3D INTERACTION ORIENTED OBJECT MODEL

Cheng Cheng and Jia Lu
Department of Computer Science, Beijing Institute of Technology
Haidian Zhongguancun South No.5 Street, Beijing, China

Keywords: Object Model, Human Computer Interaction, Virtual Manufacturing, Perception, Behavior.

Abstract: Human computer interaction (HCI) is the most urgent challenge to virtual environment based manufacturing today. Traditionally *VRML/X3D* standards are used as the main object models in virtual environment (VE), and collision detection serves as the exclusive perception mechanism in such an environment. It is observed that these models are difficult to satisfy the requirement of 3D human computer interaction. This paper tries to provide a new object model scheme for virtual assembly which is a typical virtual environment application that has the most critical HCI requirements. The work of this paper includes introduction of object scheme, new perception mechanism, and object behaviour abstraction. The provided object model is applied and validated in virtual assembly prototype system.

1 INTRODUCTION

Virtual manufacturing (VM), which refers to virtual environment based manufacturing, has potential advantage for mechanical product design and validation. It is essentially a kind of critical real-time interactive 3D graphical computing, with the most urgent HCI requirement.

(Drieux et al. 2006) discussed the need of shape adaptation for virtual environment applications and gave a description of the process that is used to acquire the models. (Figueiredo 2007) briefly reviewed the collision detection algorithms used in virtual assembly. From this paper it can be concluded that collision detection is still the unique perception method used in VM. (Seo et al. 2004) discussed the object model for virtual environment, and used state chart to define the object's discrete event and behavior. (Lee et al. 2002) gave a behavior design method in virtual environment for game and animation, but the behaviors still referred to predefined motions as the same as the "route" in *VRML/X3D* model. (Callison 1995) presented the time-sensitive object (TSO) model structures systems making object values extend to object histories in which a sequence of time-constrained values describe the evolution of the object over time. (Pons 2002) proposed a method that promoted and incorporated temporal constraint specifications to the class/type-level in a compatible fashion with the object-oriented framework. (Shoyo et al.) studied how to abstract human's motion. The author discussed a primitive motion, whereas complicated object behavior abstraction had not been investigated.

2 STATIC CONTENTS OF VIRTUAL OBJECT

The fundamental static contents of virtual object should be constructed in advance. The objects in manufacturing are initially formed by features, but these features are not used in virtual environment because the data models used by the design environment and the virtual environment are quite different. 3D interaction oriented virtual object in figure 1 shows four static aspects of the virtual object which abstract all the necessary data for dynamic mechanisms.

Instead of only using *VRML/X3D* models in VE, the authors retrieved geometric features through parsing the *STEP* files and reconstructed a description of the part and the features. The interaction oriented attributes for both the part and the feature include the shape type, the dimensions and some auxiliary features, such as '*Origin*', which refers to the barycenter of the object; '*Major*', '*Minor*', and '*3thAxis*', which refer to the perpendicular axes of the local coordinate of each

Cheng C. and Lu J. (2010).
3D INTERACTION ORIENTED OBJECT MODEL.
In *Proceedings of the International Conference on Computer Graphics Theory and Applications*, pages 393-396
DOI: 10.5220/0002825103930396

object. These attributes are retrieved for both part itself and some of the assembly features of the part. The feature class definition thus is given in the object model.

The *VRML* shape node is linked with the object, and mappings between the features and different segments of the *VRML* shape surfaces are also constructed. We use *VRMLref* as a reference to the *VRML* file of an object. A set of auxiliary features for every geometric feature are established and the corresponding appearances are *VRML* shapes referenced by *AuxFeaRef*s.

Direct manipulation in VE demands a real-time constraint perception and satisfaction. Constraints are considered as properties of objects and forced on features. The constraint objects are claimed in the virtual object. One object makes reference to another, and the constraint dependency relationships form a DAG (Directed Acyclic graph). Fast searching and tracing is so important that constraint dependency is explicitly specified as a relationship in the composite object model, i.e., assembly model.

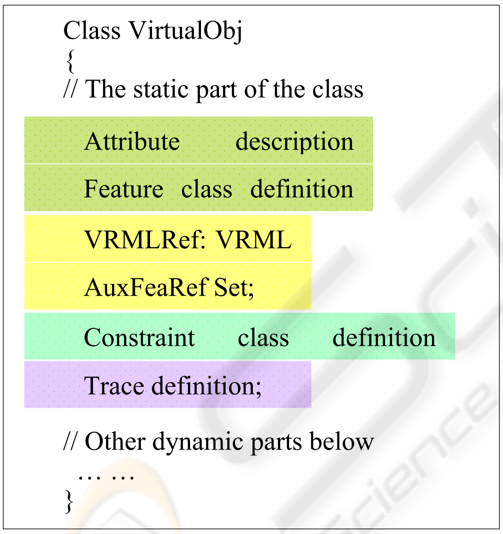


Figure 1: The framework of static description of virtual object class.

As behavior contains much technical information, recording the object's behaviors has a great significance in virtual manufacturing. Behavior is abstracted and formalized as a trace. All the participant's manipulation will finally transited into object's behavior. For the sake of succinctness, the behavior formalization is laid in next section.

Class *VirtualObj* basically gives a template because every application has its specific data requirements. The researchers should give a concrete definition of every attribute in the field based on their specific application and also make efforts to acquire the attributes from the original design.

3 CONSTRUCTIONS OF PERCEPTION AND BEHAVIOUR

The most difficult work in 3D interactive system construction is to design the dynamic characteristics. It mainly depends on designer's experience currently. It is necessary to put forward new perception and behaviour mechanisms for virtual object.

3.1 Mechanism of Perception

The computing based perception mechanism provides to users with the space relationship consciousness to facilitate object manipulating and semantics producing. The strategies of searching all the potential target features are designed. A series of rules of creating some spatial relationships are established. The rules are formed which specify when and how the special semantics are established and the object transit among states. Every kind of domain specific determining algorithm is given. These algorithms are linear and domain dependent. Representations of all kinds of the perceptions are devised. The spatial relationships are built step by step.

Figure 2 shows the example of coincidence constraint perception. This perception works to sense the align constraint that user want to add. After the matched feature pair has been found, shown as the highlighted hole, *Coincidence* perception is working. The '*Major*', '*Minor*' and '*3thAxis*' of the matched feature pair are determined as the potential feature set, shown as the red dashed lines. Spatial relationship rule is like: (*Rule 1*) Parallel relationship is true if one of the '*Major*', '*Minor*' and '*3thAxis*' of the current feature is parallel with one of the '*Major*', '*Minor*' and '*3thAxis*' of the target feature; The domain specific algorithm can be described as: let v_1, v_2 are respectively the vectors of axes of current feature and target feature, the formula of judging the coincidence constraint in the interactive process is as equation 1, and the visual communication is defined as highlighting the coincident auxiliary feature pair.

$$\cos\theta = \frac{|v1 \cdot v2|}{|v1||v2|} \geq \delta \qquad (1)$$

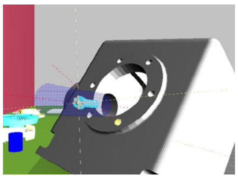


Figure 2: An example of coincidence perception.

3.2 Behavior Construction

Object traces are constructed during the human computer interaction process and interpreted and simulated when the operations are required to repeat.

Definition 1: *(Temporal behavior segment) Temporal behavior segment (tbs) is a basic cell of object behavior, has the form: <t_i ,cn_{start} ,bhv_k, cn_{end}>. Let BHV be set of the primitive object behavior type set，CN be a constraint set, T be the time set. Where t_i□T is the start time of the behavior segment, bhv_k□BHV is a primitive behavior type, cn_{start}□CN is an original constraint, cn_{end}□CN is the terminate constraint.*

Behavior segment above is used to specify the flow transitions. Here the typical behavior types mainly include translation (Trans) and rotation (Rot). The constraint types mainly include '*Point*', '*Line*', '*Plane*', '*Sphere*', '*Circle*', '*Distance*', and etc.

Definition 2： *（Short behavior sequence）Short behavior sequence (sbs) is a sequence made up of three temporal behavior segments which are respectively corresponding to feature-matching perception, coincidence constraint perception and face-mating perception, has the form: $tbs_1;tbs_2;tbs_3$.*

The notation '*;*' expresses the sequential operation between two temporal behavior segments. Short behavior sequence is an abstract of the assembly operation from the domain. In order to simplify it, translation motion and rotation motion are separated apart, that is, when translation is executed, the degree of rotation freedom is frozen and vice versa.

Definition 3: *(Object trace) Object trace (otrace) is a successive short behavior sequences from an initial state to a terminate state, it has the form: $sbs_1;sbs_2;\ldots;sbs_n$.*

Object behaviour is used to formalize the user's operation. In this way, the user's direct manipulation is translated into object's behaviour when an object is manipulated. The expert's knowledge is thus embedded into the object and guides the object to simulate the human operations in a similar circumstance. From the object's assembly trace, we can easily reverse or repeat the assembly process at any time.

4 THE EXPERIMENTS AND ANALYSIS OF 3DIOOM MODEL

The authors conducted experiments to validate the provided 3D interaction oriented object model (*3dIOOM)*. The first aspect is about semantics construction. *IOS* represents the metric of system capability of semantics construction. The second aspect is about perception performance and behavior capability. *IOB* represents the metric of perception and behavior. The last aspect is interaction load. *IOC* represents the metric of total cognition load.

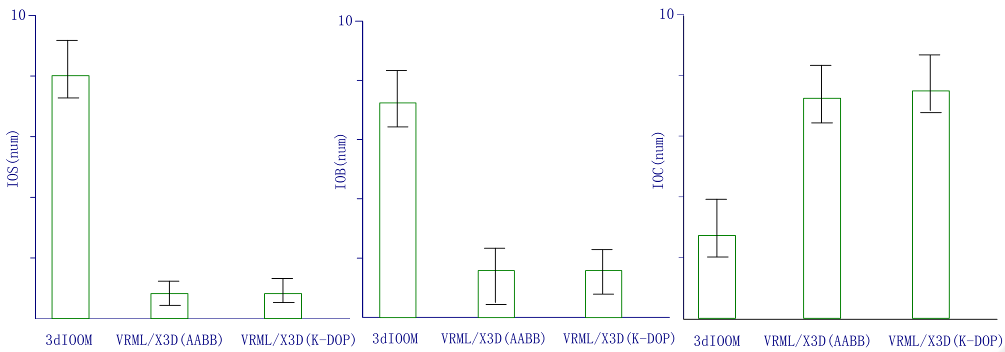
The hardware the authors used is PC machine and Logitech 3D spacemouse. The software platform used is Open Inventor 5.0 and Microsoft Visual C++ 6.0. A gear case which includes 36 parts is used as an example for verifying and validating *3dIOOM*.

Three models: *3dIOOM*, *VRML/X3D* with *AABB* algorithm and *VRML/X3D* with *K-DOP* algorithm are compared because most of the applied models have the same complexity with them. There were 27 participants. All are regular students at Beijing Institute of Technology. The participants were randomly divided into three 9-member groups for the three experiments respectively. The task is to assembly 35 parts on the gear case in the virtual workshop. The metrics are formalized and all values are normalized in 10 scales. The discrete events and the evaluated indexes are counted by the programs. The experiment results are shown by Figure 3.

On the aspect of semantics, *3dIOOM* model can create 8 relationships, i.e., approaching, feature matching, aggregation, constraint dependency, coincidence, face mating, and two traces, while other models only create a collision.

On the aspect of perception and behavior, the perception efficiency and behavior capabilities on representation, adaptation are compared. Perceptions used in *3dIOOM* model are real-time. The behavior mechanism put forward in this paper can make behavior be constructed directly from participant's direct manipulation while others cannot. The behavior in *3dIOOM* is adaptable to the changed environment, while the behaviors in other models can not have this capability.

On the aspect of HCI supporting, the cognitive load in *3dIOOM* comes from the sensing of feature matching searching, coincidence and the face mating. On the contrary, in models of *VRML/X3D* combining with *AABB* or *K-DOP*, the load comes from collision events which occurred a lots of times.



(a) Semantics (b) Perception and Behavior (c) Cognitive load

Figure 3: Experiment results of virtual object model.

5 CONCLUSIONS

The authors put forward an object model *3dIOOM* to promote HCI in VE. It can be clearly depicted by the static and dynamic parts. New perception and behavior mechanisms in VE are established. The experiments showed that *3dIOOM* model can well support 3D direct manipulation in VE. The disadvantages of this model lie in the difficulty of reconstructing the feature based design information. The current situation is a semi-automation process.

ACKNOWLEDGEMENTS

This paper is jointly supported by national science foundation of China (No.60773046), the state key lab foundation of China under grant No. SYSKF0905 and Beijing key discipline program.

REFERENCES

G. Drieux, J-C. Leon, F. Guillaume, N. Chevassus, October 4-6, 2006. Integrating Design with Downstream Applications through Product Shapes Adaptation Processes, *IMACS Multiconference on "Computational Engineering in Systems Applications"(CESA)*, Beijing, China.

Mauro Figueiredo, July 2007. Adjustable approach for collision detection in virtual prototype environments, *Geometric Modelling and Imaging (GMAI'07)*, Zurich, Switzerland, 4-6.

Jinseok Seo and Gerard Jounghyun Kim, Jun, 2004. Explorative Construction of Virtual Worlds: An Interactive Kernel Approach, *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, Singapore.

Gun A. Lee, Gerard Jounghyun Kim, Chan-Mo Park, November 11-13, 2002. Modeling Virtual Object Behavior within Virtual Environment, *VRST'02*, Hong Kong.

H. Reibecca Callison, July 1995. A Time-Sensitive Object Model for Real-Time Systems, *ACM Transactions On Software Engineering and Methodology, Vol .1, No 3, , Pages 287 -317*.

Alexander P. Pons, July 2002. Temporal Abstract Classes and Virtual Temporal Specifications for Real-Time Systems, *ACM Transactions on Software Engineering and Methodology, Vol. 11, No. 3, Pages 291–308*.

Shoyo Hyod, Kouhei Ohnishi, A Method for Motion Abstraction Based on Haptic Information Directionality and An Application to Haptic Motion Display System, *Accepted for publication in a future issue of IEEE Transaction of industrial electronics*.