Knowledge Engineering and Ontologies for Object Manipulation in Collaborative Virtual Reality

Manolya Kavakli
Visor (Virtual and Interactive Simulation of Reality) Research Group, Virtual Reality Lab, Department of Computing, Macquarie University, North Ryde, Sydney, Australia

Keywords: Collaborative Engineering, Virtual Reality, Object Manipulation.

Abstract: This project describes an ontology for a collaborative engineering task. The task is to take apart an interactive 3D model in 3D space using virtual reality and to manipulate an object. The project examines a virtual environment in which two engineers can perform a number of tasks for manipulating object parts controlling a wiimote inside an immersive projection system. The interface recognizes hand-gestures of the engineers, pass commands to a VR modelling package via a gesture recognition system, perform the actions on the 3D model of the object, generating it on the immersive projection screen. We use retrospective protocol analysis for knowledge engineering and ontology building analysing the cognitive processes.

1 INTRODUCTION

"A major problem in the design and application of intelligent systems is to capture and understand: the data and information model that describes the domain; the various levels of knowledge associated with problem solving; and the patterns of interaction, information and data flow in the problem solving space. Domain ontologies facilitate sharing and re-use of data and knowledge between distributed collaborating systems." (Ugwu et al, 2001). We need ontologies for the following reasons:

- To have shared understanding of the topic
- To enable reuse of domain knowledge
- To make domain assumptions explicit
- To analyze domain knowledge

Ontologies have become core components of many large applications yet the development of applications has not kept pace with the growing interest (Noy and McGuinness, 2001). This paper describes an ontology for collaborative engineering platforms using virtual reality and knowledge acquisition techniques. The paper shows that a common ontology facilitates interaction and negotiation between engineers (agents) and other distributed systems. The paper discusses the findings from the knowledge acquisition, their implications in the design and implementation of collaborative virtual reality systems, and gives recommendations on developing systems for collaborative design and object manipulation in engineering sector.

2 OBJECT MANIPULATION IN VIRTUAL REALITY

The first effort on object manipulation can be traced back to late 70's. Parent (1977) proposed a system which was capable of sculpting 3D-data. The significant problem solved within the system was hidden-line elimination by choosing planar polyhedral representation. Parry (1986) developed a system using constructive solid geometry (CSG) that can only carry out a number of simple sculpting tasks using traditional devices such as mice and keyboards as input medium. Coquillart (1990) developed a sculpting system using 3D free-form deformation which was more capable of generating arbitrarily shaped objects in comparison to Parry's system. Mizuno et al (1999) built a system for virtual woodblock printing by carving a workpiece in the virtual world using CSG. Recent developments in VR led to a number of important innovations. Pederson (2000) proposed Magic Touch as a natural user interface that consists of an office environment containing tagged artifacts and

Kavakli M..
Knowledge Engineering and Ontologies for Object Manipulation in Collaborative Virtual Reality.
DOI: 10.5220/0004170202840289
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2012), pages 284-289
ISBN: 978-989-8565-30-3

wearable wireless tag readers placed on the user's hands. Bowman and Bilinghurst (2002) attempted to develop a 3D sketchpad for architects. However, the 3D interface with menus did not respond to the expectations of architects, and there was a need for a greater understanding of users' perceptions and abilities in 3D interface development. Salomon (2005) introduced non-uniform rational b-spline (NURB) deformation to Lau's Vsculpt system (2003) with the integration of CyberGloves. In this system, the users could generate arbitrary-shaped objects by manipulating a number of control points that required the users to learn the parametric control techniques. Jagnow and Dorsey (2006) applied haptic displacement maps to process the graphics data in an efficient manner in a virtual sculpting system. In this system, models could be described by a series of partitioned local slabs, each representing a vector field. However, haptic displacement map could not be applied to the dynamic scenes that change frequently. In spite of these innovations, there are still a number of core questions waiting to be answered. These are as follows:

- How can we develop a robust method for object manipulation, configuring complex engineering and design systems by using VR technology?
- How can we support the communication between geographically separated engineers and the CAD/CAM model of the product?

2.1 Collaborative Engineering in VR

In this project, we have developed a collaborative engineering platform to investigate the nature of shared information. The project examines a virtual environment in which two engineers can perform a number of tasks for manipulating object parts controlling a wiimote inside an immersive projection system (Figure 2). The engineers wearing stereoscopic goggles have the benefit of being able to work with a stereo image. The interface recognizes hand-gestures of the engineers, pass commands to a VR modelling package via a gesture recognition system, perform the actions on the 3D model of the object, generating it on the immersive projection screen. Wiimote (Wii Remote) is the primary controller for Nintendo's Wii console. The main feature of the Wii Remote is its motion sensing capability, which allows the user to interact with and manipulate items on screen via movement and pointing through the use of accelerometer and optical sensor technology.

The expected outcomes of this study are:

- Novel human computer interaction techniques; and
- Ontologies demonstrating the structure of cognitive actions of engineers in object manipulation.

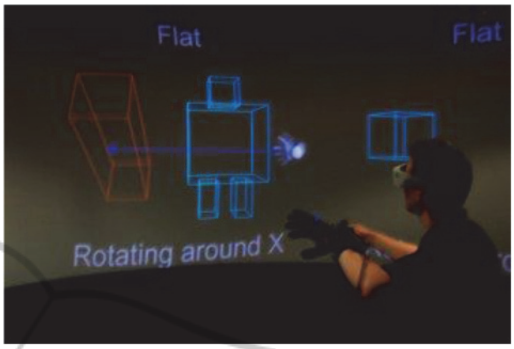

Figure 1: Collaboration in Co-DeSIGN.

2.2 Requirements Analysis

The objective of this project is to design and build a collaborative platform (Co-DeSIGN) for disassembling a mechanical product using Virtual Reality technology. Each task the mechanical engineers to perform using this collaborative platform refers to a module in the system architecture. The main tasks and modules are specified as follows:

2.2.1 Explore and Navigate

This module manages the exploration and navigation in the virtual world. The user is expected to explore a 3D object and move around it. The user must control a cursor to perform different actions to complete the task. The actions are as follows:

- Visualisation: The user must see, perceive, and investigate the product. We must create a point of view to represent the sight of the engineer.
- Navigation: While the user is able to move around the product, he must be able to zoom in/out, rotate, and translate his point of view.
- Interaction: The user must be able to control the cursor using wiimote. The user must have a control over the depth of the cursor, getting closer or far from the product. The icon of the cursor must reflect the changes depending on the action performed by the user.

2.2.2 Disassemble

This module manages the disassembling process. We have categorised all of the actions the user needs to perform in order to disassemble the given mechanical product. It is obvious that in reality the number of possibilities and tools a technician can use is quasi infinite. In our application, we need metaphors to realize a subgroup of these actions and possibilities. In future, we may be able to simulate more actions by adding new modules to this application. We hope to be able to integrate a variety of mechanical links between various parts of the product. The actions relevant to this module are specified as follows:

- Movement: The user must be able to move various parts of the product.
- Selection: The user needs to select various parts.
- Integration or Disintegration: The user must be able to perform specific actions to the selected parts, such as mounting parts with a screwdriver or to take apart.
- Collision detection: The application must manage the collision between various parts.
- Logical decision-making: The application must manage the disassembling scenario with a logic engine. For example, the user is not allowed to perform any tasks at anytime in the scenario (he may need to remove the base first, to disassemble the parts above).
- Position handling: Finally, the application needs to manage various states of the parts and know their positions as well as where they belong to as the part of the product.

2.2.3 Collaborate

This module manages the collaboration of the users with each other. The engineers must be able to collaborate to perform the tasks together. This involves not only the communication with each other, but also following the partner's task performance. Some tasks may require the performance of a specific action at the same time to complete it. The number of different tasks the team must perform together depends on the chosen scenario. We have limited the number of users of this collaborative platform with two. Therefore there is no use in building a complex network to manage the collaboration in our case. A simple network where an engineer can communicate with another without a central server is sufficient.

We defined the actions for the collaborative process must as follows:

- Processing the State of Actions: The application must keep a record of actions of both users, perform these actions subsequently on the product, and inform the users about what the other is doing or has done in a timely manner.
- Speech Processing: The application must allow both users to speak with each other.
- Task Processing: Specific tasks must be completed only if both users have performed the right actions at the right time.

3 EXPERIMENTATION

System set up includes two immersive projection systems and two wiimotes. The 3D model is generated in Catia 3D modelling software, and transferred to Vizard virtual environment. Following this, we conduct pilot studies to test the system.

In this application the goal to be reached by the team is not to disassemble a new product, but to disassemble a well-known product the fastest and the most efficient way possible. The process that leads to task completion depends on the product the team must disassemble. There are instructions to follow and there is often a unique way to disintegrate a product. Users must follow a specific order. The assembly given to the user is composed of a base with two slide rails and a moving base beard by two bearings as shown in Figure 2. We conducted pilot experiments and have had 2 engineers to test the system. They have used the interface (Co-DeSIGN) to disassemble a product in VR collaborating at a distance.

In future, we plan to use 20 mechanical engineers to test the system in ENSAM, France and at the VR LAB, Australia. All participants will be videotaped, while performing the task in a design session of 15 minutes in duration. Having completed the task, the model disassembled by the engineers is displayed and participants are viewed to video records of their own engineering session. Then, we ask them to interpret the reasoning of their hand gestures, speeches, and motor actions. We also give them a questionnaire to assess the quality of the system they have used in comparison to the traditional methods.

Thus we collect the Retrospective engineering protocols. We plan to use matrix analytic methods to give a probability distribution of paths of consecutive actions in cognitive processes. To specify usability requirements in system development, it is important to understand how humans perceive the world, how they store and process information, how they solve problems, and how they physically manipulate objects. We use the Task Analysis method (the study of the way people perform tasks with existing systems) to model the system. This involves not only a hierarchy of tasks and subtasks, but also a plan that consists of the order and conditions to perform subtasks. Knowledge-based task analysis includes building taxonomies of objects and actions involved.

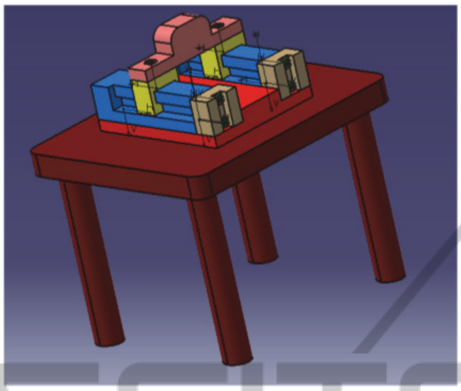

Figure 2: The assembly used in the experiments.

4 METHODOLOGY

The methodology is based on the simulation of cognitive processes in object visualisation, drawing, and manipulation. Kavakli et al. 1998 conducted a series of experiments on artists' free hand sketching and found that objects are drawn 90% part by part. Later, they explored the nature of the design process (Kavakli et al., 1999) and found that there was evidence for the coexistence of certain groups of cognitive actions in sketching (Kavakli and Gero, 2001a), which resembles mental imagery processing. Investigating the concurrent cognitive actions in designers, they found that the expert's cognitive actions are well organized and clearly structured, while the novice's cognitive performance has been divided into many groups of concurrent actions (Kavakli and Gero, 2001b, 2002, 2003). This structural organisation can be exploited to model an intelligent system to be used for object manipulation, especially in teams involving novice and expert engineers. VR technology in this paper refers to the interface that enables the user to interact with a VE. It includes computer hardware in the form of peripherals such as visual display and interaction devices used to create and maintain a 3D VE. A VR interface provides immersion, navigation, and interaction. The project defined in this proposal examines a VE in which an engineer can manipulate the parts of a 3D object using a pointer, motion trackers, and stereoscopic goggles. As stated by Kjeldsen (1997), hand gestures occur in space, rather than on a surface, consequently positioning is inherently 3D. This can obviously be an advantage when developing gesture-based object manipulation systems. The usability of 3D interaction techniques depend upon both the interface software and the physical devices used. However, little research has addressed the issue of mapping 3D input devices to interaction techniques and applications.

Our approach is to investigate 3D object manipulation in collaborative engineering, hand gestures and design protocols as the language of the engineering process. We focus on the structures in visual cognition and explore bases for rudimentary cognitive processes to integrate them into an intelligent VR system. The results provided by protocol analysis studies are used to construct a user interface for both visual cognition and hand-gesture recognition. Retrospective protocol analysis is influential in understanding visual cognition in the engineering process.

4.1 Retrospective Protocol Analysis

Retrospective Protocol Analysis involves following stages:

- *Identifying the part–based structure of the object:* The completed model is decomposed into parts to be used as a reference for the coding of related cognitive actions.
- *Interpretation of video protocols:* We transcribe the verbal protocols of designers from video records for the analysis of engineering protocols.
- *Segmentation of design protocols:* Transcribed engineering protocols are divided into segments. A cognitive segment consists of cognitive actions that appear to occur simultaneously.
- *Coding:* We code cognitive actions of designers using a coding scheme developed by Suwa et al (1998). In the coding scheme, the contents of what engineers see, attend to, and think of are classified into four information categories, namely: depicted elements, their perceptual features and spatial relations, functional thoughts, and knowledge. There are four modes of cognitive actions (Kavakli et al., 1999): physical (drawing actions, moves, looking actions), perceptual, functional, and conceptual (goals). Each mode has a number of subgroups.

In the sample (Figure3), the goals of bisecting the building and splitting the space, triggers a number of perceptual actions driven by drawing a circle (Dc: create a new depiction). Perceptual actions about Attention to relations between the object features (Prn1 and Prn2: create or attend to a new relation) are dependent on Drawing a circle (Dc) and Looking at (L1) previously drawn depictions (line 67). One of these perceptual actions (Prn1) triggers the Discovery of a space (Psg: discover a space as a ground). We will particularly focus on correlations between the cognitive actions coded "Dc, L, Prp, Prn, Fo, Fn" as the path to discoveries, based on Kavakli and Gero (2002). Our task is to mainly focus on motion tracking, as well as the relationship between the physical (especially moves) and perceptual actions. We improve the category of physical actions (moves) in the existing coding scheme.

Segment no: 248	so I am going to have to segment this a little bit. Something has to be here and something back here. And I am not going to bisect the main space.				
Action type	index	class	Description (where, of what, among what?)	Dependency	
				index	On what
Drawing Dc	new	Circle 3			
Looking L1	old	Line 67			
Moves					
Perceptual Psg Prn1 Prn2	New New new	i-space l-relation g-relation	The rest space spatial rel (separate): the two spaces spatial rel (included): the new space is on the side of the building	New/new New/old	Dc, Psg Dc, L1
Functional					
Goals					
type	content		Source Seg/type	Trigger what?	
Type 2 Type1.3	I am not going to bisect the main space of the building I am splitting the building on the side, not in the center		256	Type1.3 Prn1, Prn2	

Figure 3: Coded cognitive segment.

4.2 Ontology Development

The Artificial-Intelligence literature contains many definitions of an ontology; many of these contradict one another. In this paper, similar to Noy and McGuinness (2001), we consider an ontology as a formal explicit description of concepts in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concept (slots (sometimes called roles or properties)), and restrictions on slots (facets (sometimes called role restrictions)). An ontology together with a set of individual instances of classes constitutes a knowledge base. In reality, there is a fine line where the ontology ends and the knowledge base begins.

In this project, our aim is to lay fundamentals for an ontology development for gesture recognition systems to be used by an intelligent user interface. Currently, we are working on the development of an ontology for gestures. We need to account for a wide-range of physical actions (hand gestures) as described by Mulder (1996):

- Goal directed manipulation: Changing position (lift, move, heave, raise, etc.), Changing orientation (turn, spin, rotate, revolve, twist), Changing shape (mold, squeeze, pinch, etc.), Contact with the object (grasp, seize, grab, etc.), Joining objects (tie, pinion, nail, etc.).
- Indirect manipulation: (Whet, set, strop)
- Empty-handed gestures: (twiddle, wave, snap, point, hand over, give, take, urge, etc.)
- Haptic exploration: (touch, stroke, strum, thrum, twang, knock, throb, tickle, etc.)

In the design of a hand-gesture based interface, we plan to address the following issues (Kjeldsen,1997): object selection, action selection (pose and position, pose and motion, multiple pose), action modifiers and rhythm of interaction (syntax of hand gestures). We will explore the semantics of pause (action stops then continues), comma (action completed and repeated) and retraction (another action). Assuming that hand gestures generally have a Prepare-Stroke-Retract cycle, we develop a vocabulary of hand gestures such as:

Prepare/Pose/Pause/Select/Retract,
Prepare/Pose/Comma/Pose/Stroke/Retract.

The following syntax may be used to address a hand-gesture interface and phrase can be further decomposed to implement the issues described above:

Gesture-> Prepare <Stroke> Retract
Stroke -> [Phrase Comma] Phrase*
Phrase -> [Pose|Motion Pause] Pose|Motion*

In this paper, we discuss general issues to consider and offer one possible process for developing an ontology. We describe an iterative approach to ontology development: we start with a rough first pass at the ontology, and then revise and refine the evolving ontology and fill in the details.

5 CONCLUSIONS

In this project, we build a hybrid reality system, where the user's hands form dynamic input devices that can interact with the virtual 3D models of objects in a Virtual Environment (VE). In the current phase, we have been trying to complete the gesture ontologies to feed the gesture recognition system and then we will start the experimentation with a large number of participants. As stated by McNeill (2006), gestures can be conceptualized as objects of cognitive inhabitance and as agents of social interaction. Inhabitance seems utterly beyond current modelling, but an agent of interaction may be modelable. Coordinative structures in collaborative engineering may help explain the essential duality of language which is at present impossible to model by a computational system.

ACKNOWLEDGEMENTS

This project has been sponsored by the Australian Research Council Discovery grant DP0988088 to Kavakli, titled "A Gesture-Based Interface for Designing in Virtual Reality", and by two French government scholarships given to Stephane Piang-Song (2008) and Joris Boulloud (2009) to complete their internships towards the degree of MEng at the Department of Computing, Macquarie University.

REFERENCES

Bowman, D. and Billinghurst, M. 2002. Special Issue on 3D Interaction in Virtual and Mixed Realities: Guest Editors' Introduction. *Virtual Reality*, vol. 6, no.3, pp. 105-106.

Coquillart, S., 1990. Extended Free-Form Deformation: A sculpturing tool for 3D geometric modeling. In *SIGGRAPH '90*, vol 24, number 4, pages 187-196, ACM, August.

Jagnow, R., Dorsey, J., 2006: *Virtual Sculpting with Haptic Displacement Maps*. robjagnow.com/research/HapticSculpting.pdf, last accessed: 10th Jun 2006.

Kavakli, M., Scrivener, S.A.R, Ball, L.J., 1998. The Structure of Sketching Behaviour, *Design Studies*, 19, 485-517.

Kavakli, M., Suwa, M., Gero, J. S., Purcell, T., 1999. Sketching interpretation in novice and expert designers, *in* Gero, J.S., and Tversky, B. (eds), *Visual and Spatial Reasoning in Design*, University of Sydney, Sydney, 209-219.

Kavakli, M., Gero, J. S., 2001a. Sketching as mental imagery processing, *Design Studies*, Vol 22/4, 347-364.

Kavakli, M., Gero, J. S., 2001b. Strategic Knowledge Differences between an Expert and a Novice, Preprints of the 3rd Int. Workshop on Strategic Knowledge and Concept Formation, University of Sydney, 44-68.

Kavakli, M., Gero, J. S., 2002. Structure of Concurrent Cognitive actions: A Case Study on Novice & Expert Designers, *Design Studies*, Vol 23/1, 25-40.

Kavakli, M., Gero, J. S., 2003. Difference between expert and novice designers: an experimental study, *in* U Lindemann et al (eds), *Human Behaviour in Design,* Springer, pp 42-51.

Kjeldsen, F. C. M.,1997. *Visual Interpretation of Hand Gestures as a Practical Interface Modality*, PhD Thesis, Grad. Sch. of Arts and Sci., Columbia Uni.

Lau,R., Li, F., and Ng, F., 2003. "VSculpt: A Distributed Virtual Sculpting Environment for Collaborative Design," IEEE Trans. on Multimedia, 5(4):570-580

McNeill, D., 2006. *Gesture and Thought*, The Summer Institute on Verbal and Non-verbal Communication and the Biometrical Principle, Sept. 2-12, 2006, Vietri sul Mare (Italy), organized by Anna Esposito, http://mcneilllab.uchicago.edu/pdfs/dmcn_vietri_sul_mare.pdf (last accessed on 19.8.2012).

Mizuno, S., Okada, M., Toriwaki, J., 1999. Virtual sculpting and virtual woodblock printing as a tool for enjoying creation of 3d shapes. *FORMA*, volume 15, number 3, pages 184-193, 409, September.

Mulder, A., 1996. Hand Gestures for HCI, Hand Centered Studies of Human Movement Project, *Technical Report 96-1*, School of Kinesiology, Simon Fraser University, February.

Noy, N. F. and McGuinness, D. L. 2001. Ontology Development 101: A Guide to Creating Your First Ontology. *Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880*, March.

Parry, S. R., 1986. *Free-form deformations in a constructive solid geometry modeling system*. PhD thesis. Brigham Young University.

Parent, R. E., 1977, A system for sculpting 3-D data. In *SIGGRAPH'77*, volume 11, pages 138-147. ACM, July.

Pederson, T., 2000. Human Hands as a link between physical and virtual, Conf. on *Designing Augmented Reality Systems (DARE2000)*, Helsinore, Denmark, 12-14 April.

Salomon, D., 2005. *Curves and Surfaces for Computer Graphics*. Springer Verlag August 2005. ISBN 0-387-24196-5. LCCN T385.S2434 2005. xvi+461 pages.

Scrivener, S. A. R., Harris, D., Clark, S. M., Rockoff, T., Smyth, M., 1993. Designing at a Distance via Real-time Designer-to-Designer Interaction, *Design Studies*, 14(3), 261-282.

Suwa, M., Gero, J. S., and Purcell, T.: 1998. Macroscopic analysis of design processes based on a scheme for coding designers' cognitive actions, *Design Studies* 19(4), 455-483.

Ugwu, O. O., Anumba, C. J., Thorpe, A., 2001. Ontology development for agent-based collaborative design, Engineering, *Construction and Architectural Management*, Vol. 8 Iss: 3, pp.211 – 224.