Path Drawing for 3D Walkthrough

Takeo Igarashi[1], Rieko Kadobayashi[2], Kenji Mase[2], Hidehiko Tanaka[1]

[1]Dept. of Info. Engineering, Univ. of Tokyo
7-3-1 Hongo Bunkyo-ku
Tokyo, 113-8656, JAPAN
Tel: +81 3 3812 2111 ext. 7413
E-mail: {takeo, tanaka}@mtl.t.u-tokyo.ac.jp

[2]ATR Media Integration & Communications
Research Laboratories
Seika-cho, Soraku-gun, Kyoto, 619-0288, JAPAN
Tel: +81 774 95 1443
E-mail: {rieko, mase}@mic.atr.co.jp

ABSTRACT
This paper presents an interaction technique for walkthrough in virtual 3D spaces, where the user draws the intended path directly on the scene, and the avatar automatically moves along the path. The system calculates the path by projecting the stroke drawn on the screen to the walking surface in the 3D world. Using this technique, the user can specify not only the goal position, but also the route to take and the camera direction at the goal with a single stroke. A prototype system is tested using a display-integrated tablet, and experimental results suggest that the technique can enhance existing walkthrough techniques.

KEYWORDS: interaction techniques, virtual spaces, 3D walkthrough, pen computing, user study.

INTRODUCTION
Efficient 3D navigation techniques are required to meet the increasing popularity of virtual spaces. Existing walkthrough techniques can be divided into roughly two categories. One is *driving*, where the user continuously changes the camera position using advancing and turning buttons (arrow keys, joysticks, or button widgets on the screen). The other is *flying*, where the camera automatically jumps to the goal position that the user had specified using a pointing device [1]. Driving is commonly used for computer games, but can cause unwanted overhead when the walking is not the primary purpose of the interaction, because the user has to continuously press buttons during the movement. This problem gets serious especially when the rendering speed is slow, which is often the case with current desktop VR on PCs. Flying provides a solution to the problem, freeing the user from continuous control. All the user has to do is to click the target, then he can arrive at the target position instantly. However, flying suffers from its limited expressive power. The user cannot specify which route to take during the movement, nor can he control the orientation of the camera directly.

PATH DRAWING FOR 3D WALKTHROUGH
We propose a *path drawing* technique for 3D space navigation, which is an extension of the flying technique. It allows the user to draw the desired walkthrough path directly on the screen using a free stroke. Then, the system automatically calculates the moving path in the 3D world by projecting the stroke onto the walking surface, and presents the movement of the avatar and the camera in an animated manner. The avatar's direction is fixed to the tangent of the projected stroke. The user can draw a new stroke during the movement to modify the path, which is important because the far end of a stroke can easily get out of control. Figure 1 illustrates an example of path drawing navigation.

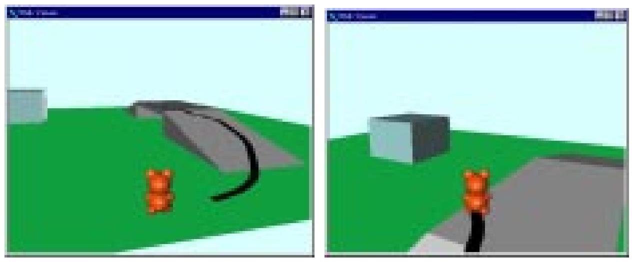


Figure 1: An example of path drawing walkthrough. The user draws the desired path directly on the screen (left), and the avatar and camera move along the projected path (right).

The user can draw either a long stroke specifying the detailed intermediate route to follow, or a very short stroke near the goal position. Long strokes are useful when the user is interested in how to get to the target position, while the user can conveniently specify the goal position and camera direction at once using short strokes. The user can also *turn* at the current position by drawing a short stroke at his foot in the intended direction.

This technique can work more effectively when the system is given a detailed structure of the virtual space. For example, it is possible to achieve the automatic avoidance of obstacles when the direct projection of the user's stroke intersects the obstacles. Climbing slopes and going through a gate can be detected by checking the polygon connectivity along the projected path (Figure 1).

A prototype system is developed using Inventor 2.1 on SGI graphics workstations. Automatic obstacle avoidance, slope climbing, and gate through are implemented and tested. However, these additional functions are turned off during the following evaluation.

EVALUATION

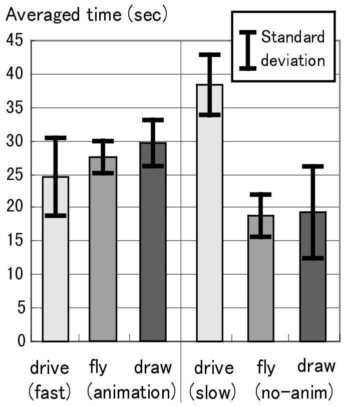
An experiment is performed to clarify the characteristics of path drawing against driving and flying techniques. Twelve subjects (computer science researchers) are instructed to get to the specified goal as rapidly as possible, navigating through a virtual space while avoiding obstacles. The subjects perform the task under the following six conditions, in a balanced order. In each condition, the camera is fixed just behind the avatar, and the avatar stops when it collides with an obstacle while traversing. A standard keyboard is used for “driving”, while a display integrated tablet is used for “flying” and “drawing”.

1) Driving (fast): the user controls the avatar using arrow keys. The left and right keys correspond to turn operations. Each movement occurs every 0.1 sec. (assumed to be the best setting).
2) Flying (animated): the user clicks the intended position directly, and the avatar smoothly moves toward the target in an animated manner. The moving speed is identical to that in 1).
3) Drawing (animated): the avatar moves along the drawn path in an animated manner. The speed is identical to that in 1).
4) Driving (slow): The same condition as 1), except that the each movement occurs every 0.2 sec.
5) Flying (no animation): flying without animation. The avatar instantly jumps to the target position after a click.
6) Drawing (no animation): path drawing navigation without animation. The avatar’s position and direction change instantly to the final state.

Figure 2 shows the averaged elapsed time to get to the goal, and Figure 3 shows users’ preferences and subjective impressions on how fast they finished the task under each condition. Among the first three conditions, driving is fastest and drawing was the slowest (statistically significant ($p<0.05$)). However, the users prefer drawing most, and driving least (not significant). The users also feel driving is the slowest (not significant). This suggests that drawing is comparable to flying in improving users’ subjective evaluation. Driving (slow) simulates the current tedious 3D navigation when the rendering speed is extremely low; flying and drawing without animation are tested as alternatives in the case. Flying and drawing are significantly faster than driving, while preference are almost equal (difference is not significant).

DISCUSSION

Path drawing can be used with any pointing device, but is most suitable for a pen-based or touch panel system. It is also possible to use this technique in immersive VR



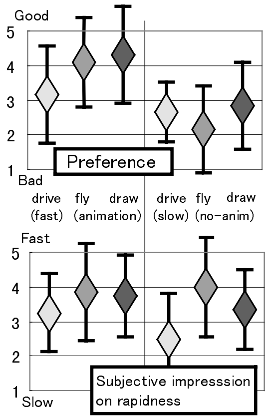


Figure 2 (left): Averaged time to get to the goal. A subject performed six tasks three times each, and the best among the three was selected and averaged.

Figure 3 (right): Subjective evaluations. The subjects were required to give relative scores ranging from 1 to 5 depending on their preferences of the techniques and their subjective impressions on rapidness.

environments with HMD and data gloves, where the user draws the path using the finger [2].

A limitation of path drawing is that it cannot be directly applied to completely free 3D movements (not constrained to a walking surface). Another limitation is that the avatar must be present on the screen in order for a path to be drawn at the avatar’s feet, but this problem may not be so serious because path drawing can naturally coexist with flying and driving in real applications.

CONCLUSIONS AND FUTURE WORK

We presented a technique for 3D virtual space walkthrough, in which the user specifies the intended path by drawing a free stroke on a virtual walking surface on the screen. This technique is superior to conventional *driving* in that the user does not have to continuously control the movement, and enhances *flying* by letting the user specify the route and direction at once. Experimental results show that the technique can be a good alternative at least for some users, improving subjective evaluation while maintaining a comparable operation speed.

Path drawing navigation is useful especially when the rendering speed is low or the communication delay is large, because the user can give detailed instructions to the computer at once, and the system can take time to perform time-consuming operations. We plan to apply this technique to remote robot control and wheelchair navigation, where the user draws strokes on camera images.

REFERENCES

1. Mackinlay, J., Card, S.K., Robertson, C.G., Rapid Controlled Movement Through a Virtual 3D Workspace. In *Proc. of SIGGRAPH’90*, pp. 171-176, 1990.

2. Pierce, J., Forsberg, A., Conway, M., Hong, S., Zeleznik, R., Image Plane Interaction Techniques in 3D Immersive Environments. In *Proc. of Interactive 3D Graphics'97*, 1997.