Finger Walking in Place (FWIP): A Traveling Technique in Virtual Environments

Ji-Sun Kim*[1], Denis Gračanin**[1], Krešimir Matković***[2], and Francis Quek†[1]

[1] Virginia Tech, Blacksburg, VA 24060, USA
[2] VRVis Research Center, Vienna, Austria

Abstract. In this paper we present a Finger Walking in Place (FWIP) interaction technique that allows a user to travel in a virtual world as her/his bare fingers slide on a multi-touch sensitive surface. Traveling is basically realized by translating and rotating the user's viewpoint in the virtual world. The user can translate and rotate a viewpoint by moving her/his fingers in place. Currently, our FWIP technique can be used to navigate in a plane but it can be extended to navigate in the third axis, so that the user can move to any direction in a 3D virtual world. Since our FWIP technique only uses bare fingers and a multi-touch device, finger motions are not precisely detected, especially compared with the use of data gloves or similar sensing devices. However, our experiments show that FWIP can be used as a novel traveling technique even without accurate motion detection. Our experiment tasks include finding and reaching the target(s) with FWIP, and the participants successfully completed the tasks. The experiments illustrate our efforts to make the FWIP technique robust as a scaled-down walking-in-place locomotion technique, so that it can be used as a reliable traveling technique.

Key words: Virtual environments, Finger-walking, Navigation, Traveling techniques, Multi-touch device

1 Introduction

One of the main interaction tasks in virtual environments (VEs) is navigation. The interaction techniques are based on input devices [3] specific to VE systems or simulators [4, 10]. Slater et al. [9] state that more natural locomotion enhances the sense of presence in VEs. We can assume that users may have the better spatial awareness because natural locomotion can give more sensory cues such as proprioception, vestibular apparatus, and kinaesthetic sense as well as vision that would help them get the spatial knowledge [8]. Several interaction techniques have been developed for different types of the navigation tasks, to effectively support natural locomotion in VEs [3].

---

* hideaway@vt.edu
** gracanin@vt.edu
*** matkovic@vrvis.at
† quek@vt.edu

One classification of traveling techniques is based on the overall interaction metaphor (e.g. physical locomotion, steering, route-planning, target-based, manual manipulation, and scaling) [3]. If we see the interaction techniques developed with a metaphor, especially "physical locomotion", we notice that the metaphor is straightforwardly transferred from the real world to the virtual world. For example, walking locomotion techniques, such as "real walking", "walking in place", and "simulated walking" [4, 9, 10], are only realized with physical walking motion. On the other hand, most of the interaction techniques with the steering metaphor (e.g. gaze, pointing, or steering props) are transformed to the flying locomotion technique. While the natural walking locomotion has been applied to get better spatial knowledge [5, 8], the users can get less benefits (due to physical body fatigue), compared with the use of the steering metaphor. On the other side, the steering metaphor based techniques provide easy ways to navigate a virtual world, but the effects on the spatial knowledge acquirement are smaller compared with the natural locomotion techniques [5, 8].

We provide a traveling technique, named Finger Walking in Place (FWIP), to take advantages of walking motion and steering control by fingers. It is a walking metaphor based interaction technique *transformed* from one of physical walking motion, i.e. "walking in place", to the finger walking motion. It is realized with bare fingers on a physical surface. Users can move forward and backward, and rotate in a virtual world. The users actually *feel* traveling in the virtual world by walking. It allows users to control virtual walking speed, i.e. by controlling the distance and the frequency of the finger movement, as the same way in their physical walking by controlling the distance and the frequency of the leg movement. The sense of presence is reduced with FWIP as "walking in place" does not provide the real vestibular cues [3]. However, the physical motion with FWIP is scaled-down, and consequently physical body fatigue should be diminished, compared with the physical locomotion technique.

If fingers represent human legs, walking motion using fingers should be able to translate the viewpoint in the virtual world. In a technical way, the viewpoint translation can be implemented with a single-touch device using the sliding motion of fingers as human legs slide on a treadmill because two fingers need not to touch the device surface at the same time. However, it is almost impossible to fully rotate one's hand on the surface to rotate the virtual world in place. There are two ways to control the viewpoint; physical and virtual techniques [3]. The former means that a user physically moves to translate or rotate the viewpoint, while the latter means that a user's body remains in place to control the virtual viewpoint. Since our technique is transformed from a physical locomotion technique and is realized with fingers only, we assume that the user's body remains in place during the navigation. Hence, although the full rotation with the user's body is possible in the physical technique, the virtual technique realized with fingers has a limitation in terms of the rotation angle. In order to allow the user to freely control both translation and rotation of the viewpoint with fingers, we need to consider different motions for the viewpoint rotation from the sliding walking motion. Consequently, we decided to use a multi-touch sensitive surface.

1.1 Multi-touch devices

Currently, the general multi-touch techniques are becoming more available for various device sizes from a palm size (e.g. handheld devices) to a table (e.g. Microsoft Surface) or wall size (e.g. SensitiveWall) displays. The applied technologies vary from simulating with computer vision to using resistive or capacitive touch screens. For example, one of our previous work [7] used an Augmented Reality technology using computer vision and a regular tabletop surface for navigation task in VEs. To the best of our knowledge, the first commercial multi-touch display device was Lemur [6] by JazzMutant in 2005, followed by Apple's iPhone [1] in 2007. Other devices followed, including Dexter by JazzMutant and iPod Touch [2] by Apple. As touch display technology advances from single-touch to multi-touch sensing, the number of interaction techniques is growing. For example, one of the most popular interaction techniques is a two-finger interaction technique applied to mobile handheld devices (e.g. iPhone and iPod Touch) for zooming and resizing tasks.

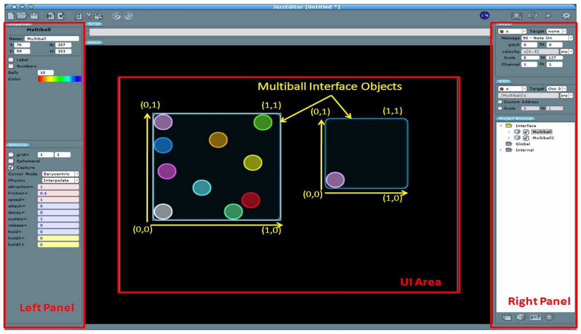


Fig. 1. JazzEditor: a property panel (left), UI area (center), and a message panel (right)

We chose Lemur [6] to implement our technique because it already comes with JazzEditor which provides a user interface (UI) development environment, while the Software Development Kit (SDK) for iPhone has been published on March 2008 after we implemented our technique with Lemur. Lemur is originally developed for MIDI applications. It provides a multi-touchable 12" display surface (800 x 600 pixels resolution) and completely customizable screens that are editable with JazzEditor. A designer can freely position UI objects, e.g. Container, Monitor, Multiball, Fader, Switches, and so forth, on the screen and easily change their properties such as size, color and inertia effect. The coordinate system used for the UI in JazzEditor is localized to each object placed on the screen. For example, Multiball, which we used to implement our technique, is an x-y rectangle area controller that can have up to ten controllable balls. Regardless of the size of the rectangle, the left-bottom corner is the origin (i.e. 0, 0) and the scale is up to 1, as shown in Figure 1(UI area). Lemur can recognize up to ten different spots at once. When a user places her/his fingers on the MultiBall object area, the ball nearest the spot touched becomes the one that she/he controls. The Lemur uses Open Sound Control (OSC) messages to communicate with a host computer over Ethernet. The properties of each UI object and the messages to the host computer can be customized and edited (Figure 1).

2 Design

VT-CAVE [11] installation driven by Linux machines is used in this study, including a Fakespace 4-wall CAVE display, which is 10x10 feet long, and an Intersense IS-900 VET tracking system for a head tracker, which is attached on the shutter glasses. The Lemur is placed on the table and its position is fixed in the middle of the CAVE immersive space (Figure 2). As long as the Lemur stays in place, its position acts as a persistent spatial reference.

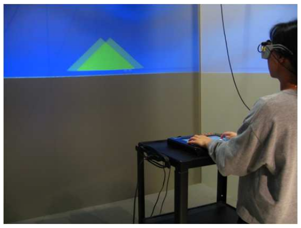
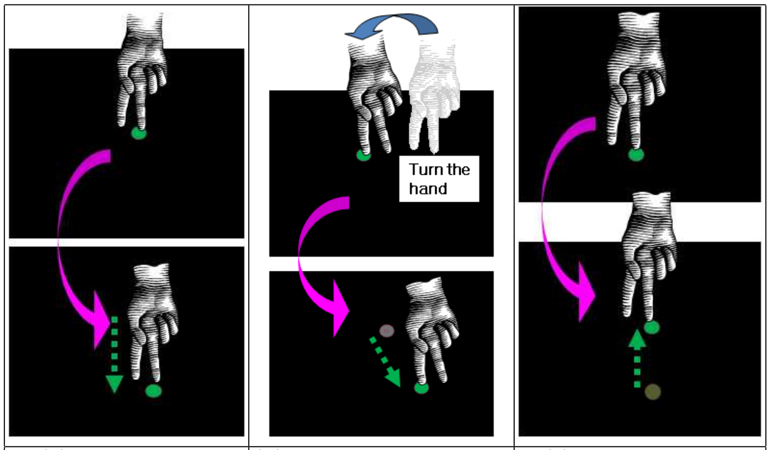


Fig. 2. Experiment setup for FWIP

2.1 Interaction Technique

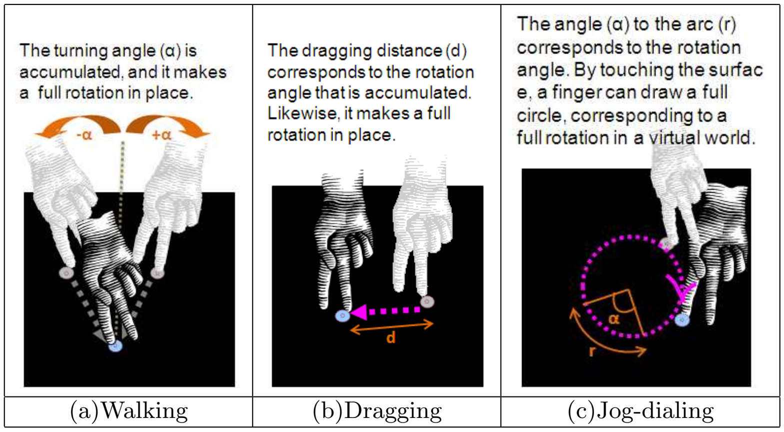
We provide three types of basic functions; walking in place to walk forward, backward and turn, rotation in place, and control of the walking speed.

Walking in place: In order to move forward (Figure 3(a)), one finger first touches the surface and slides down while touching the surface, as if human legs move on a treadmill. The finger then leaves the surface and immediately the next finger (or the same finger) touches the surface and slides down. Thus, a user can move forward by repeating this process. Figure 3(b) shows that the user can move forward, changing the viewpoint by turning the hand. When moving backward, one finger first touches the surface, slides up, and leaves the surface (Figure 3(c)). The user can freely walk forward and backward by changing the sliding motion.



(a)Walk forward (b)Change the direction (c)Walk backward

Fig. 3. Walking in place


The turning angle (α) is accumulated, and it makes a full rotation in place.

The dragging distance (d) corresponds to the rotation angle that is accumulated. Likewise, it makes a full rotation in place.

The angle (α) to the arc (r) corresponds to the rotation angle. By touching the surface, a finger can draw a full circle, corresponding to a full rotation in a virtual world.

(a)Walking (b)Dragging (c)Jog-dialing

Fig. 4. Rotation-in-place technique

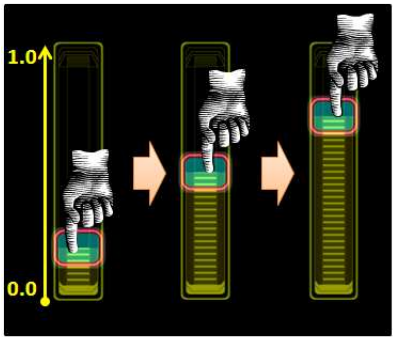


Fig. 5. Walking speed control example: speed-up

Rotation in place: Since it is difficult to turn the hand in place for more than 60–90 degrees, we designed three rotation-in-place techniques that can be performed on a different area separately from the walking area on the device surface. The rotation-in-place techniques are only used to rotate the viewpoint while the user's position does not change in the virtual world. In order to simultaneously translate and rotate the viewpoint, the user needs to use two hands, one for walking and another for rotation. The first one, Figure 4(a), mimics the walking motion for turning. The angle for the viewpoint rotation is changed by the angle which is formed by turning the hand. The second design, Figure 4(b), is based on the traditional mouse technique to change the user's perspective when the user navigates a 3D virtual world using a keyboard and a mouse. The longer dragging with a mouse makes the larger change for the user's perspective. Likewise, the dragging distance with a finger corresponds to the rotation angle in our technique. The third one, Figure 4(c), is designed from the finger technique of the traditional remote control device to slowly play a video. This technique can be more intuitively applied for the viewpoint rotation in that the viewpoint rotation corresponds to the circle angle. However, for this paper, only first two techniques are evaluated due to time constraint.

Control of the walking speed: Since the basic algorithm for walking is designed with a vector defined by two touch spots, the walking speed is basically controlled by the vector length (i.e. finger sliding length) in addition to the frequency of the finger movement. As shown in Figure 5, Fader object can be used for controlling the walking speed. The fader object is resizable, but its value is always ranged from (0.0) to (1.0), according the local coordinate system. When its control pointer is at (0.0), the walking speed is only controlled by the finger movement. With the current algorithm, as its control pointer is going up to (1.0), the walking speed is increasing up to twice the original speed.

3 Evaluation

3.1 Virtual World

The virtual world is designed to have no visual aid for finding a path to reach a target point. The space boundaries are fixed with the 40x40 meters and displayed using small cylinders. A green cone object represents the starting point. A red cone is placed behind of the starting point, so that we can examine which one a user would prefer, walking backward or rotation in place, to reach it. In order to trigger the user to find the target points randomly placed with the 120 degree angle (Figure 6), two lines are displayed at 10m and 20m from the starting point.

3.2 Pilot Study

We performed an initial study with five participants to evaluate the FWIP technique. The experimental tasks included finding one or two targets in near or far

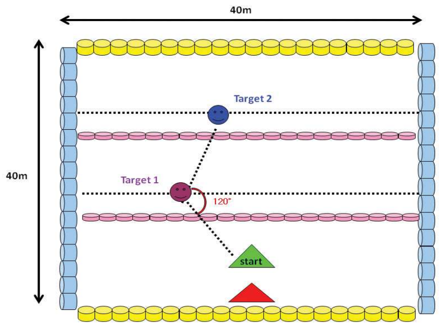


Fig. 6. Virtual World

distances from the starting point. In this study, the rotation-in-place technique was not provided. We also asked the participants to use a joystick-based traveling technique for the same tasks. They were asked to fill out the pre-experiment questionnaire which includes demographic questions, such as age, gender, frequency of playing computer or video games, and VE experience level. Since they were all novice VE users, they had to be trained with both user interfaces before starting the experiment. A post-experiment questionnaire obtained subjective preference of two interfaces as well as free-form comments. All of them were interested in using both user interfaces, but more preferred the joystick technique to complete the experiment tasks because flying locomotion is continuous and faster than walking locomotion. In addition, there was the latency issue between a user's input and the visual feedback. We also observed that most of participants occasionally moved out of the touch area while they were moving their fingers without looking at their finger movement to navigate the virtual world.

Enhancement

- **Latency:** To address the latency issue uncovered in the pilot study we made a change of the finger walking algorithm specific to the Lemur device to improve usability. While the original algorithm was based on the simple vector of the first and last touch spots per sliding, the improved algorithm is based on two consecutive touch spots. We also adjusted the angle of the vector formed by two spots. For this, we used the walking pattern acquired from the pilot user study. Figure 7 shows the partial data which the Lemur sends to the host computer. We analyzed x and y values and found that the y-range depends on the walking area size. We also found that the x-range is slightly changed even though users think they are walking straight to the target. These findings were applied to change the algorithm for the second experiment.
- **Rotation techniques:** We used our two rotation-in-place techniques after the first user study was conducted. While the first user interface is designed with one Multiball object (Figure 8(a)), the second interface is designed with

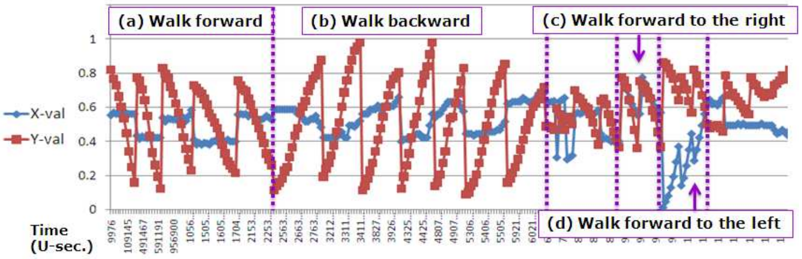
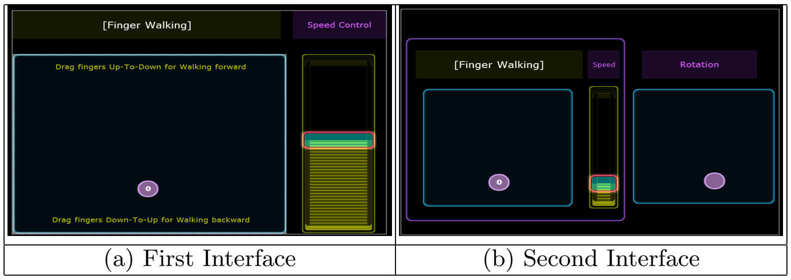


Fig. 7. Finger walking pattern; (a)Walk forward: per sliding, y–value is changed over the 0.4–range from the bigger number to the smaller one. (b)Walk backward: per sliding, y–value is changed over the 0.4–range from the smaller number to the bigger one. X-value is not considered in both (a) and (b) unless x-value is over 0.02–range per sliding. (c)Walk forward to the right: per sliding, both y–value and x–value are changed to the same direction. (d)Walk forward to the left: per sliding, y–value and x–value are reversely changed. The virtual movement direction is determined with the vector of y–value and x–value.

two objects for walking and rotating respectively (Figure 8(b)). The second one allows a user to translate and rotate the viewpoint simultaneously with two hands. We also observed that users preferred shorter sliding in the first study. Because of the Lemur's local coordinate system, the smaller size of the Multiball object makes the longer movement with the same length of sliding. Hence, we reduced the object size (i.e. touch area) in the second interface.



(a) First Interface | (b) Second Interface

Fig. 8. UI change from (a) walking only to (b) walking and rotation

- **Tactile constraint:** In order for users not to touch out of the Multiball object area during navigation, we attached a tactile constraint fit to the walking areas on the Lemur surface. We first used the sticky tape, but later changed to the thin wire (Figure 9) because it is more reusable and easily detached than the sticky tape from the surface. This tactile constraint prevents users from leaving out of the walking areas.

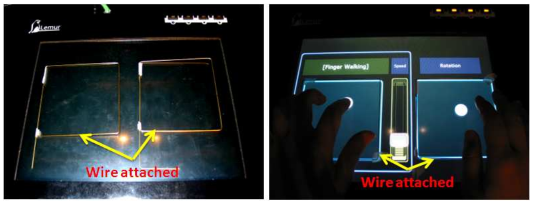
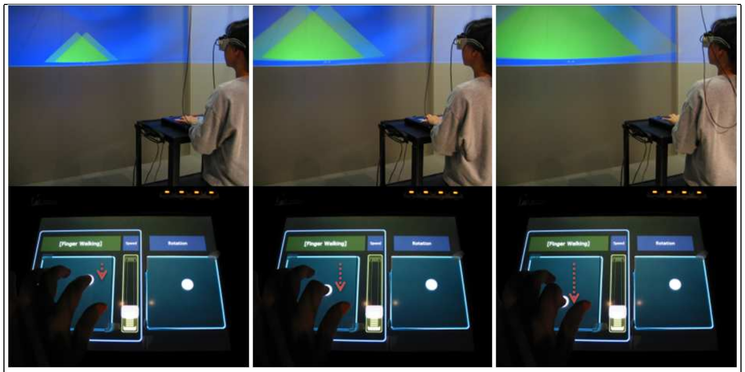


Fig. 9. Wire fit to the Multiball objects on the Lemur

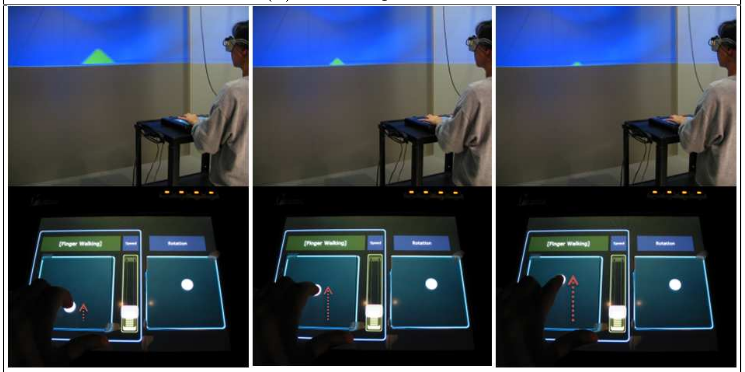
3.3 Formative Study

Twenty five users participated in the second study. Five of those users evaluated our initial implementation redesigned from the first study, and the rest of them evaluated the stabilized interface. The first nine users evaluated the walking–motion rotation technique and the remaining users evaluated the dragging rotation technique. The study procedure was pretty much same as in the first study except for the actual tasks. We asked participants to reach the red cone object, go back to the green cone (i.e. the start point), reach the target(s), and then go back to the green cone again in order. Thus, each task has four sections in total. All participants found the target(s) in seven different tasks; (1) go straight and find one target at 25m, (2) go forward to the right to find one target at 22m, (3) go forward to the left to find one target at 22m, (4) go forward to the right to find one target at 35m, (5) go forward to the left to find one target at 35m, (6) go forward to the right to find two targets at 20m and 35m, and (7) go forward to the left to find two targets at 20m and 35m away from the start point. We took videos of users' motions during the experiment. We focused on usability of FWIP without comparison with other types of techniques. Figure 10 shows user interactions using "walking forward", "walking backward", and "rotation in place" techniques.

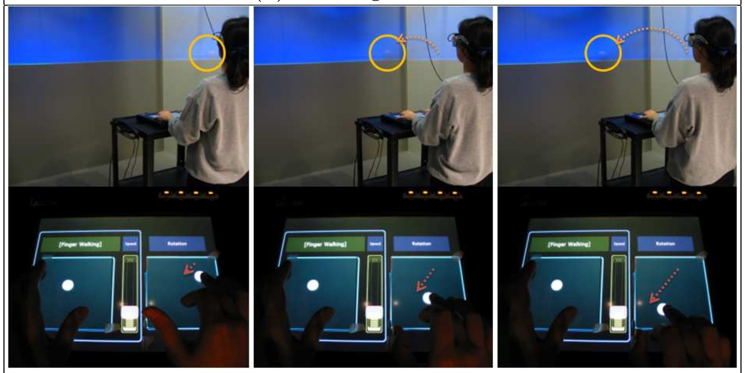
Subjective Result and User Feedback Users showed various but similar finger movements. For long distance, they usually used two fingers of one hand. For short distance, they carefully used one finger-sliding. One user used two hands to walk forward with two fingers, and she moved forward very fast. Some users who play 3D games a lot used both walking and rotation techniques with two hands simultaneously to reach the target. We observed that the tactile constraint attached (refer to Figure 9) on the surface was very helpful for users to reduce the unnecessary context-switching between the surface and the CAVE screens. For the data analysis, we only considered twenty participants since we stabilized the second interface after five users tested it. They evaluated the techniques, walking forward, walking backward and rotation techniques in categories of Satisfaction, Fastness, Easiness and Tiredness using a 1 to 7 scale. For the first two categories, the higher value is better, but for the rest categories, the lower value is better. Figure 11 shows the average ratings as follows in the order of walking forward, walking backward, and rotation techniques: Satisfaction



(a) Walking forward



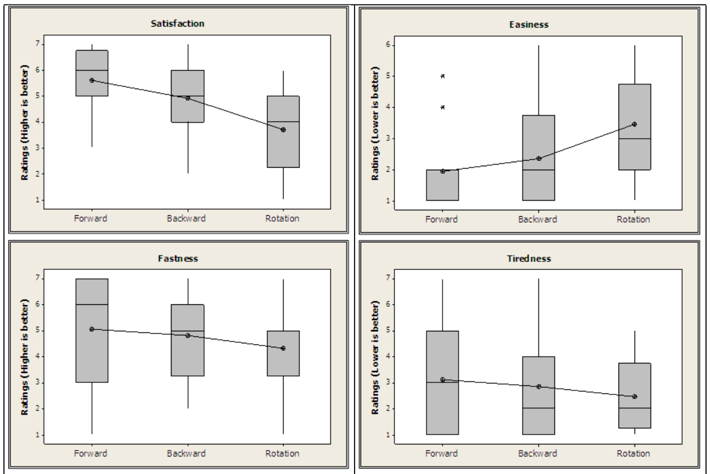
(b) Walking backward



(c) Rotation in place('Walking')

Fig. 10. User Interaction

((5.6), (4.9), (3.7)), Fastness ((5.1), (4.8), (4.3)), Easiness ((2.0), (2.4), (3.5)), Tiredness ((3.1), (2.8), (2.5)).



(a) Satisfaction and Fastness (b) Easiness and Tiredness

Fig. 11. Subjective Results

We used a one-way ANOVA to analyze the subjective results of three techniques in four categories. There was a significant difference between walking forward and rotation in two categories, Satisfaction ($p=0.000 < .05$) and Easiness ($p=0.004 < .05$), respectively. Most of the participants felt very comfortable with the walking forward technique. According to their comments, they gave the higher rate to it because they used a lot this technique to complete the tasks. Compared with walking techniques, rotation techniques were not quite natural to participants. For example, with the 'walking' rotation technique, they were not aware of how much they rotated until they got the visual feedback. Sometimes they over-rotated or under-rotated the viewpoint in place than they thought. Since the users did not look at their hands over the surface, in order to estimate the rotation angle, they might have to rely on how long they are continuously touching the surface, rather than how much angle their hands turn or how much length they drag. They were slightly less tired in 'dragging' (average rate=2) than in 'walking' (average rate=3) for rotation. Based on their demographic information, we can assume that the reason is because 'dragging' technique was quite familiar to the users who play computer games a lot. They also commented that it was not really tiring their fingers. Rather, it was boring to keep on walking by moving fingers. Since the walking backward is very rare even in the real world, they used the walking backward technique only in the case that they already knew the position of the target object behind them.

4 Summary and Future Work

This paper presents FWIP as a novel interaction technique with bare fingers and a multi-touch device for navigation in VEs. FWIP allows users to navigate in VEs as their fingers slide on a multi-touchable surface. They can change the viewpoint by turning the hand, or by using a rotation-in-place technique. Our experiments showed that FWIP can be used as a traveling technique without accurate motion detection. The experiments illustrated our efforts to make the FWIP technique robust as a scaled-down walking-in-place locomotion technique, so that it can be used as a reliable traveling technique.

In future work we will provide the third rotation technique, 'jog-dialing', and compare it with the other two rotation-in-place techniques. If there is no significant difference among three rotation-in-place techniques, we will allow users to use any technique while they are traveling in the virtual world. In order to thoroughly compare FWIP with the most common locomotion technique in VEs, i.e. a joystick-based locomotion technique, we will use a more complex virtual world, such as a maze, where users have to frequently change their viewpoint. Since we used an open plane as a virtual world in the first user study, participants preferred using a joystick due to its continuous control characteristic. For the open plane, the continuous controller would be more appropriate used rather than the discrete controller (i.e. FWIP) because they don't have to frequently change their viewpoint to reach the targets. We will objectively measure the task completion time and the total movement distance to examine which one shows the better performance in the complex virtual world.

References

1. Apple: iPhone. `http://www.apple.com/iphone/` [Last accessed Jan. 2008]
2. Apple: iPod–Touch. `http://www.apple.com/ipodtouch/` [Last accessed Jan. 2008]
3. Bowman, D.A., Kruijff, E., LaViola, J.J.Jr., Poupyrev, I.: 3D User Interfaces: Theory and Practice. Addison-Wesley (2005)
4. Darken, R.P., Cockayne, W.R., Carmein, D.: The omni-directional treadmill: A locomotion device for virtual worlds. In Proceedings of UIST, (1997) 213–221
5. Iwata, H., Yoshida, Y.: Path Reproduction Tests Using a Torus Treadmill. Presence, 8(6) (1999) 587–597.
6. Jazzmutant: Lemur. `http://www.jazzmutant.com/` [Last accessed Jan. 2008]
7. Kim, J., Gracanin, D., Signh, H.L., Matkovic, K., Juric, J.: A Tangible User Interface System for CAVE Applications. In the Proc. of IEEE Virtual Reality, (2006) 261–264.
8. Peterson, B., Wells, M., Furness III, T.A., Hunt, E.: The Effects of the Interface on Navigation in Virtual Environments. In Human Factors and Ergonomics Society 1998 Annual Meeting, (1998) 1496–1500(5).
9. Slater, M., Usoh, M., Steed, A.: Taking steps: The influence of a walking metaphor on presence in virtual reality. ACM Transactions on Computer Human Interaction (TOCHI), 2(3) (1995) 201–219.
10. Templeman, J.N., Denbrook, P.S., Sibert, L.E.: Virtual locomotion: walking in place through virtual environments. Presence, 8(6) (1999) 598–617.
11. Virginia Tech: VT–CAVE. `http://www.cave.vt.edu/` [Last accessed Apr. 2008]