LLCM-WIP: Low-Latency, Continuous-Motion Walking-in-Place

Jeff Feasel **Mary C. Whitton** **Jeremy D. Wendt** *

Department of Computer Science – University of North Carolina at Chapel Hill

ABSTRACT

Walking-in-place techniques for locomotion in virtual environments typically have two problems that impact their usability: system latency (particularly troublesome when starting and stopping locomotion), and the fact that the change in the user's viewpoint may not be smooth and continuous. This paper describes a new WIP interface that improves both latency and the continuity of synthesized locomotion in the virtual environment. By basing the virtual avatar motion on the speed of the user's heel motion while walking in place, we create a direct mapping from foot-motion to locomotion that is responsive, intuitive, and easy to implement. In this paper, we describe the technique, analyze its starting and stopping latency, and provide experimental results on the suppression of false steps and general usability of the system.

KEYWORDS: virtual locomotion, virtual travel, walking-in-place, virtual environments, 3D User Interfaces

INDEX TERMS: H.5.2 [Information Interfaces and Presentation]: User Interfaces; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality

1 INTRODUCTION

The quality of a virtual-environment (VE) locomotion interface has a significant impact on the level of presence a user feels in a virtual environment [1, 2] and the interface affects the way a user moves [3]. Although head-tracked real walking in VEs consistently evokes user behavior most like walking in the real world, locomotion by really walking is impractical in large-scale VEs, because the tracked space must be as large as the virtual space. Scaling high-precision tracking systems to arbitrarily-large sizes is expensive; wide-area tracking systems do not provide sufficient precision for a first-person display [4]. Because of these problems, the many VEs that require locomotion in large virtual scenes employ interfaces through which users can move their avatar (and viewpoint) to anywhere in the scene while remaining essentially stationary in the real world.

Although various stationary-user locomotion interfaces have been proposed, previous research demonstrated that walking-in-place (WIP) is more presence-inducing than pointing interfaces [1, 2]. One of WIP's greatest strengths is its similarity to real walking: The user controls their motion by moving their legs. From experience with our own WIP systems and others described in the in the literature [4, 5, 6, 7], we have identified two problems that impact WIP usability: system latency (particularly troublesome when starting and stopping movement), and the fact that the change in the user's viewpoint may not be smooth and continuous. Latency in visual feedback decreases the user's ability to precisely control their speed and stopping-position.

* e-mails: {feasel, whitton, jwendt}@cs.unc.edu

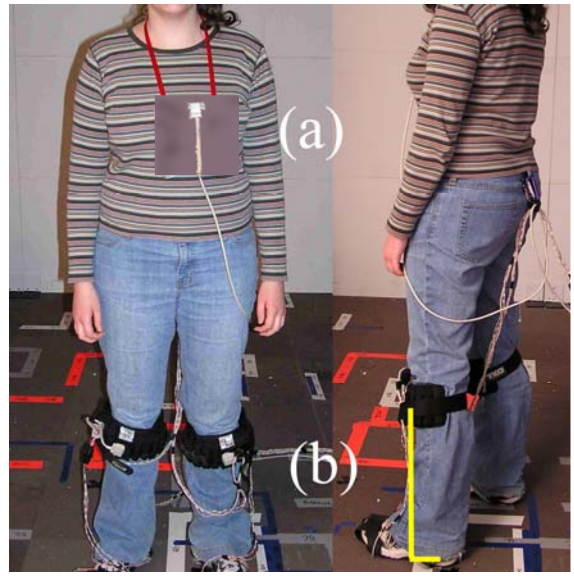

Figure 1: The LLCM-WIP hardware. (a) Chest-orientation tracker (b) Foot trackers (rigid offset from mount-point to heel shown),

In some systems viewpoint-movement is implemented as one or a series of discrete increments triggered by the detection of a step. This can lead to motion that feels jerky. A more natural motion-synthesis method is to distribute the locomotion over several frames, making sure that the change is smooth and the oscillations during sustained walking are not too great.

Our system – called the *Low-Latency, Continuous-Motion Walking-in-Place (LLCM-WIP) System* – has four design goals. The following are the goals and how they were realized:

1. **Low latency.** Experimental average of 138 ms when starting and 96 ms when stopping – less than $^1/_8$ of a gait-cycle.
2. **Smooth locomotion between steps.** Smoothing of the heel-speed signal eliminates pauses in motion that occur during and between steps.
3. **Continuous control of locomotion speed within-step.** Locomotion speed is updated continually during the step (with each frame), based on instantaneous vertical heel-speed.
4. **Incorporation of real-world turning and short-distance maneuvering into virtual locomotion.** Subtracting a constant offset from the vertical heel-speed creates a threshold value below which heel-motion will not be interpreted as locomotion. So steps that are low to the ground or taken very slowly will not register as locomotion. Thus the system can use real-world turning – actually rotating the body through a series of short turning-steps – to control the avatar's orientation independently of the virtual locomotion speed. This also allows virtual locomotion to be combined with short-distance maneuvering in the real-world, without the small steps being interpreted as virtual locomotion.

Paper Organization: The remainder of this paper is dedicated to describing the LLCM-WIP system and its effectiveness. The next section, 2, cites previous work on WIP systems. Section 3 describes the components of LLCM-WIP. Section 4 presents experimental measurements of latency and drift-suppression effectiveness taken on real users of the system. Section 5 describes a study that tests the effectiveness of LLCM-WIP compared to other locomotion methods. In the final section, 6, we discuss our conclusions on the system's effectiveness and direction for future development

2 PREVIOUS WORK

Locomotion in virtual environments is a broad topic ranging from devices such as treadmills and trackers, to the psychology of the visual-motor system, to signal processing. The topic is covered in some depth in [8], which includes a case study on walking-in-place locomotion interfaces. Our discussion here will focus on a subset of the previous walking-in-place systems that are most relevant to LLCM-WIP.

2.1 Latency

Slater and his team in London developed an early WIP implementation for virtual environments [1]. Their implementation employed a neural net to recognize patterns of head movement that correspond to walking-in-place. Based on the fact that users are more confused by making false steps that cause them to move where they didn't intend to go than they are confused by motion not starting promptly, the system was set up so that if the user was still, it required four positive "step" signals from the neural network before motion began, and, if the user was moving, it required two full cycles of "no step" signals before movement stopped. The extra steps before stopping could cause the user to overshoot their target stopping-location, sometimes resulting in collisions and visual inter-penetration of user avatars and objects in the scene [2].

The aim of the LLCM-WIP system is similar to that of the Gaiter technique, developed at the Naval Research Laboratory by Jim Templeman and his team [5, 9, 10]. Gaiter allows users to generate virtual locomotion by walking-in-place in combination with real-world maneuvering over short-distances [5]. Gaiter analyzes the horizontal components of knee motion to discriminate between the real and virtual steps. In order to discriminate, Gaiter must wait for the knee to reach the point of maximal extent before it recognizes a virtual step – causing one half-step's latency (about 400ms when walking at a moderate pace). Later work by the Gaiter team suggests a new system for reducing the half-step's latency [9], but details are not yet in print.

A recent WIP system from Yan et al. uses the speed at which the knee lifts at the outset of the step to estimate the user's intended locomotion speed [7]. They state that this system has lower latency than previous ones but do not provide an analysis.

The lowest-latency leg-driven locomotion system that we know of is a *seated* interface called Pointman [10]. In Pointman, the user slides a set of reciprocating foot-pedals forward and back. The speed at which the pedals are pushed maps to instantaneous locomotion speed. Although leg-driven, Pointman requires the user to be seated and use a joystick for turning – thus it is not strictly a walking-in-place system in the same sense as Gaiter or ours. For instance, the user doesn't physically rotate their body to turn, which is an important part of the naturalness of a full walking-in-place system.

Both Pointman and LLCM-WIP produce near instantaneous motion for the same reason: they are *continuous-motion* walking-in-place interfaces. There is a direct mapping from the instantaneous speed of the heel's motion to the virtual locomotion. The only delay in the controls comes from signal processing and system latency, which is small relative to the length of a gait cycle. The user does not need to complete a step (or even half a step) before the resulting virtual locomotion begins.

2.2 Synthesis of Virtual Locomotion

We found that previously published descriptions of WIP interfaces (even our own) do not describe in detail exactly how movement detected by some sensing mechanism causes changes in the rendering viewpoint. As a result, and because we think it important for people to know what has been tried and abandoned, most of the following discussion is based on personal communications with system developers.

Slater [11] remembers that in the original neural network WIP system the viewpoint was moved forward by the distance of a full step all at once—i.e., the viewpoint jumped ahead the distance of one full step between frame n and frame $n+1$. In Figure 2a we have represented the locomotion-speed curve of this method as a series of impulse functions. The WIP implementations of [2, 3] used this method of moving the viewpoint.

When revisiting the neural network system for a later study performed in Slater's laboratory, Razzaque implemented two new motion types [12]. When the viewpoint movement is divided equally over several frames, the speed can be modeled as a box function (Fig 2b). Motion starts immediately when the step is detected, a positive characteristic, but it also ends abruptly, which is unnatural. Users experience this style of sustained-walking movement as a sequence of forward glides (during which the speed is a non-zero constant) separated by short pauses (zero-speed).

To overcome the unnaturalness of gliding, Razzaque tried modeling velocity as a saw-tooth function (Fig 2c). The saw-tooth retains the positive characteristic of quick onset of motion, but eliminates the abrupt stop that the box function has by linearly interpolating speed back to zero over several frames. The team compared the three types of motion informally and judged the saw-tooth function as most natural-feeling, using it in the work reported in [6]. None of the three patterns of velocity discussed here is a particularly good approximation of the nominally sinusoidal velocity curve characteristic of the rhythmic phase of human walking [13].

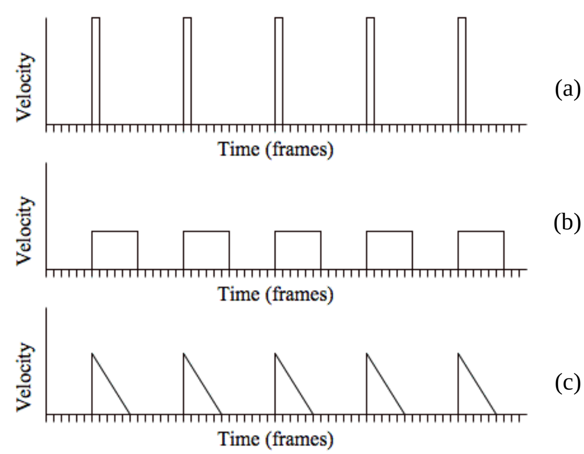

Figure 2: Models of viewpoint movement speeds used by early walking-in-place systems.

Both Gaiter and Yan et al.'s system generate locomotion based on the movement of the knees [5, 7]. Both describe employing a metric to calculate the distance. Both papers state that the change in the viewpoint is distributed across several frames but do not describe the shape of this distribution. Templeman and Sibert [14] described their speed curves as looking much like Figure 3(c) — claiming to achieve the smoothness and continuity that are also our goals for the LLCM-WIP system.

Of all of these locomotion systems Pointman is the only one that may be considered continuous, since it directly maps pedal movement to avatar/viewpoint movement. The other systems detect a pattern of motion of some part of the body and identify that pattern as a step. When a step is identified, the amount of motion corresponding to a step is applied. Thus each step is a discrete event, and the pattern-detection mechanism must necessarily wait until enough of a step motion has been performed by the user to fully identify a step.

The Low-Latency, Continuous-Motion Walking-in-Place (LLCM-WIP) system – described in the remainder of this paper – improves on previous work in regard to lower latency and continuous motion. In LLCM-WIP, we consider user input as a continuous signal, not a series of discrete step events. In fact, LLCM-WIP has no concept of a step. It simply maps the speed of the heels to locomotion speed, employing minimal signal processing instead of pattern recognition. The result is a responsive system that produces smooth and realistic locomotion.

3 Description of the Low-Latency, Continuous-Motion Walking-in-Place System

The implementation of LLCM-WIP is quite simple: It is a sequence of simple signal-processing operations on heel-tracking data. Section 3.1 gives an overview, and the following subsections describe the operations.

We developed LLCM-WIP to meet four important goals, each of which is fundamental to the experience of walking:

- Low latency
- Smooth locomotion between steps
- Continuous control of locomotion speed within each step
- Incorporation of real-world turning and short-distance maneuvering into virtual locomotion

We will point out how the components described in the following sub-sections, along with careful choice of their "tuning" parameters, accomplish the above goals.

3.1 The System, in Brief

The LLCM-WIP system consists of the following stages. Figure 3 shows the signal at several points in the process.

1. Track heel position
2. Isolate vertical component (z)
3. Numeric differentiation
4. Invert negative parts (absolute value)
5. Smoothing (low-pass filter)
6. Subtract a constant offset
7. Clamp negative values to zero
8. Combine (sum) signals from Left and Right feet
9. Scale to user-specific walking speed
10. Apply orientation (chest-direction with optional joystick)
11. Integrate locomotion velocity to get avatar-position offset
12. Sum position offset with real-world head position
13. Render viewpoint from combined virtual/real-world position

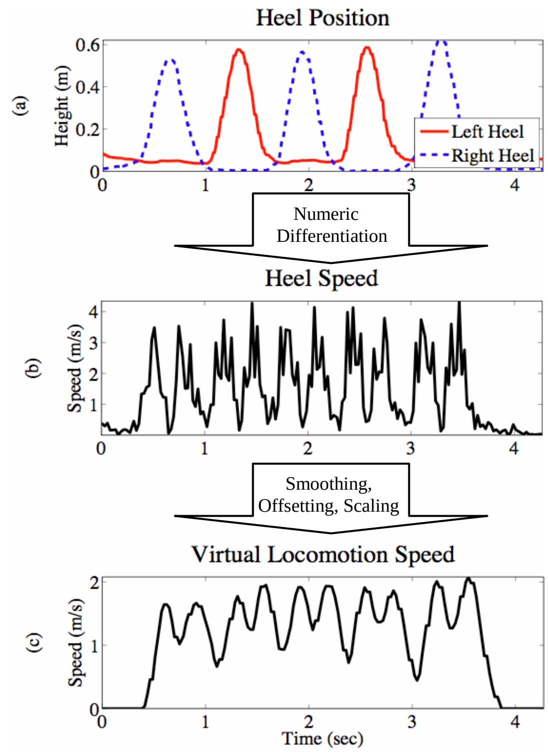

Figure 3: System overview. We track the vertical position of the user's heels. Through numeric differentiation, we obtain the heel speed. After some signal processing operations, we have a virtual locomotion speed.

3.2 Vertical Speed of the Heels

Whereas previous systems focused on knee motion, we start with the vertical motion of the user's heel. To obtain an estimate of the heel's vertical speed, we track its position and perform numeric differentiation on the vertical component to yield vertical velocity. We then invert the negative portions to yield a non-negative signal: the vertical heel-speed. Figure 3b shows the combined vertical heel-speed, summed over the left and right feet.

Because we employ a magnetic tracker and our lab floor is metal, we are unable to place trackers directly on the user's feet due to unacceptable distortion in the reading. Instead, we place trackers on the shins, just below the knees, and perform a simple, rigid-body transform to estimate the user's heel location. The offset from the shin-mounted tracker to the bottom of the heel is roughly fixed, regardless of the ankle pose. Even an approximate offset serves quite well: Instead of measuring each individual user's shin length, we use the same (typical) value for all users.

The output of this stage is a pair of scalar signals: the vertical speed (a positive quantity) of the left and right heels. Sections 3.3 and 3.4 will explain how these signals are combined and mapped to locomotion speed.

A direct benefit of considering only the vertical component of the heel's velocity is that real-world turning and short-range maneuvering can be distinguished from intentional virtual locomotion. For instance, nearly all natural, in-place turning involves vertical heel motions on the order of several centimeters or less, occurring at a speed that is much slower than that of actual locomotion steps. Users quickly learn to perform in-place turns and short, real-world steps with little vertical motion of the heels. To prevent incidental vertical motions from producing unintended virtual locomotion (i.e. drift), we subtract a fixed offset from heel speed, which effectively eliminates small or slow vertical motions of the heel (see section 3.3).

3.3 Synthesizing Locomotion

We turn heel speed into locomotion speed by process of smoothing and subtracting a fixed offset. Summing the smoothed, offset signals of the two heels produces a signal that will be scaled appropriately to become locomotion speed.

As can be seen in Fig. 3b, the speed derived by numeric differentiation is a very noisy signal. We employ a 4th-order, low-pass, Butterworth filter with a cutoff frequency of 5 Hz to smooth the speed signals of the left and right feet. This particular filter was chosen because it is easy to implement, but other low-pass filters may be used in its place.

An effect of the smoothing is to eliminate pauses in the heel's vertical motion that occur mid-step (when the heel reaches the top of its arc) and between steps (when one heel returns to the ground and the other has not yet been lifted). Without filtering, the locomotion speed would drop to zero at these points, producing choppy motion that is unrealistic.

However, in choosing the cutoff frequency of the low-pass filter, we are faced with a tradeoff between smoothness of locomotion and low latency: The lower the cutoff frequency, the smoother the locomotion, but the slower the system responds to user input. We chose a cutoff frequency of 5 Hz, after comparing it to several others (see Fig. 4). The smoothing adds about 100 ms of latency to both starting and stopping, though the offsetting operation, applied next, will serve to reduce the stopping latency.

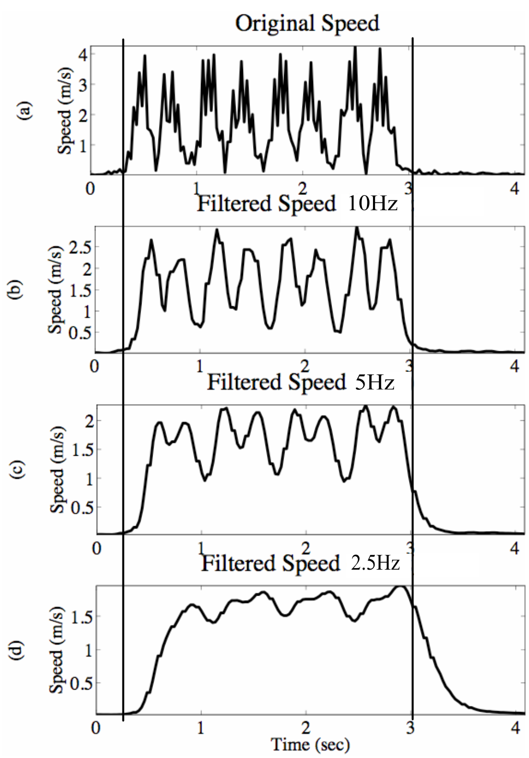

Figure 4: Plot of smoothed heel-speed vs. time during a short walk, shown under several choices of cutoff frequency. **(a)** Unfiltered. **(b)** 10 Hz cutoff. **(c)** 5 Hz cutoff (our choice). **(d)** 2.5 Hz cutoff. Vertical lines added to demonstrate increased latency with higher filter cutoffs.

The next stage in signal processing is subtracting a constant offset. This minimizes virtual-locomotion drift caused by vertical heel motion from in-place turning and short-distance, real-world maneuvering. To do this, we subtract a constant offset from the output of the smoothing stage and clamp all negative values to zero. The offset is shown as a horizontal line in Figure 5a.

Care must be taken in choosing the offset constant. If set too high, the output speed approaches zero between steps and at mid-step. If set too low, virtual-locomotion drift will not be adequately controlled during turning and local maneuvering. After some testing, we found that an offset constant of 0.2 m/s strikes a good balance.

The offset constant has a mixed effect on latency. It slightly increases the latency at the initiation of locomotion – since the heel speed must exceed a threshold before locomotion begins. However, the offset reduces the latency at the cessation of locomotion, countering the latency introduced by smoothing. See section 4.2 for an analysis of latency.

The smoothing and offset operations are performed on each heel separately; afterwards the two signals are summed to yield a single 1-d signal. This signal will be scaled to produce locomotion speed (section 3.4) and then oriented to produce locomotion velocity (section 3.5).

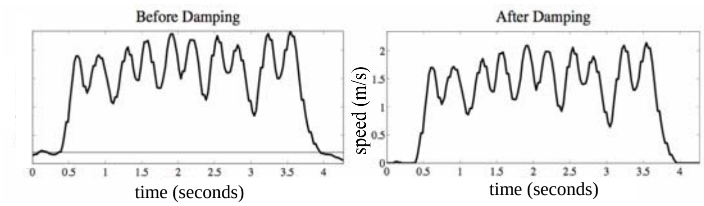

Figure 5: The offsetting operation suppresses virtual locomotion drift. Subtract the offset constant (horizontal bar) and clamp negative values to zero.

3.4 Scale of Locomotion Speed

The smoothed, offset heel-speed shown in Fig 3c is converted to locomotion speed through multiplication by a scale factor that is determined on a person-by-person basis. We measure the user's average real-world locomotion speed while walking at a comfortable pace. We then scale the output of LLCM-WIP so that, on average, a user's virtual-locomotion speed matches the speed of their real-world locomotion.

The user-specific speed calibration procedure is simple: the user walks between two tape-marks on the floor in the real world, about 6 m apart. The user is instructed to reach a comfortable, sustained walking pace before passing the first tape-mark. The time taken to traverse the 6 m is measured by stopwatch to obtain a user-specific measurement of sustained walking speed. The user then performs a sustained walking-in-place motion of about the same pace using the LLCM-WIP system, and the sustained virtual locomotion speed that results is scaled to match that of the real-world estimate.

In practice we find that this calibration procedure is not strictly necessary, and fixing the scale-factor to a single typical value works fine in day-to-day use.

3.5 Direction of Locomotion

Virtual locomotion is a velocity, and thus far we have described only the generation of locomotion speed. This section deals with the separate problem of what direction to apply the virtual locomotion.

For a WIP system to be useful, it must allow virtual locomotion in a direction that is independent of the view direction: The user must be able to walk a straight path while looking around. Thus LLCM-WIP uses the orientation of the user's chest to control the direction of the virtual-locomotion. This effectively allows the view direction and locomotion direction to be decoupled.

3.6 Optional Inclusion of Side-Stepping via Joystick

Chest-directed locomotion – while sufficient for many tasks – does not allow for side-stepping and backing up, which are helpful when moving within a small or crowded space. Thus the LLCM-WIP system may be augmented by using a joystick to modulate the direction of the locomotion.

In this case, the magnitude of the joystick's displacement is ignored; only the direction of displacement is used. The joystick's direction is applied *relative* to the chest's orientation, thus locomotion direction is the product of chest orientation and joystick direction. For example, pressing forward on the joystick will move in the direction the chest is facing, and pressing left on the joystick will move 90-degrees counter-clockwise from the direction the chest is facing.

The joystick is optional; if present it makes maneuvering within close-quarters in the VE easier. Without it the system uses chest-directed motion alone, which is equivalent to keeping the joystick in the forward position.

Chest-directed locomotion (with or without joystick modulation) is not a new concept, nor is it necessarily the best means of controlling locomotion direction. Chest-direction is simple to implement, but any alternative means of specifying direction could be combined with LLCM-WIP's speed output. For instance, one might use the direction-specifying component of Gaiter (knee-lift or heel-kick direction [5] in the horizontal plane) and then use LLCM-WIP to determine the magnitude of locomotion.

4 EXPERIMENTAL EVALUATION

The previous section described LLCM-WIP. This section provides a quantitative analysis of two important measures of the system: latency and the effectiveness of virtual-locomotion-drift suppression.

4.1 Latency Analysis

Latency within LLCM-WIP is affected by several system components: the smoothing filter, the offset constant, and the VE-system latency (tracker latency, rendering time, etc.). The first two may be altered by tuning LLCM-WIP's parameters: cutoff frequency and offset constant. In this subsection, we describe how we analyzed the latency caused by the values we chose for these tunable parameters.

We define **starting latency** as the delay between the moment the foot leaves the ground and the moment virtual locomotion begins to be generated. We define **stopping latency** as the delay between the moment the user's foot touches the ground (and no further steps are taken) to the time when locomotion speed drops to zero.

As previously mentioned, both of these latencies are affected by our choice of cutoff frequency and damping constant. However, a further factor – the the pace of the user's motion – affects the degree to which these values impact latency. Thus in the following study, we varied the pace at which we asked the participant to walk.

We instructed 5 participants to walk at three different pace levels (slow, medium, and fast) using the LLCM-WIP system. The participants were placed in an immersive virtual environment via a head-worn display. The virtual environment was kept as simple as possible: a sidewalk between a building and the street, with several trucks and people to provide scale.

After entering the VE for the first time, the participant was given a period of practice with the system. The participant then performed 12 trials of walking and stopping. For each trial, the participant began at a stop, walked 6 steps in a straight line, and then came to a full stop again.

These 12 trials were evenly divided into slow, medium, and fast paces. While we instructed the participant to go slow, medium, or fast in each trial, we left it up to the participant to set the pace according to their own interpretation. In each trial we measured the starting and stopping latency. For stopping, we also measured the distance traveled during the stopping delay.

Figure 6 shows the break-down of mean starting and stopping latency of users at the three different pace levels.

The longest starting latency occurred in the slow-paced trials, which is understandable since a slower step speed will take longer to exceed the offset threshold. However the slow, medium, and fast paced trials are not well-separated in terms of latency: there is significant overlap. The average starting latency over all the 60 trials (combining all subjects and paces) is 138 ms, with a standard deviation of 41.6 ms.

Walking pace has a mixed effect on stopping latency, and there is no significant separation in latency among the three pace groups. The average stopping latency over all 60 trials (combining all subjects and paces) is 96 ms, with a standard deviation of 45.2 ms. During that interval of stopping latency, the participants traveled an average distance of only 6 cm (standard deviation 3.0 cm) before coming to a full stop.

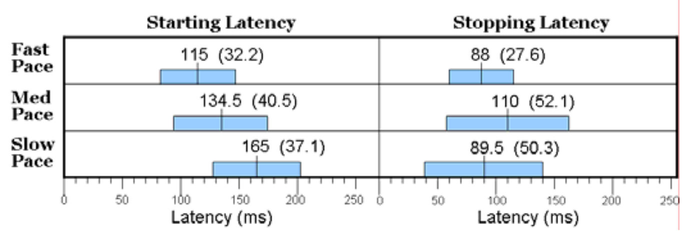

Figure 6: Starting and Stopping Latency at Various Paces. Latency mean/s.d. is shown for each of 3 paces, combined over all trials from all users. Each box-plot shows the mean plus or minus one standard deviation. Standard deviation (ms) is given in parentheses above. The overall mean starting latency was 138 ms (s.d.: 41.6 ms), and the overall mean stopping latency was 96 ms (s.d.: 45.2 ms).

4.2 Suppression of Virtual-Locomotion Drift

The user turns in the virtual world by physically turning the body in place in the real world. Likewise, the user may make short-distance real-world maneuvers. So it is important that the small steps that accompany these actions are not interpreted as locomotion, otherwise unintentional drift will occur. In this subsection, we discuss what factors influence virtual-locomotion-drift and how we measured the effectiveness of drift suppression in LLCM-WIP.

False steps are suppressed by subtracting a constant value, called the *damping constant*, from the smoothed heel speed. The effectiveness of this mechanism depends on the size of the damping constant and the cutoff frequency of the smoothing filter. The cutoff frequency is chosen to produce a smooth and realistic visual flow. The damping constant is chosen to be the smallest value that effectively suppresses the majority of false steps.

To test how well false steps are suppressed when turning, we had several users perform a routine of walking and in-place turning motions in a virtual environment. This study employed the same participants and VE set-up as the latency analysis above (section 4.1).

We measured the average amount of drift due to false steps while turning the body a total of 720 degrees. Each of 5 users was asked to walk several steps, stop, turn 180 degrees to the left, turn 360 degrees to the right, turn 180 degrees back to the left, and then walk once again. This sequence of actions represents a relatively large amount of real-world maneuvering in place – effectively more than one would typically encounter during a single turn within an actual VE locomotion task. Thus the measures serve as an upper-bound on the amount of drift that could occur while turning during typical usage.

This 720-degree turn was repeated 4 times for each of the 5 users. For each trial, we computed the total amount of drift that occurred while performing the maneuver: in other words, the total arc-length of the locomotion path generated during the turn (which would ideally be zero, with no drift). The average drift over 20 trials was only 3 cm (with a standard deviation of 2 cm) per execution of the 720-degree turn.

5 Usability Evaluation

As part of an IRB-approved, between-subjects study comparing virtual locomotion techniques, we gathered data that allow us to make some qualitative comments about the usability of LLCM-WIP, as it compares to real-world walking and joystick, in terms of performance. The experimental task was to travel from one side to the other of a virtual bombed-out building. Participants had to hide behind pillars (safe-points) periodically to avoid gunfire and negotiate through a maze of low rubble as they moved from safe-point to safe-point. Figure 7 shows a view of the virtual scene. Eight participants, four females and four males, used the LLCM-WIP system. Each repeated the task 12 times, taking 75-90 seconds for each trial.

Our evaluation consists of an analysis of the time required to train to competence with LLCM-WIP, as compared to other locomotion methods, and comments from users that suggest areas for improvement.

We look at two (related) measures: the amount each user was in the open and not safely sheltered within a safe-zone during gunfire (body-seconds of exposure) and the length of the head's trajectory during gunfire (which would be near zero if the participant reached the safe-zone in time and didn't need to take any corrective steps to get into place). Figure 8 shows the results.

The study had five conditions – Real, Cowl, VRWalk, Joystick (JS), and Walking-In-Place (WIP). In the Real and Cowl conditions, the participant saw the real world; in the other three conditions the participant saw the virtual world using a head-worn display. In Cowl, the participant wore a hollowed-out head-worn display (open in the front) in order to reduce their field-of-view to be equivalent to that of an actual head-worn display. In Real, Cowl, and VRWalk, the participant physically walked to move through the environment, which included styrofoam mock-ups of the rubble and barriers. In the WIP and JS conditions, the participant remained essentially stationary, using LLCM-WIP or a joystick-only technique (respectively) to move in the virtual world. The joystick was a cordless, hand-held gamepad, operated with either thumb. The gamepad vibrated when the avatar collided with rubble or barriers. The joystick was used in both the WIP and JS conditions, however in the WIP condition it only controlled direction of locomotion, whereas, in JS, it controlled both direction and magnitude.

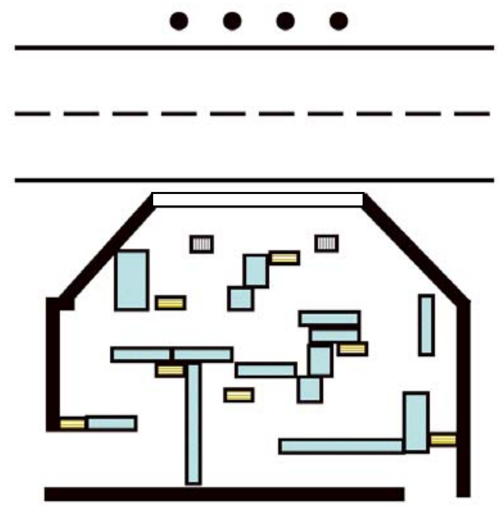

Figure 7: A top-down view of one of the maze environments the subjects navigated. The environment includes several safe-zones among low rubble (less than 1 m tall). The thick black lines show the extents of the virtual room, with a large open window spanning the north wall. The four black circles indicate enemy shooter positions. Red Cross trucks pass between the shooters and the room along the intervening road.

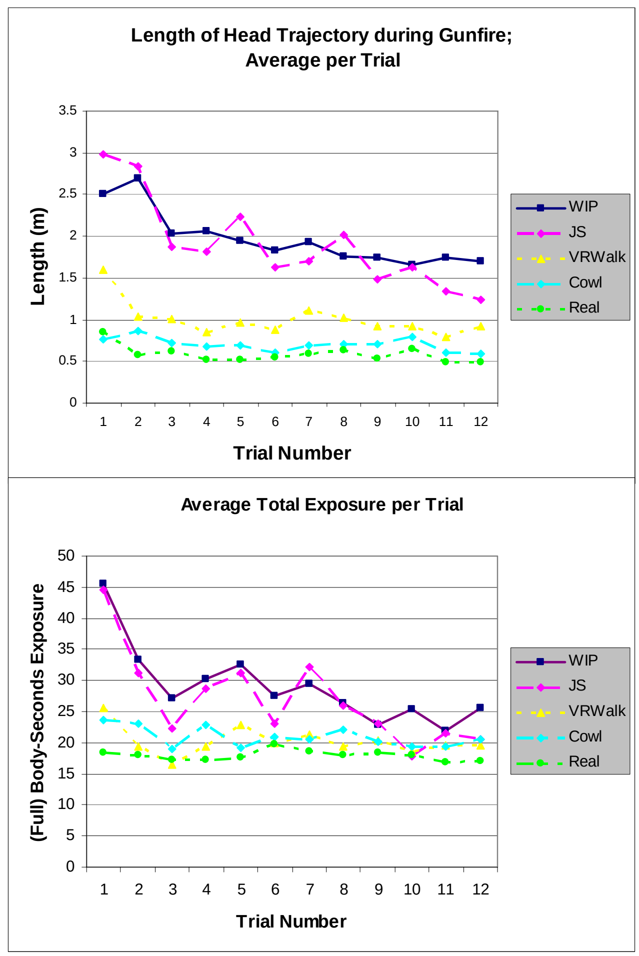

Figure 8: Measures of performance over time during training.

5.1 Training to Competence with the Interface

Before participants in the WIP and JS conditions began their 12 trials, they spent a few minutes using the system in a simple virtual environment. The duration of the practice session (determined by the participant) was typically about 3 minutes. This gave the participants enough training to familiarize themselves with the controls, but not enough time to train to competence, before the trials began.

Figure 8 shows that WIP users improved their performance (lowered their exposure and motion-under-fire scores) over the course of the 12 trials. The measures show a downward trend over time (i.e. improvement) and do not appear to have reached an asymptote by the end of the 12 trials. However, the measures do reach roughly the same level of consistency as Real, Cowl, and VRWalk after the first 3 to 6 trials – about 10 minutes of practice in the task environment – albeit at a higher level. The measures for the three conditions in which participants traveled by real-walking (Real, Cowl, and VRWalk) do not exhibit the same sort of drop over the first 6 trials, so we can arguably attribute the drop to learning the locomotion interface, not to learning the study task (navigation of the maze). User comments support this interpretation:

> *After I got the hang of it, it was much easier to move around in [trials 7-12].*
> *Participant 34, Question 8*

> *The coordination of thumb on analog stick and foot movements did not become natural until 4-5 [trials] into the VR portion.*
> *Participant 14, Question 8*

> *[During the twelve trials] I found it somewhat difficult to move with the walking in place setup. As I learned, I got better.*
> *Participant 10, Question 2*

> *The virtual environment got easier each time.*
> *Participant 38, Question 2*

There are two lessons here: (1) users should train until they are comfortable in an environment with complexity comparable to the task environment, and (2) training to competence on the LLCM-WIP system may take 15 minutes or more (about 6 trials).

5.2 User-reported Difficulties with the LLCM-WIP

Even after the 12 trials of practice, WIP users still reported some difficulty moving and maneuvering. These comments point us to elements of the interface that still need improvement.

> *I had difficulty making turns and avoiding obstacles in the virtual environment. …*
> *Participant 38, Question 8*

> *I often undershot the point where I wanted to stop, because of the mapping of stride size while walking in place to size of movement in VR was not intuitive.*
> *Participant 14, Question 10*

> *The most difficult part was deciding which motions to relegate to the hand set and which to actually move my body. For example, if I needed to adjust slightly to better hide behind the columns, should I use the game controller or creep over physically?*
> *Participant 21, Question 11*

6 Conclusions

The usability study (section 5) showed that WIP users did about as well as joystick but were still a good deal worse than all real-walking cases. No artificial locomotion system can hope to match real-world walking with regard to control and naturalness. However, stationary locomotion interfaces are much more cost-effective and convenient, making the tradeoff worthwhile. Similarly, a WIP system cannot be expected to out-perform joystick with regard to accuracy and responsiveness, especially since the joystick is the most familiar interface in gaming and joystick-based motion is more responsive than realistic locomotion motion to begin with. However WIP interfaces have the advantage of providing a physical component and stimulation of the proprioceptive system that resembles the feeling of real walking. WIP interfaces, because the virtual locomotion is combined with real-world head position, also produce a bobbing and oscillation of the viewpoint that a joystick-only interface does not. This makes the choice of WIP over joystick worthwhile despite the necessary performance loss. For LLCM-WIP, we have found that this performance loss is relatively small, particularly after a few trials of practice with the interface. If the best a WIP system can do is to rival joystick control in terms of performance, then LLCM-WIP is a success in this regard.

Previous work on WIP tends to focus on step detection. It provides scant information – and even scanter analysis – on the way in which virtual steps are actuated in the VE. The effectiveness of a WIP interface depends just as much upon the motion-synthesis method as it does upon the step detection. We hope that by providing detailed explanation of LLCM-WIP's motion-synthesis process we might encourage others to do the same in future work.

We also develop the idea of a continuous mapping from foot-motion to locomotion, casting artificial locomotion as a signal-processing problem rather than one of step-detection. LLCM-WIP is a continuous-motion system, in other words locomotion is not based on detecting a series of discrete footstep events that are mapped to virtual steps of a proscribed size. Rather, the output is instantaneous speed and direction of locomotion, a 2-d continuously-varying signal that is integrated by the VE application to yield the avatar's position offset. Thus the user of LLCM-WIP has direct control of locomotion velocity as a function of time. The idea is attractive because of the low latency and within-step control that it provides.

Future Work: Momentum. In real walking, a person does not go from stationary to their full cruising speed within a single step; it takes several steps to ramp up to the desired speed. The motion produced by the LLCM-WIP system (and most other existing systems) is not entirely realistic in this respect. Currently the user reaches full speed within the course of a single step. There is no concept of inertia. A possible solution would be to change the scale factor (from heel speed to locomotion speed) so that it is smaller while moving slowly and increases as the user speeds up.

Future Work: Making the System Less Expensive. One goal for future work is to make the system less expensive by replacing the magnetic tracker system with other sensors. The Ascension trackers we are using provide a 6-DOF reading: position and orientation. The tracker on the chest could be replaced with one that senses rotation only, since position is not used. The trackers on the shins could be replaced with accelerometers placed directly on the foot (since there is no risk of magnetic distortion from the floor). However, there is the issue of separating the vertical component of the step from the side-to-side movements that happen while maneuvering and turning-in-place.

Future Work: Alternative Means of Specifying Direction. One of the main benefits to WIP is that it uses the legs to control locomotion (as they are meant to do) and frees up the hands for other uses. The ideal WIP system is one in which all aspects of speed and direction can be controlled with the legs only. Chest-directed locomotion alone (without joystick modulation) is sufficient for many exploration tasks, but in tactical situations requiring motion in fully arbitrary directions we must supplement this with a joystick. A future goal for this system is to experiment with means of specifying arbitrary direction-control using the feet alone, possibly in a manner similar to Gaiter [5].

REFERENCES

[1] Mel Slater, Martin Usoh, and Anthony Steed. Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality. *ACM Transactions on Computer-Human Interaction*, 2:201-2019, September 1995.

[2] Martin Usoh, Kevin Arthur, Mary C. Whitton, Rui Bastos, Anthony Steed, Mel Slater, and Frederick P. Brooks, Jr. Walking > Walking-In-Place > Flying, in Virtual Environments. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, 1999), pages 359-364.

[3] Mary C. Whitton, Joseph V. Cohn, Jeff Feasel, Paul Zimmons, Sharif Razzaque, Sarah J. Poulton, Brandi McLeod, and Frederick P. Brooks, Jr. Comparing VE Locomotion Interfaces. In *Proceedings of the IEEE Virtual Reality, 2005* (2005), pages 123-130.

[4] Stephen DiVerdi, and Tobias Hollerer. GroundCam: A Tracking Modality for Mobile Mixed Reality. In *Proceedings of the IEEE Virtual Reality 2007* (2007), pages 75-82.

[5] James N. Templeman, Patricia S. Denbrook, and Linda E. Silbert. Virtual Locomotion: Walking in Place through Virtual Environments. *Presence*, 8:598-617, December 1999.

[6] Sharif Razzaque, David Swapp, Mel Slater, Mary C. Whitton, and Anthony Steed. Redirected Walking in Place. In *Proceedings of the 8th Eurographics Workshop on Virtual Environments* (2002), pages 123-130.

[7] L. Yan, R. S. Allison, and S. K. Rushton. New Simple Virtual Walking Method – Walking on the Spot. In *8th Annual Immersive Projection Technology (IPT) Symposium Electronic Proceedings* (2004).

[8] Mary C. Whitton and Sharif Razzaque. Locomotion, in *HCI Beyond GUIs*, Philip T. Kortum (Ed.), Morgan Kaufman (in press for Spring 2008).

[9] Linda E. Sibert, James N. Templeman, Robert C. Page, Jeremy T. Barron, Justin A. McCune, and Patricia S. Denbrook. Initial Assessment of Human Performance Using Gaiter Interaction Technique to Control Locomotion in Fully Immersive Virtual Environments (2004). Naval Research Laboratory, Technical Report NRL/FR/5510—04-10,086.

[10] James N. Templeman, Linda E. Sibert, and Robert C. Page. Immersive Simulation to Train Urban Infantry Combat. In *Virtual Media for Military Applications* (2006).

[11] Mel Slater. Personal Communication, February 2007.

[12] Sharif Razzaque. Personal Communication, February 2007.

[13] Jessica Rose and James G. Gamble (Eds). *Human Walking.* Lippincott Williams & Wilkins (2005).

[14] James N. Templeman and Linda E. Sibert Personal Communication, September 2007.