Systems and Computers in Japan, Vol. 30, No. 11, 1999
Translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol. J81-D-II, No. 5, May 1998, pp. 822–831

A Model for Manipulation of Objects with Virtual Hand in 3D Virtual Space

Kenji Funahashi*

Graduate School of Engineering, Nagoya University, Nagoya, Japan 464-8603

Takami Yasuda and Shigeki Yokoi

School of Informatics and Sciences, Nagoya University, Nagoya, Japan 464-8601

Jun-ichiro Toriwaki

Graduate School of Engineering, Nagoya University, Nagoya, Japan 464-8603

SUMMARY

This paper describes a system for virtual object manipulation with a virtual hand. It is necessary that we be able to grasp, move, and release virtual objects with the hand similarly to real ones in order to have experiences in various virtual worlds. Therefore, we developed a new model for object manipulation with a virtual hand. Basic hand actions such as grasping, moving, and releasing are implemented and gravity and inertia are introduced in a simplified way in the proposed model. The objects are naturally moved with interactive rate in a picture generated by the graphics workstation. © 1999 Scripta Technica, Syst Comp Jpn, 30(11): 22–32, 1999

Key words: Virtual manipulation; virtual hand; real-time interaction; grasping check; kinematics.

*Presently at Educational Center for Information Processing, Nagoya Institute of Technology, Nagoya, Japan 466-8555

1. Introduction

Because of technical progress in recent years, we can see virtual space defined as data on the computer by computer graphics (CG), and experience it like real space with virtual reality (VR) technology. The virtual space is a new space that we can use. Future VR technical development holds increasing promise. We consider that the use of virtual space will spread. We desire to be able to use virtual space like real space. In order to manipulate a virtual object, it is also possible to use a keyboard, a mouse, and the like. However, when we manipulate an object in the real world, we use our hands in most cases. Our aim is to be able to manipulate a virtual object in a way similar to a real object. For example, in operation simulations in medicine, the simulation of an operation does not need to be carried out by the same motion as the operation process for the design of the operation and its possibility evaluation. However, for advance training in an operation, it is desired and needed to imitate the actual operation. It is also desired and needed to

CCC0882-1666/99/110022-11

provide design and training in industrial fields and in education.

In such a situation, research on direct operation of objects by the hand (virtual hand) defined in virtual space which moves corresponding to an operator's hand has been performed [1–8]. This research simulates the feature of operation of an actual hand, but in extremely limited form. For example, it limits judgment to only two fingers for judging interference between the hand and an object, or interference of the thumb with the object in grasping. In Refs. 2 and 3, high-quality images of the process of grasping and the state where the operator is grasping are generated by interference judging which observes details, but the judgment of whether an object is grasped is simplified very much: the operator is recognized as grasping if two or more fingers including the thumb have interfered.

Moreover, research on control of a robot by a computer also deals with the interference problem of a hand and an object [9]. The direction of operation is the reverse, in that the robot hand in the real world is driven by a computer.

Much research on simulation without interaction and on interactive manipulation of virtual objects with the mouse has also been performed [10–14]. However, since these studies manipulate an object in terms of one point, the direct application to "grasping," which is manipulation of the hand by fixing from both sides, is difficult.

In this paper, we describe software for manipulating objects in virtual space in the same way as real space without force feedback. The operator controls his virtual hand by a glove-based input device with sensors. We developed a manipulation model of the virtual objects including finger motion, which can control the motion of a virtual hand more finely, and the software that allows virtual object manipulation in a similar way to real object manipulation [15, 16]. Using physical laws for realization of object manipulation by the virtual hand, it is necessary to solve the appropriate equations and to generate equations which change step by step. Using the finite element method, since it is necessary to calculate the solution of simultaneous equations, the calculation for real-time interaction is time-consuming. Thus, we developed a system of object manipulation by a virtual hand realizable as software by introducing a model which simplified the geometric description of a virtual hand, the interference state of a hand and an object, and objective movement. The objective is a system in which we can manipulate objects freely (grasp, move, and throw) with the virtual hand in a virtual environment where two or more blocks are on the desktop. Therefore, it is the system that considers not only the interaction of a virtual hand and virtual objects, but also the motion of the object after the operation and the collision of the object with the floor.

2. Basic Orientation of Object Manipulation System

In this section, we list the problems which must be solved in order to be able to manipulate objects freely in a desktop environment where two or more blocks are placed, and clarify the basic orientation. First, interactive manipulation must be possible. For good-quality animation, it is generally said that a 24 frames/s refresh rate is required [17]. For realization of interactive manipulation in virtual space, about 12 frames/s is needed [12]. Then, we set the index of real-time processing and targets as 12 frames/s.

Next, in order to manipulate an object, we must define the hand and the object in virtual space. The hand and object interfere at only one or more points; this is described in detail in Section 3.1. Various object manipulations can be classified as follows.

(1) Push at one point.
(2) Grasp at more than one point.
(3) Push at more than one point.

However, we generally move an object by manipulation (1) or (2); manipulation (3) produces a very complicated action by virtue of the positional relations of the contacting points. Thus, we consider chiefly manipulations (1) and (2) and introduce the interaction model.

Since the object currently not manipulated also exists when there are two or more objects for manipulation, we must consider gravity and the interference of objects in virtual space. When an actual situation is considered, interference of objects is an important problem that includes indirect manipulation of other objects by a grasped object and the piling of objects. Since this is a complicated problem, it needs considerable calculation time for processing. In this paper, since we aim at realization of direct manipulation of objects by the hand (after which the operator detaches the hand), we simplified the interaction of objects sharply (Section 4).

3. Object Manipulation by Hand

3.1. Hand and object in virtual space

In order to check the interference between a virtual hand and an object, we set up check points at the tip and joints of each finger, and we check whether interference has occurred at these points (Fig. 1). Furthermore, we detect interference at check lines, which are edges parallel to the generating line of the polygonal cylinder, which is the line decided by two neighboring check points. We can manipulate objects at the "surface" of a finger. Each finger is

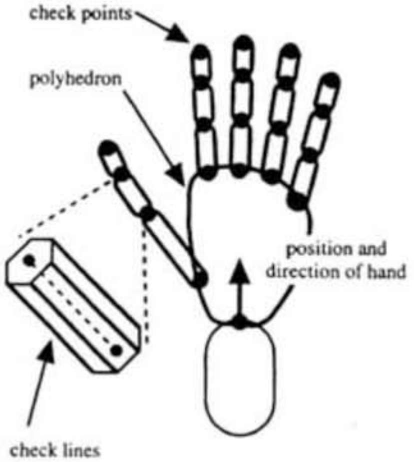


Fig. 1. Model of virtual hand.

displayed by a polyhedron as described above, and the palm is represented by a predefined polyhedron. In actual manipulation, since contact between the object and the palm is limited, we do not consider interference with an object in the palm. However, when required, it is possible to set check points and check lines on the palm. Moreover, below, a standard point is set for the wrist. This is considered as the position of the virtual hand in space. The direction of hand is considered as the direction of the vector at this point. Rotation centered around the vector, and changes of the position and direction of the hand constitute motion of the hand.

When the virtual object is a soft object pushed from both sides, there arises a complicated problem related to modification and multifinger grasping. Our approach is limited to rigid bodies expressed by polyhedra, without movable portions and not fixed. Each surface of an object can interfere with a check point of the virtual hand, and each edge can interfere with a check line.

3.2. Interaction model of hand and object

In this section, we describe the interaction model of one hand and one object. A series of processing outlines for judging the situation of object manipulation is shown in Fig. 2. If interference of a hand and an object is limited to a point, as described in the foregoing section, the situation can be divided into noncontact, contact at one point, and contact at two or more points. Furthermore, if two or more points contact, the situations of grasping and nongrasping can be considered. After boundary checking, two kinds of detection are independently performed: (A) detection of collision points of object surfaces and the check points of the fingers, (B) detection of collision points of edges and check lines.

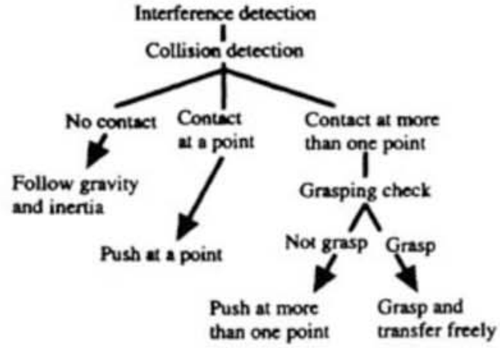


Fig. 2. Interaction model.

Next, the motion of the object is determined according to the number and positions of the collision points detected by (A) and (B). When an object is not touched by the hand, it moves according to gravity and inertia as described in Section 4.1. If it was touched by hand (for example, if it was grasped), this means that the manipulated object has been released.

(1) Contact at one point

When the virtual hand has interfered with the object only at one point, the object is pushed by the hand. As shown in Fig. 3, for example, the check point at the tip of a

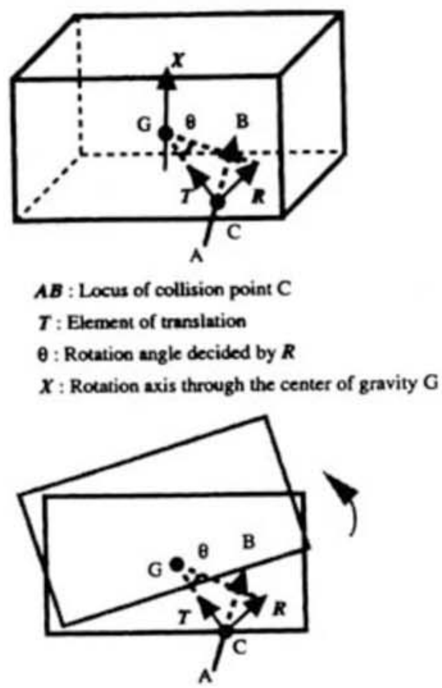


Fig. 3. Pushing an object on a point.

finger has moved to point B from point A. The collision point and the center of gravity of an object are set to point C and point G, respectively. That is, point C on the object surface is moved to point B. Apart from that, generally, movement of a three-dimensional object can be divided into two elements, parallel movement and rotational movement. We consider that the mass of an object centers on the center of gravity, and that parallel movement of the object takes place in the direction of the center of gravity from the interfering point. In rotational movement, without restricting conditions, for example, the object is not fixed, the object rotates around its center of gravity. Then the movement vector $\overrightarrow{CB}$ of point C on the object surface is decomposed into vector $\boldsymbol{T}$ and vector $\boldsymbol{R}$ according to the following formula:

$$\overrightarrow{CB} = \boldsymbol{T} + \boldsymbol{R}, \quad \overrightarrow{CG} \,//\, \boldsymbol{T}, \quad \boldsymbol{T} \perp \boldsymbol{R} \tag{1}$$

Next the object is translated with vector $\boldsymbol{T}$ and is rotated through an angle θ around vector $\boldsymbol{X}$ at the center point G, which is an approximate expression of the movement of point C by vector $\boldsymbol{R}$. Vector $\boldsymbol{X}$ and angle θ are defined by the following formula:

$$\theta = \arctan \frac{\|\boldsymbol{R}\|}{\|\overrightarrow{CG}\|}, \quad \boldsymbol{X} = \boldsymbol{R} \times \boldsymbol{T} \tag{2}$$

At time $\overrightarrow{CB} \,//\, \overrightarrow{CG}$, we set $T = \overrightarrow{CB}$ and $\boldsymbol{R} = \boldsymbol{0}$, and the object is only translated. At time $\overrightarrow{CB} \perp \overrightarrow{CG}$, it is set to $\boldsymbol{T} = \boldsymbol{0}$ and $\boldsymbol{R} = \overrightarrow{CB}$, and the object is only rotated around $\boldsymbol{X} = \boldsymbol{R} \times \overrightarrow{CB}$.

(2) Contact at two or more points, and grasping

If the virtual hand is touching the object at two or more points, we need to check whether the virtual hand is grasping the object or it is pushing without grasping. In this case, the check is not based on a force relation, such as friction, but on the positional relation of an object and a hand. In grasping with point contact, such as the tip of a needle in the real world, the object sometimes slides and rotates as a result of the position of contact, friction, and the direction of the force. Apart from that, when we grasp with our own hand, it is strictly surface contact and there is little variation in the friction coefficient of a hand. Since we may not grasp with contact so that an object is rotated intentionally, the direction of the surface with which the object is in contact governs whether the virtual hand can grasp it.

If the angle between any two normal vectors of the touched surfaces and the edges of an object is more than a predefined threshold value, the object is considered to be grasped by the virtual hand. When all of the angles are less than the threshold value, the object is considered to be pushed by two or more points of the virtual hand. However, we assume the perpendicular to the edge at the center of gravity as the pseudo-normal vector, and we refer to the normal vector of a surface and the pseudo-normal vector of an edge (Fig. 4). For example, when an object is interfered with at three points (Fig. 4), if at least one of the angles θ_{12}, θ_{23}, and θ_{13} decided by the three normal vectors N_1, N_2, and N_3 is more than the threshold value, the object is considered to be grasped. Furthermore, when a finger has penetrated into the object, the joint angles of the finger are rectified so that the contacting point is moved to the object surface [16]. Moreover, successive instances in which grasping is recognized constitute continuation of grasp.

We set the threshold value of the grasping check as 90°. This is also based on the surface state of the object, because we can grasp an object where two surfaces (e.g., of a cube) adjoin each other, and a small object may actually be grasped in this way. Moreover, ease of grasping, which is a function of the mass of an object and the difference in friction between fingers, can be altered by changing the threshold value.

When the object is grasped by the virtual hand, it is made to follow the motion of the hand by considering that its position relative to the hand does not change.

(3) Contact at two or more points without grasping

An object is pushed by two or more points of a virtual hand if it is not recognized as being grasped in the above-mentioned check. The motion of the object is very complicated, depending on the relative positions of the contacting points (each finger moves independently) in the real world. It is not realistic to solve the interference problem dynamically, and it is equivalent to the interference problem of objects with only point contact (see Section 4.3). A virtual object pushed at two or more points only translates, follow-

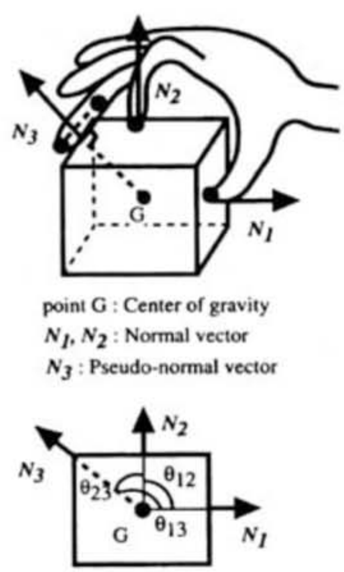


Fig. 4. Pseudo-normal vector and grasping check.

ing the motion of the hand. This is a realistic simplification, in order to allow mainly grasping of an object by fine manipulation of the finger points when moving the object on a desk.

4. Virtual Environment

4.1. Gravity, inertia, and friction

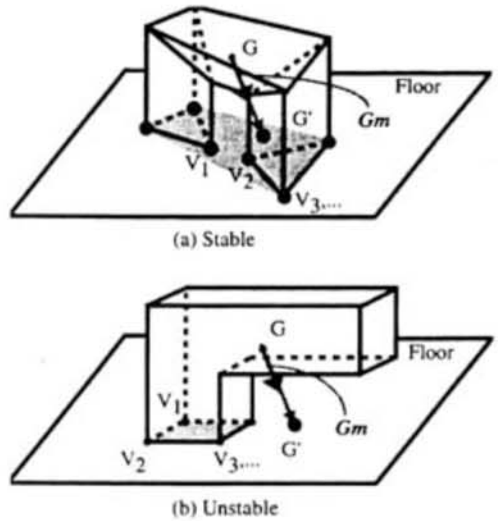
When virtual object manipulation is realized, consideration must be given to the object currently not manipulated, that is, which is not currently touched by the virtual hand. Furthermore, the model must provide a calculation method such that these objects always are able to be manipulated. This model includes simplified gravity and inertia. When an object is not interfered with by a hand or other objects, the position of its center of gravity $\boldsymbol{P}(t)$ is obtained from the position at time $t - \Delta t$, namely, $P(t - \Delta t)$, the change of inertia, $\boldsymbol{P}(t - \Delta t) - \boldsymbol{P}(t - 2\,\Delta t)$, and the acceleration of gravity, $\boldsymbol{g}\,\Delta t^2$. $\boldsymbol{P}(t)$ is defined by the following formula:

$$\boldsymbol{P}(t) = 2\boldsymbol{P}(t-\Delta t) - \boldsymbol{P}(t-2\Delta t) + \boldsymbol{g}\Delta t^2 \qquad (3)$$

where $\boldsymbol{g}$ is the downward gravitational acceleration vector, and the refresh time Δt is the inverse of the refresh rate $\boldsymbol{F}$ (frames/s). When an object is thrown, its initial velocity is $[\boldsymbol{P}(t - \Delta t) - \boldsymbol{P}(t - 2\,\Delta t)]/\Delta t$. In rotation, a released object continues in relative rotation around the center of gravity. Moreover, inertia is not conserved and an object does not rebound after interfering with other objects, such as a floor. As regards friction, without dynamic calculation an object does not slide. By having such an assumption, we can manipulate an object which moves and is not currently being manipulated, and can grasp and throw an object.

4.2. Interference with a falling object and the floor

In this section, we describe the motion of a falling object and its behavior after collision with the floor. An object is moved according to gravity and inertia as mentioned above until interfering with a floor. When a falling object penetrates into the floor at time Δt, the situation is rectified by translating the object so that its vertex which has penetrated the floor most deeply is at the level of the floor. The concrete procedure is as follows: the object's vertices (contact points) which are on the floor are set to V_i ($i = 1, \ldots,$ the number of points of contact), with contact points V_i making a convex hull (Fig. 5). Point G′ is the projection of the center of gravity G in the direction of movement vector $\boldsymbol{Gm}$:



$V_1, V_2, \ldots, V_i$: Vertices of an object on the floor
point G′ : Projected point of the center of gravity G in the direction of moving vector $\boldsymbol{Gm}$

: Convex-hull area decided by $V_1, V_2, \ldots, V_i$

Fig. 5. Stability check of an object on the floor.

$$\boldsymbol{Gm} = \boldsymbol{P}(t-\Delta t) - \boldsymbol{P}(t-2\Delta t), \qquad (4)$$

$$\boldsymbol{Gm} \parallel \overrightarrow{GG'} \qquad (5)$$

When the point G is projected into the convex hull made by the contact points V_i (G′ is in this convex hull), the object is stable on the floor and does not move on it [Fig. 5(a)]. When the point G′ is located outside of the convex hull, the object is unstable [Fig. 5(b)]. In this case, the object moves until judged to be stable according to the relation with the floor.

We consider the three states as regards the number of contact points with the floor, as shown below (Fig. 6) [15].

(1) Contact at a vertex V_1 [Fig. 6(1)]

Point V_1 is set to point C [Fig. 6(1)]. The vector $\boldsymbol{Gm}$ is decomposed into vector T which is an element of slide and rebound, and vector $\boldsymbol{R}$ which is an element of rotation, according to the following formula, as in pushing manipulation by one finger:

$$\boldsymbol{Gm} = \boldsymbol{T} + \boldsymbol{R}, \quad \overrightarrow{CG} \parallel \boldsymbol{T}, \quad \boldsymbol{T} \perp \boldsymbol{R} \qquad (6)$$

However, the object is not translated by vector $\boldsymbol{T}$ and the contacting point does not move. The object is acted on by gravity only at the center of gravity. The object rotates through an angle θ around vector $\boldsymbol{X}$ at the center point G. At time $Gm \parallel \overrightarrow{CG}$, it is in the well-balanced state, and does

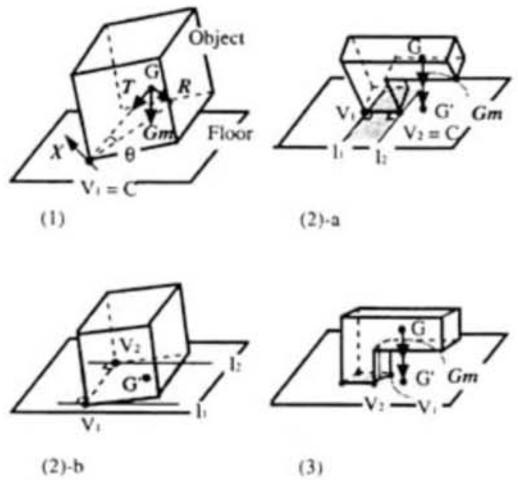


Fig. 6. Motion of an object on the floor.

not move. Vector X and angle θ are defined by the following formula:

$$\theta = arctan \frac{\|R\|}{\|\overrightarrow{CG}\|}, \quad X = R \times T \tag{7}$$

(2) Contact at two vertices V_1 and V_2 [Figs. 6(2)-a and 6(2)-b]

Two lines are drawn on the floor passing through V_1 or V_2 and perpendicular to the line $\overline{V_1V_2}$. If point G′ is not located between them [Fig. 6(2)-a], let one of the two vertices closer to G′ be point C. The motion of this object is decided by the rule described in (1). Otherwise [Fig. 6(2)-b], the two points in contact with the floor do not move, namely, the object rotates around the line $\overline{V_1V_2}$. In this case, the object also rotates similarly to rule (1).

(3) Contact at more than two vertices V_i [Fig. 6(3)]

When G′ is no longer inside the convex hull, in other words, when the object is unstable, it is considered that no contact points move but that the others move in the direction away from the floor with the exception of two points or one point. Renaming the two vertices closest to G′ on the convex hull as V_1 and V_2, the motion of the object follows case (2) above.

4.3. Interference among objects

Interference of objects is a problem which must be considered, in order to realize indirect manipulation of other objects by an object which is grasped, and also the arranging of objects. However, the dynamic solution for rigid bodies interfering with each other may become NP-complete, as described in Ref. 12. Thus, when several objects collide with each other, we could solve the problem in the same way as the case of contact between an object and a hand or the floor. However, when an object which was not being manipulated directly interferes with a hand as a result, it is not easy to remove the inconsistency in virtual space without restraining the operator's hand.

In this research, we aim at realizing object manipulation by the virtual hand similarly to the situation in real space. Thus, at present, we have simplified this situation in the following way. If one object is touched by a hand and moved, all objects interfering with it are also translated along with it. Calculation of the stability of intricately stacked objects is difficult at an interactive rate, and it is one of the elements which reduce the sense of grasping manipulation. If objects interfering with each other are not touched by a hand, they are not affected by gravity and inertia.

5. Example of Experiment

5.1. Composition of experiment system

Using the model mentioned above, we realized a system of virtual object manipulation by a virtual hand in C language on a graphics workstation (SGI Onyx RE2, R8000). This system is constructed as shown in Fig. 7(a). It obtains the joint angle of each finger via a glove-type input device (at center of the figure, equipped on the hand). It has 18 sensors, each of which can obtain a value of 8 bits (0 to 255). It is set to 0.5°/value. The system gets the position and the direction of the hand via the motion sensor, using a low-frequency magnetic field (at center of the figure, mounted on the wrist). It can get the position and angle data with a static accuracy of 0.03 inch and 0.15°, and a resolution of 0.0002 inch/inch and 0.025°, respectively. Stereo viewing with stereo glasses using liquid-crystal shutters (at right of the figure, on the head) and HMD is also possible.

5.2. Experimental results

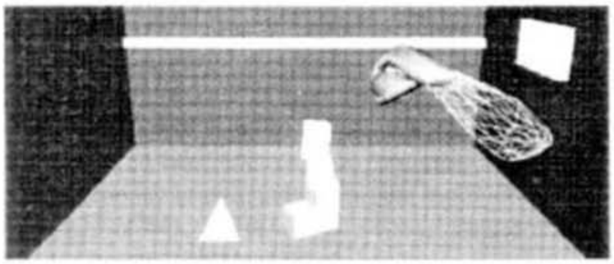
[Experiment 1]

In an experiment with this system, three subjects (a, b, c) perform the following virtual object manipulation after practicing for about 5 minutes.

(1) Free virtual object manipulation
(2) Grasping without using the thumb
(3) Flicking by the tip of finger
(4) Throwing an object which is grasped
(5) Putting an object on another object

An example operation is shown in Fig. 7. Figure 7(a) shows object manipulation in virtual space, and Fig. 7(b) shows the CG image of the virtual space defined in the computer. As shown in Fig. 8, the subjects can manipulate a virtual

(a) System and example of manipulation



(b) CG image of virtual space

Fig. 7. System configuration and examples of manipulating objects with a virtual hand.

object of arbitrary polyhedral shape with arbitrary parts of the virtual hand. This shows that the operator can manipulate the virtual object similarly to a real one, not based on gesture, but taking the interaction of the five fingers and the object into consideration. Figure 9 shows a pyramid flicked by the index finger. Figure 10 shows a thrown object stationary on the floor after parabolic movement. Figure 11

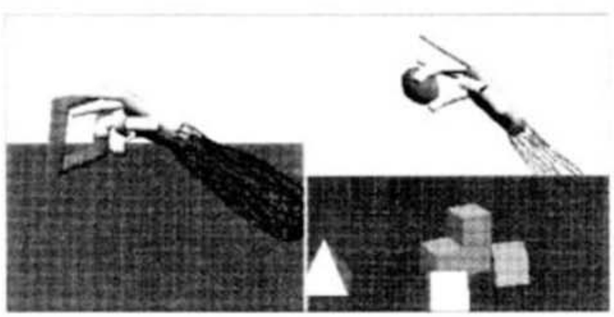


Fig. 8. Manipulation of objects (arbitrary shape of polyhedrons) with virtual hand.

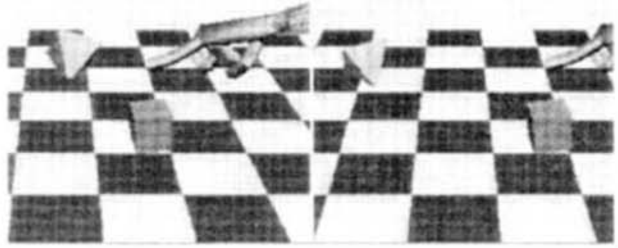


Fig. 9. Flicking an object with a finger of virtual hand.

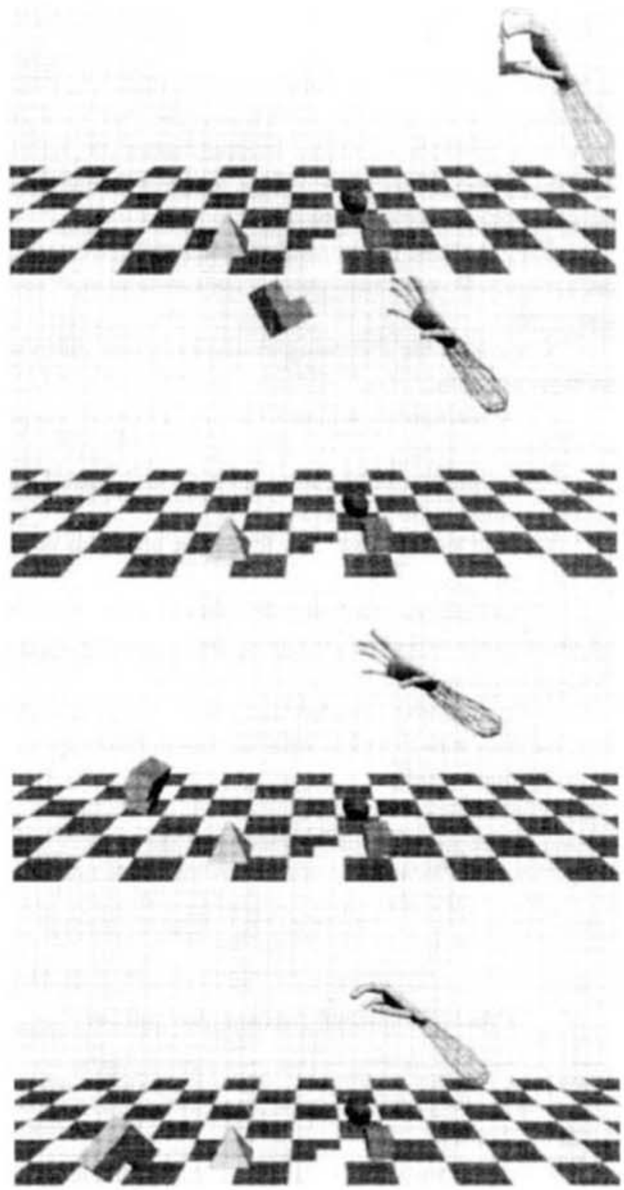


Fig. 10. Throwing an object with a virtual hand.

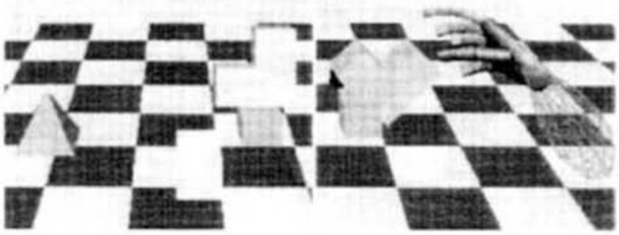


Fig. 11. Putting an object on another object with a virtual hand.

shows an object piled on another, when the operator pushes an L-shaped block along the floor into a cube (Section 4.3); there is a problem of object interpenetration (at right), handled by the rectification process for the penetration of the cube into a floor (Section 4.2).

Table 1 shows the evaluation of each manipulation by each subject; ◎: sufficient mobility and naturalness, ○: fair, and △: poor.

Generally, the following evaluations were received from the subjects.

- Can manipulate a virtual object in a similar way to a real one.

- The motion of the thrown object and that of the object rolling on the floor are natural.

On the other hand, the following problems were pointed out.

- Since he can determine the contact of the hand and an object only from the motion of the object, he cannot manipulate quickly.

- The motion of the objects interfering with each other is unnatural, manipulation with the glove differs greatly from the real world.

The evaluation △ of manipulation (5) in Table 1 is related to the problem of interaction of the objects mentioned above. The evaluation of operation (3) is related to

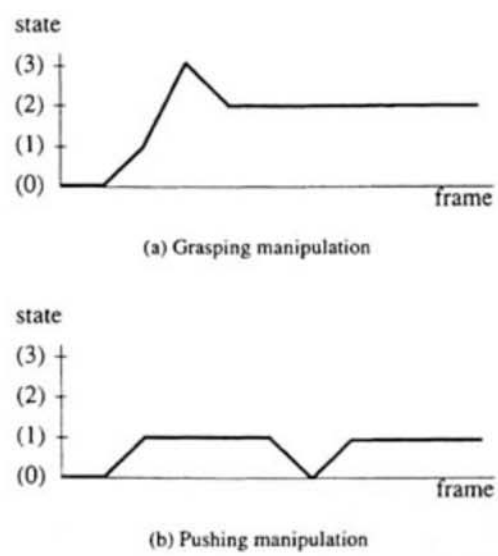
Table 1. Evaluation for each manipulation

Manipulation		(1)	(2)	(3)	(4)	(5)
subject	a	◎	◎	○	◎	○
	b	○	◎	○	◎	△
	c	◎	△	◎	○	○

the problem of 3D position recognition. The operator can bring his finger close to an object slowly for grasping. But he must move his finger quickly for flicking, and the moment of contact is unclear. As regards manipulation (2), subject c commented that "I am not dexterous with the tip of my finger."

The subjects were mainly grasping the object in operation (1). One or two frames before grasping, the subjects sometimes pushed the object at one or more points. When the subjects pushed an object intentionally, they were mainly pushing with the tip of the index finger. The change of contact state in one manipulation is shown in Fig.12. The same result was also obtained in other manipulations. It can be judged that the importance of pushing an object at two or more points as described in Section 3.2 (3) is low.

In the object manipulation system proposed in this paper, we can manipulate blocks set on the desktop in virtual space freely, with moving, stacking, and throwing by a virtual hand. The motion of the thrown object (after it has left the hand) can also be expressed by taking into consideration not only the interaction of the virtual hand and the virtual object but also the motion after manipulation of the virtual object and the collision of an object with the floor. However, when objects interfere, the problems of interpenetration mutually and of unnatural motionlessness remain.



state: (0): no contact
(1): push at one point
(2): grasp (with more than one point)
(3): push at more than one point

Fig. 12. The change of contact state.

[Experiment 2]

The experiment needed evaluation not only in a laboratory as mentioned above but also by a wider range of subjects. We obtained data at the demonstration of our system at the New World Expo (1997, Nagoya, Japan), based on our observations and on conversations with testers. The demonstration, run on an SGI Indigo2 HighIMPACT, R10000, was a game in which we throw a block at a target whose radius is 15 cm and which is 2 m away in virtual space. The testers were not only researchers but also many lay visitors of various nationality, sex, occupation, and age (including schoolchildren).

About 100 persons tested the system for several minutes after explanation, and about 90% of them could throw a block to a target. Many of their evaluations of the object manipulation system were positive, such as "it is interesting that I can touch a virtual object with my own hand." On the other hand, there were problems, such as the lack of a sense of touch (there is no force feedback), and unnatural stacking (simplification of interference of objects).

[Experiment 3]

We evaluated the computation time of interference between cubes defined by 6 surfaces and 12 edges and a virtual hand defined by 20 check points and 60 check lines without fixing the refresh interval Δt. The change in the amount of computation for all objects and grasped objects is shown versus the number of frames which could be drawn in a second in Fig. 13. We also evaluate the calculation time for interference with an ellipsoid approximated by 32 surfaces and 56 edges [Fig. 7(b)].

In this experiment, the system can refresh its screen at 17 frames/s when there are 10 objects and the operator grasps 2 objects (cubes). Therefore, interactive manipulation is realizable at 12 frames/s which is the needed refresh rate. As regards the increase in the computation time accompanying an increase in the number of objects interfering with the hand, interactive manipulation with a hand is realizable at the above refresh rate, because the operator usually touches few objects simultaneously. As regards the complexity of object form, although the system cannot now achieve manipulation of objects defined by many surfaces and many edges, manipulation of a sphere approximated by a polyhedron is realizable at 17 frames/s. Since the change of computation time in response to a change in the number of all objects (horizontal axis of Fig. 13) is small when the number of objects interfering with the hand is fixed, the system can refresh its screen at almost the same rate even when there are several dozen objects. Operators can manipulate objects in a situation with many objects if they touch few objects simultaneously. However, this is because we considered operational feelings to be more important than interference between objects, which is simplified in this system. We should improve this aspect in order to realize more exact object interference.

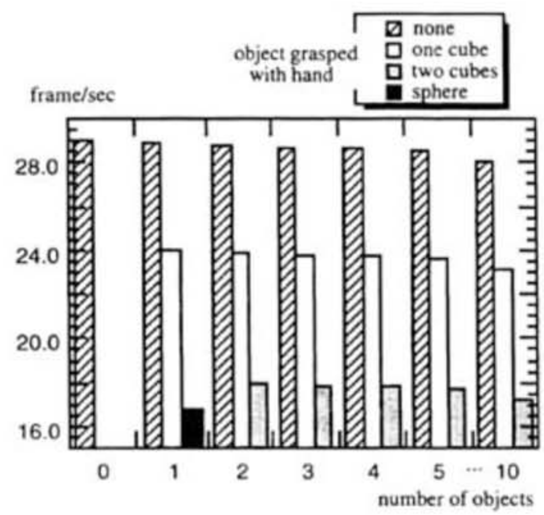


Fig. 13. Number of frames drawn in the unit time.

6. Conclusions

In this paper, we have proposed a model for object manipulation by a virtual hand and have presented an example of an experiment in a virtual-reality system. We can grasp, move, and throw objects in this system at an interactive refresh rate. This model does not have restrictions on manipulation but uses major simplifications already used conventionally. An operator can manipulate objects with every portion of each finger. For the motion of an object which is not manipulated by a hand, the model introduces simplified gravity. The operator receives an impression close to that of the real world.

This system is in the stage of realization of the basic functions, and the following subjects are left for future realization of interactive manipulation of 3D objects by the virtual hand in a more natural way.

[Interaction of hand and object]

- Realization of natural action when pushing with two or more fingers

- Correspondence to object manipulation by the cooperative work of both hands or of the hand of many operators

[Action of the object not being manipulated by a hand]

- Natural action of an object which is interfering with other objects

- Motion of an object which rebounds or slides on the floor and contacts other objects

The addition of these functions, realization of operations on soft objects, introduction of force feedback, verification of ease of realization, and addressing of problems of object manipulation by this model will require experiments on virtual manipulation in various situations representing concrete problems. It will also be important to compare and examine the amount of calculation and the motion of objects in interactive physical simulations.

Acknowledgment. We thank our colleagues in the Toriwaki laboratory of Nagoya University for useful discussions.

REFERENCES

1. Bergamasco M, Degl'Innocenti P, Bucciarelli D, Rigucci G. Grasping and moving objects in virtual environments: A preliminary approach towards a realistic behavior. Proc IEEE Int Workshop on Robot and Human Communication, 1994:44–49.
2. Huang Z, Boulic R, Thalmann NM, Thalmann D. A multi-sensor approach for grasping and 3D interaction computer graphics: Developments in virtual environments. Proc Computer Graphics International'95, Academic Press, p.235–253.
3. Boulic R, Rezzonico S, Thalmann D. Multi-finger manipulation of virtual object. Proc ACM VRST, ACM, 1996:67–74.
4. Kunii Y, Hashimoto H. Object grasping in virtual environment using dynamic force simulator. Proc IEEE Int Workshop on Robot and Human Communication, 1994:261–264.
5. Kawakatsu et al. Object selection method for virtual reality. Inf Proc Soc J Graphics and CAD 1993;66-1:25–32.
6. Iwata H. Artificial reality with force-feedback: Development of desktop virtual space with compact master manipulator. Computer Graphics 1990;24:165–170.
7. Hirata et al. Virtual workspace for groupwork. Trans IEICE 1993;J76-D-II:1788–1795.
8. Kishima et al. Virtual physics: object manipulation in virtual space. Tech Rep IEICE 1995;MVE95-24.
9. Japan Robotics Society. Robotics engineering handbook. Vol III. Control techniques. 3. Hand movement and control. Corona Press; 1990. p 297–320.
10. Moore M, Wilhems J. Collision detection and response for computer animation. Computer Graphics 1988;22:289–298.
11. Kitamura Y, Takemura H, Ahuja N, Kishino F. Interference detection among objects for operator assistance in virtual cooperative workspace. Proc IEEE Int Workshop on Robot and Human Communication, 1993:442–447.
12. Shinya M, Forgue M-C. Laying out objects with geometric and physical constraints. The Visual Computer 1995;11:188–201.
13. Braff D. Interactive simulation of solid rigid bodies. IEEE Computer Graphics and Applications 1995;15:63–75.
14. Takeda et al. Simulation of rigid body motion based on the laws of physics. Trans IEICE 1996;J79-D-II:860–869.
15. Funahashi K, Yasuda T, Yokoi S, Toriwaki J. Block manipulation with hand in virtual space. Proc Virtual System and Multimedia'95, p 189–194.
16. Funabashi et al. Object manipulation in virtual space and hand-object interaction models. Papers of First Computer Engineering Society Meeting, 1996;1:15–18.
17. Television Society of Japan. Television and image engineering handbook. 1. Light, color, impression, image. Ohm Press; 1990. p 57–75.

AUTHORS (from left to right)

Kenji Funahashi is a research associate in the Educational Center for Information Processing at Nagoya Institute of Technology. His research interests are computer graphics and virtual reality. He received his B.E. degree in electronics and information engineering from Gifu University in 1993. He received his M.E. and Ph.D. degrees in information engineering from Nagoya University in 1995 and 1998.

Takami Yasuda is an associate professor in the School of Informatics and Sciences at Nagoya University. His research interests include computer graphics and virtual reality on the Net. He received his B.E. and M.E. degrees in electronic engineering from Mie University in 1982 and 1984. He received his Ph.D. degree in information engineering from Nagoya University in 1989.

Shigeki Yokoi is a professor in the Graduate School of Human Informatics at Nagoya University. He received his doctoral degree in electrical engineering from Nagoya University. He has been an associate professor at Mie University and an associate professor in the School of Engineering at Nagoya University. His research interests include computer graphics, virtual reality, interactive media, and social affection of digital technologies.

Jun-ichiro Toriwaki has been a professor in the Department of Information Engineering at Nagoya University. He specializes in the areas of pictorial pattern recognition, biomedical image processing, and computer graphics with applications. He received his B.E. and M.E. degrees in electronics engineering and his Ph.D. degree in electrical engineering from Nagoya University in 1962, 1964, and 1969.