Virtual 3D Manipulation Using Cutting Plane Lines

Hidetoshi Nonaka
Graduate School of Information Science and Technology, Hokkaido University
Sapporo, 060 0814, Japan

ABSTRACT
Various techniques have been proposed for virtual 3D user interfaces. They are roughly classified into two categories, virtual 3D manipulation, and virtual 3D navigation. In this paper, we present a virtual 3D/6DOF manipulation system using conventional 2D pointing device, such as a mouse with single button.

Keywords: 3D user interface, Pointing device, 6DOF operation, Cutting plane line

1. INTRODUCTION
There have been various methods to realize virtual 3D manipulation using 2D pointing devices such as mouse.

In order to achieve the rotation of 3DOF, Hanson [1] proposed a method by assigning the movement of x-axis and y-axis to the positive (counter-clockwise) rotation around y-axis and the negative (clockwise) rotation around x-axis (Rolling ball). In the result, the positive rotation of a virtual 3D object around the z-axis is achieved by a negative rotation of the mouse. Chen et al. [2] proposed a 3DOF rotation method using a virtual track ball in addition to 3DOF rotation using the slider, the menu, and the button, and evaluate the operational performance. Shoemake [3] formulated the correspondence of the turn of the arc on a virtual sphere and virtual 3D object in geometrical consideration, and has achieved the mouse operation not to accompany hysteresis (ARCBALL). Henriksen et al. [4] reformulated the rotation operation method with the virtual track ball, and improve the error margin. Khan et al. [5] proposed a 3D widget as an orientation controller (View Cube).

In order to achieve the movement of 3DOF, many systems adopted “ray casting” method or “camera control” method, for example, Balakrishnan et al. [6] proposed a method using two-handed operation. Stuerzlinger et al. [7-8] introduced a translation method by restricting the movement to sliding on a surface for the mouse cursor (SESAME), and it was applied to a virtual 3D puzzle system [9]. Moreover, a lot of input devices specialized to 3D/6DOF manipulations have been also proposed [10-11].

We have presented a demonstration of a 3D/6DOF direct manipulation system, using mouse with single button [12]. It means 3D rotation and 3D movement of virtual objects are achieved with only left button and none of right button, mouse wheel, or other devices are required. It also means this method is applicable to most 2D pointing devices such as pen-tablet, pointing-stick, trackball, touchpad, and so on.

Unlike the past techniques such as virtual trackball, virtual sphere, or 3D widgets, the user can rotate a virtual 3D object with the feeling like gripping it directly. By using cutting plane line, the user can rotate a virtual 3D object along a trajectory on a plane, which can be adjusted within a 3D space. The cutting plane line is also helpful for the perception of depth and positional relationship of the object and other objects.

In this paper, we present a virtual 3D/6DOF manipulation system using revised method of cutting plane line.

2. ROTATION IN 3DOF
In this section, we summarize our method described in [12]. By assuming that a virtual 3D object is adequately modeled and displayed to the user, and the user perceives it properly with the pictorial cue, user’s pointing on a perspective image of the object can be regarded as the pointing on the object in a virtual 3D space.

In the Figure 1, a virtual icosahedron is shown. The origin of the screen, the origin of the virtual 3D space, and the center of rotation are assumed to be identical for only the simplicity. The coordinate system of the virtual 3D space is right-handed. Suppose the user presses the left button when the mouse cursor is on the point P, the corresponding 3D position vector \( \mathbf{p} \) is given by (1).

\[
\mathbf{p} = \mathbf{OP} = (p_x, p_y, 0)^T, \quad \mathbf{p} = \mathbf{OP} = (\tilde{p}_x, \tilde{p}_y, \tilde{p}_z)^T
\]

\[
\tilde{p} = \frac{n'(c-h)p + n'(c-p)h}{n'(p-h)}, \quad (1)
\]

where \( n, c, \) and \( h \) are the three-dimensional coordinate of the normal vector of triangle ABC, the coordinate of arbitrary point on the same triangle, and the coordinate of the viewpoint of the perspective transformation. The distance between the center of rotation O and the point P is given by (2).

\[
\rho = \frac{1}{n'(p-h)} \sqrt{(n'(c-h))^2 p'p + (n'(c-p))^2 h'h} \quad (2)
\]
Figure 1. Pointing on the surface of a virtual 3D object.

Figure 2. Dragging on the surface of a virtual 3D object.

In the Figure 2, suppose the user performs a dragging from P to Q. The axis of rotation \( r \) and the angle or rotation \( \theta \) are calculated by (3) and (4).

\[
q = (q_x, q_y, 0)^t, \quad u = \overrightarrow{PQ} = (u_x, u_y, 0)^t
\]

\[
r = (r_x, r_y, r_z)^t, \quad v = p^t pu / p^t p, \quad w = u - v,
\]

If \( 0 \leq p^t p < \rho^2 \)

\[
r_x = -v_y - \sqrt{1 - p^t p / \rho} w_y,
\]

\[
r_y = v_x + \sqrt{1 - p^t p / \rho} w_x, \quad r_z = \frac{1}{\rho} p \times w,
\]

If \( \rho^2 \leq p^t p \)

\[
r_x = -v_y, \quad r_y = v_x, \quad r_z = \frac{1}{\sqrt{p^t p}} p \times w,
\]

\[
\theta = \arctan(\sqrt{u^t u / \rho}).
\]

Finally, the matrix of rotation is given by (5).

\[
R = \frac{1}{\sqrt{u^t u + \rho^2}} \left( \frac{u^t u \overrightarrow{\Phi}}{\sqrt{u^t u + \rho^2} + \sqrt{u^t u [\overrightarrow{\Phi}] + \rho I}} \right)
\]

In the case that the center of rotation and the origin are different point, only the parallel translations of coordinate axes before and after the rotation are required. Consequently, the rotation of 3D object in 3DOF can be achieved by dragging using conventional 2D pointing device. In the rotation by (5), there are following characters:

1. During dragging, if the mouse cursor is on the object, the positional relation between the cursor and the surface of the object is roughly constant.
2. During dragging, even if the mouse cursor is off the object, the speed of rotation is roughly proportional to the speed of mouse.
3. During dragging, if the distance between mouse cursor and the origin is longer than \( \rho \), \( w \)-component of the vector \( u \) corresponds to the rotation around the \( z \)-axis.
4. The speed and the orientation of rotation depend on where on the surface of the object is dragged. Figure 3 and Figure 4 show the examples of the relation between dragging point and rotation.

Figure 3. An example of the relation between dragging point and rotation. The speed of rotation depends on the distance in the 3D space between the dragging point and the center of rotation.

Figure 4. An example of the relation between dragging point and rotation. The orientations of rotations are opposite between lateral side and medial side of polyhedral faces.
3. MOVEMENT IN 3DOF USING CUTTING PLANE LINE

The mouse is one of indirect-control pointing devices. Cognitive processing for hand/eye coordination and sensori-motor operation are required to the mouse user. Especially, appropriate mapping between the surface under the mouse and the plane of the display are necessary. However, once the mapping is established, the user can operate the mouse even in the cases that the display is horizontal, the display is upright, the projector-screen is obliquely backward, and so on. It suggests that the user can operate the mouse if he/she can perceive a plane in a virtual 3D space.

We assume to approximate a curve or trajectory in three-dimensional space by the concatenations of curves or trajectories within two-dimensional subspaces (Figure 5). It can be considered to be the approximation of 3D curve by a continuous sequence of piecewise torsion-free curves, and the natural dimensional extension of piecewise linear approximation of a curve.

![Figure 5. Approximation of 3D curve by 2D curves.](image)

In our method, the operation of object manipulation is divided into three phases: the phase to rotate (Rotating), the phase to adjust the plane (Tilting), and the phase to move along the plane (Translating). The movement along arbitrary trajectory can be achieved by alternately executing the last two phases. The phase switches by releasing and pressing the left button within about 200 msec. Figure 6 shows simplified state transitions among three phases and the normal phase (Neutral). In the figure, “Press” stands for the pressing mouse left button, and “Release” stands for the releasing mouse left button.

In the second phase, the axes are tilted in 2DOF (roll and pitch) according to the mouse movement. The center of rotation is the virtual 3D position that was pointed by mouse at the moment of state transition. During the second phase, the plane is indicated by cutting-plane lines on all of the 3D objects in the screen. The three axes of the plane are calculated by (6). Again, \( u \) denotes the two-dimensional vector of dragging.

\[
\begin{align*}
\theta &= \frac{\mu}{2} \sqrt{u_x^2 + u_y^2} , \quad c = \cos \theta , \quad s = \sin \theta , \\
e_x &= \begin{pmatrix} cu_x + u_y^2 \cr u_x^2 + u_y^2 \cr 0 \end{pmatrix} , \\
e_y &= \begin{pmatrix} c u_x + u_y^2 \cr u_x^2 + u_y^2 \cr 0 \end{pmatrix} , \\
e_z &= \begin{pmatrix} s u_x \cr s u_y \cr \sqrt{u_x^2 + u_y^2} \end{pmatrix} , \\
\end{align*}
\]

where \( \mu \) is a constant depending the cursor/device ratio, individual difference, and so on.

In the third phase, x-y movement of mouse corresponds to the cursor movement along the plane aligned in the second phase. The cutting-plane lines are hidden during the third phase.

Figure 7 shows an example of 3D/6DOF applications. Recently, 3D puzzle of this type was adopted as the problem of a UI contest [13]. In this example of puzzle, last two pieces remain uncompleted (1). By rotating the rightmost piece, it becomes parallel to the position of the target (2). By adjusting the tilt of the plane with the cutting plane line, the piece and the target are located on the same plane (3). The rightmost piece is translated to the target along the plane (4). In the same way, the last piece is also rotated and moved to the position of the target (5-7). Time required to complete such a virtual puzzle was less than 1.5 times of that to assemble the same real puzzle by hand, as the result of preliminary experiments for evaluation.

4. CONCLUSION

We proposed a method for virtual 3D/6DOF manipulation. The operation of object manipulation was divided into three phases: rotating, tilting, and translating. Especially, the translating of 3D object was realized using cutting plane line. Experimental evaluation is near future work.
Figure 7. An example of applications. A 3D puzzle can be operated with 3 phases: rotating, tilting plane, and translating.

REFERENCES