. Capacitive sensors are connected in parallel into multiplexers are able to detect the proximity of hands from twelve different directions. For long-distance proximity sensing, we use a transmitter-and-receiver setting in the capacitive sensing circuit.

4.2. Software Architecture

The physical interface of iSphere maps the analog input signals into a meta-model in order to drive manipulations over the target 3D object. We utilized the Alias|Wavefront Maya 6.0 C++ API (Application Programming Interface) and implemented the functions into an iSphere Plug-in. It provides us a more flexible environment to design the iSphere system. The plug-in can be loaded automatically in the command prompt in Maya.

The software architecture can be described as a flowchart, as shown in Figure 4. First, the hardware of iSphere connected to the RS232 Serial Interface. Second, a meta-sphere maps raw data into a data structure in order to control any 3D object. MEL (Maya Embedded Language) is handy for describing 3D

modification. MEL also takes great advantages in its Hypergraph interface to easily apply 3D modification functions by drawing relationships of data flows.

The software architecture also reserves flexibility to upgrade in the future. New functions can easily be added by insert new codes or nodes into the system. Another advantage is when iSphere provides more commands, switching from different commands can be done easily by connecting the links between different nodes. iSphere is able to manipulate 3D mesh-based model in Alias|Wavefront Maya, 3DS Max or Rhino.

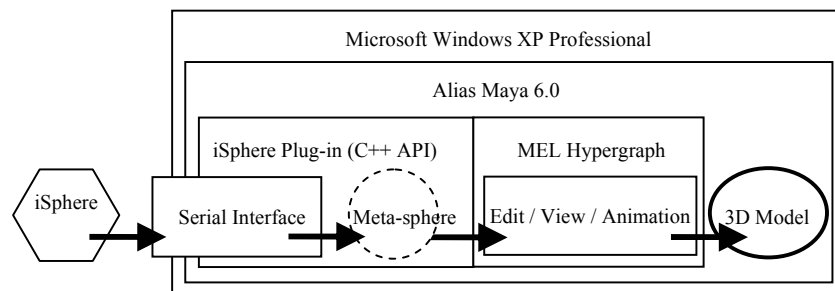


Figure 4: Software Architecture of iSphere

5. EXPERIMENT

A pilot experiment was conducted to examine potential problems before formal evaluation which contained four 3D modeling tasks. The purpose of the experiment is to study how novices and experts can adapt 3D input techniques in different input device. In our definition, experts mean designers who are familiar with traditional 3D modeling interface, while novices mean designers who are not familiar with 3D modeling interface. Our hypothesis is that when working in traditional modeling interfaces, the performance of experts is much better than novices, not because they have more knowledge of building models, but because they are familiar with employing different commands to reach the final state. If there is a more intuitive and efficient input interface to modify models which can eliminate the gap, novices should have similar performance as experts.

5.1. Experiment Set-up

A desktop 3D modeling environment was set up in the experiment. It consists of an IBM Graphics Workstation, 19" LCD display monitor, Alias|Wavefront Maya 6.0 with iSphere plug-in software, standard keyboard, mouse, and the iSphere device, as shown in Figure 5. To improve proximity sensing, subjects

were asked to sit on a chair where a mental strip attached on the edge providing a harmless reference signal (5Volts-20kHz) which let the user become an antenna. Capacitive sensors can sense hands up to six inches above the surface. The 12 facets were also covered with foam in order to provide instant feedback that allows user to feel the distance from hands and the surface. The iSphere was installed on a soft-foam base to provide arm supports for the user. In order to enhance the 3D visualization, all 3D objects were rendered in shading mode.

Experimental Condition

To examine our hypothesis, both novices and experts were asked to perform some tasks in iSphere. We didn't hold the condition of using keyboards and mice, because we discovered that in simple tasks such as move or deform objects. Experts usually perform them in routines, which mean they will do those tasks with almost the same procedure and actions. In order to analyze knowledge of how to do each task, we decided to employ the KLM-GOMS (Keystroke-Level Model GOMS) method [10] to calculate approximate time that experts need to accomplish tasks. This data will be compared with the other two conducted by both novices and experts in iSphere. In this pilot study, we allowed subjects to hold their tasks and re-start again until they felt confident in tasks.

Experimental Task

Four 3D modeling tasks were designed in this study. Each task represents a typical 3D surface shaping procedure involving a series of view and edit commands. Subjects were asked to do four tasks in a sequence. At the beginning of each task, subjects started with a new scene with a default 3D sphere appeared in the middle of the screen. In the first task, subjects were asking to pull the top surface up to 3 units. The second task is to expend the bottom of the sphere to 3 units. In the third test, subjects were asked to make an apple. The final task is to make any shape in five minutes.

Experimental Design

Subjects were divided into two groups by their experience in 3D modeling. One was the novice group. The other one was experts group. Both of them used iSphere as the input device. This condition was given about 15 minutes of exposure, which comprised a pre-test questionnaire, a short demonstration, and four tasks. In the first two tests, subjects were given 3 minutes to finish simple modeling tasks. In the following

two tests, subjects had to finish two modeling tasks in 5 minutes. Each subject was asked to fill the post-test questionnaires after four tasks. Think-aloud method is used in the user study in which a subject has to explain each action regarding what task he/she wants to do and how to deal with the task during the whole modeling process. The advantage of the method is to help us know well about the very detail actions the subject took in the study.

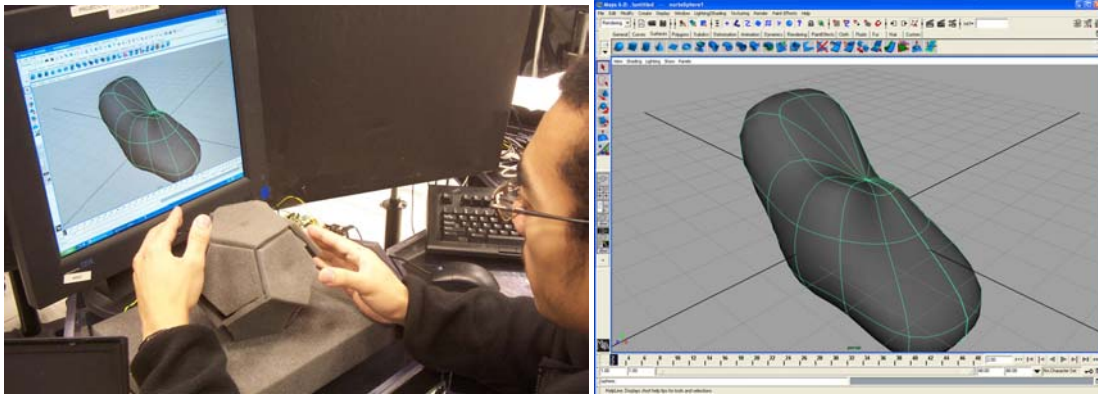


Figure 4: A pilot study of iSphere (left) and a shoe shaped by a subject (right)

5.2. Experimental Results and Discussion

Six volunteers were recruited in this pilot study. Four of them had no previous experience with Maya. Two of them had intermediate level of skills in Maya. Their ages ranged from 17 to 27, with a median of 21.2. All subjects were right hand dominated. All subjects finished the four tasks.

Analysis of the Overall Results

During the study, we recorded both sound and images for further analysis. Although the users just performed simple tasks which were to move the upper points of a sphere to reach the top of the Maya modeling window, the modeling process could still be decomposed into several steps including problem solving and moving actions. For instance, the user may pause for a while to think about how to solve problems then perform actions. We transcribed all steps of the problem solving process and calculate how much time the user spent on each action. The KLM-GOMS analysis techniques can calculate time consuming of routine works.

Compared to mentally problem solving process, procedural actions takes greater part of the time. When a user uses mouse and keyboard to perform modeling task, he/she has to act many trivial actions, for instance,

to move mouse forth and back and click buttons, as well as has to move his/her eye sight many times to search different commands on the window. However, when using iSphere, a user can easily focus on interacting with it and the model in 3D software without a lot of trivial actions; although, the interaction is more inaccurate than using mouse and keyboard.

We used KLM-GOMS to analysis subjects who had intermediate experience in Maya and then compared to novice who used iSphere. Expert users used combination of shortcut keys and mouse so that they completed the tests effectively and precisely. In the first two tasks, they presented the same routine to reach the goal state. Before they started moving, they spent 3 to 5 seconds to think over possible solutions. To summarize their actions during the tasks, they spent much time and actions to move the mouse cursor to reach icons or menus on left and top of the screen. Each movement may cost 1 to 1.5 second and all movements cost around 20 to 25 seconds depending on different tasks. The next is selection, they selected corresponding CVs (control Vertex) professionally and move them to appropriate positions that the tasks asked for. The selecting and moving actions cost around 5 to 7 seconds. Clicking mouse buttons cost the shortest time in the experiment, but was the most frequently action. Each click cost around 0.2 second and 10 to 15 times.

- Mentally prepare 1.35 second.
- Move cursor 1.10 second.
- Click mouse button 0.2 second.
- Press/release mouse button 0.1 second.
- 3 novices using iSphere: 1)Pull-up 8.6 second, 2)Expend 12.5 second.
- GOMS: 1)Pull-up 10 second, 2)Expend 20 second.

Comparing to subjects using keyboard and mouse, the results conducted by subjects using iSphere is relatively simple t. According to our demonstration before the experiment, they all could well know how to reach the goal state by controlling corresponding facets. Therefore, all of them spent less than 2 seconds to think over the solution, before they started. The novice group spent average 8.6 seconds on the first task and average 12.5 seconds on the second task. In the third and fourth task, we weren't able to calculate using KLM-GOMS. Although most of them spent much time to modify the model back and forth, they finished the two tests with shorter than those intermediate Maya users.

The preliminary result shows some important phenomena between using mouse and iSphere. iSphere exposes controls in a spatial way allowing users to directly manipulate the surface. It takes fewer steps than selection such controls from a tool bar and associations. Using iSphere can also reduce time consuming and actions to make simple models. Furthermore, iSphere allows users to move multiple facets at the same time. Comparing to mouse users, iSphere combines selection, direction and commands. iSphere can measure pressure and pulling at a 4 bits resolution. Currently, iSphere did not have the speed and accuracy of control that a mature analog input device gives. It is not able to perform actions precisely. In the experiment, users spent almost half of the time to move back and forth, because they cannot shape the model exactly as they wanted.

6. DISCUSSION

Freehand sketching is the most direct and interactive way for designers to shape ideas. The goal of iSphere is to reduce the mental load of modeling 3D geometry. We propose that freehand 3D modeling can off-load low-level manipulations from modeling processes. These processes should leave more rooms for designers to think and evaluate the shape, not involving too many interruptions from the manipulation commands. This paper presents a 3D input device that uses a physical modeling reference and the capability of using natural hand interaction to enhance the intuitive experience of 3D modeling. By using the information collected from its user, the system can simplify 3D modeling user interface. Using iSphere, the user can pay more attention to and focus on what's in their mind and hands, not the menus or commands.

iSphere could change the way 3D designers work with abstract commands into natural hand interaction and intuitive 3D modeling processes, but lacking of fidelity to make detail modification is the main problem for this interface. This physical device also doesn't require wearing any sensors.

Designing new inputs to 12 surfaces is a complex goal. It has to do with choosing, adding and making sensors with enough fidelity, choosing metaphors that match the device, and creating a transfer function that makes sense for what is it being used for. One may argue that iSphere is a specialized device for certain specific modes, while general modeling interface is designed for general input device. Mouse and keyboard are very good at mode-switching tasks. It's not fair to compare them in certain modes of modeling.

Using iSphere, in a sense, limits the ways to model. Users can only interact with this device by hand interaction, however, it can help users to finish specific task quickly. Using mouse and keyboard, on the other hand, has more freedom to perform jobs, but in most actions, it is time consuming. Both of them represent parts of our needs when modeling. Therefore, the next step of making new modeling tools may combine these two concepts. Future work will deal with making mappings robust across shapes, methods of creating models simply and improving algorithms for sensing control. To extend the analog 3D input approach using proximity sensor, we found there are several ways to go. For example, the modeling sequences can be recorded and playback. It can be used as a motion capture device to make 3D animation from realistic interaction.

Acknowledgements

We thank Rob Gens, Minna Ha and Elliott Pretcher for their UROP contributions.

References

1. Aish, R. 1979. 3D input for CAAD systems. *Computer-Aided Design* (1979), pp. 66-70.
2. Ishii, H., B. Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces between People, Bits, and Atoms. *Proceedings of the CHI 97*, ACM Press (1997), pp. 234-241.
3. Van Dam, Andries. 1984. Computer graphics comes of age: an interview with Andries Van Dam. *Communication of the ACM*, Vol 27 Issue 7, 1984, pp.638-648.
4. Zhai, S., Kandogan, E., Smith, B., Selker, T. 1999. In Search of the "Magic Carpet", Design and Experimentation of a 3D Navigation Interface. *Journal of Visual Languages and Computing*. Vol 10, No.1, 3-17, February 1999, pp.3-17.
5. Zhai, S. 1998. User Performance in Relation to 3D Input Device Design. *In Computer Graphics* 32(4), November 1998, pp. 50-54
6. Zhai, S., Milgram, P., Buxton, W. 1996. The Influence of Muscle Groups on Performance of Multiple Degree-of-freedom input. *Proceedings of the CHI 1996*, pp. 308-315.

7. Murakami, T., Nakajima, N. Direct and Intuitive Input Device for 3-D Shape Deformation. *Proceedings of the CHI 1994*. pp. 465-470. 1994.
8. Gross, M.D., Kemp, A., Gesture Modeling: Using Video to Capture Freehand Modeling Commands, in CAAD Futures, 'CAAD Futures 2001', pp. 271-284
9. Rekimoto, J. 2002. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. *Proceedings of the CHI 2002*, ACM Press (2002)
10. Llamas, Ignacio., Kim, B., Gargus, J., Rossignac, J., Shaw, C.D. 2003. Twister: a space-warp operator for the two-handed editing of 3D shapes. *ACM Transactions on Graphics (TOG)*, Vol 22 Issue 3, 2003, pp.663-668.
11. John, B., Kieras, D. 1996 The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *ACM Transactions on Computer-Human Interaction*, Vol. 3, No. 4, pp 320-351.