Development of Walking in Place System based on Zero Crossing Algorithm

Hongbiao Yang[1], Rapinder Sawhney[1], Shuguang Ji[1], Eric R. Wade[2]

[1]Department of Industrial and Systems Engineering, University of Tennessee, USA
[2]Department of Mechanical, Aerospace and Biomedical Engineering, University of Tennessee, USA

ABSTRACT — Walking in Place (WIP) is a way to facilitate locomotion tasks in the virtual environment while keeping the subject relatively static in the physical environment. This technique enables subjects to walk in a virtual space with limited physical space requirements. This paper introduces a burden-free and error-tolerant system to track the subjects' walking and turning motions, and to translate these motions to the virtual environment using the Microsoft Kinect. In addition, we introduce a zero crossing based algorithm that analyzes joint position data, detects the knee coordinates exchange pattern, and produces locomotion with low latency and jerkiness.

Index Terms — Walk in Place; Virtual Reality; Zero Crossing; Microsoft Kinect

1. Introduction

Virtual reality (VR) is defined as an immersive and interactive real time three-dimensional (3D) computer experience, which can respond the user's movements through visual graphics and provide a sense of being immersed in the virtual environment [1]. From the physical and psychological point of view, two factors critical to the VR experience are immersion and presence [2]. These vital components provide the user with the illusion of being in a real environment. Virtual reality has been used in a wide range of applications including, but not limited to, stroke rehabilitation [3], tourism [4], military training [5], [6] and entertainment [7], [8].

Locomotion in VR applications remains a challenge. The difficulty of using VR for walking interfaces is the adaptation of the proprioceptive aspects of walking to the virtual environment [9]. In order to reduce flaws of existing real-scale walking interfaces, walking-in-place (WIP) systems were proposed. The goal of the WIP system is to allow the user to move in a virtual environment in ways similar to walking in the physical environment [10]. However, WIP interfaces often suffer from latency [11], jerkiness [9], and user burden [12]. We propose a zero crossing based (ZCB) solution coupled with a speed-dampening function to address these issues. The zero crossing based WIP (ZCB-WIP) system is a WIP system that applies the zero crossing algorithm to detect stepping events and drives locomotion with low-latency and quasi-null jerkiness. This paper proceeds as follows. In Section 2, the previous studies about the development of WIP systems are reviewed. Section 3 presents the methods employed in this study. The performance and evaluation of our WIP system and algorithms are discussed in Section 4. Section 5 is the conclusion.

2. Related Work

A number of researchers have taken an interest in the design of advanced WIP systems. For example, Low-

Latency, Continuous-Motion (LLCM) WIP [13] is a high performance WIP system. The developers used sensors to collect chest orientation and heel speed data; they then converted these data into direction and motion in the virtual world. More WIP system implementations are summarized in Table 1. In general, these systems all suffer from two common problems. The first is starting/stopping latency, which is a key problem for accurate simulation of realistic forward motion [11]. Too much latency causes cyber sickness [14].

Latency also results in unrealistic virtual collisions [15] during walking, detracting from the immersive nature of the virtual interaction.

The second major problem is the jerkiness between adjacent steps. Jerkiness is a term in motion pictures that refers to a series of distinct snapshots instead of smooth and continuous motion, and is usually caused by dropped frames [16]. Jerkiness can result in non-fluent and non-smooth presentation of video [17] that annoys the video viewers [18] and detracts from the experience. In WIP systems, jerkiness can reduce the feel of realism and immersion in the virtual environment [19]. In addition to these two problems, device calibration and user burden are also considered important factors that impact the WIP system (see Table 1 for more references). The focus of this research project is on latency and jerkiness; however, we will also provide information regarding our attempts to address calibration and user burden issues.

Table 1: Summary of WIP interfaces

Author	Year	System Name	Equipment	Calibration[1]	Burden Free
J. Feasel [13]	2008	LLCM -WIP	Tracker on chest and feet	Yes	No
J. D. Wendt[20]	2010	Gud WIP	Trackers on shins, shin tracking cameras, and HMD	Yes	Yes
D. Swapp [21]	2010	Wizdish	Low fiction and dish-shaped Wizdish	No	Yes
K. K. Patel [22]	2010	Unconstrained Walking Plane	Motor-less treadmill and mechanical rotating base	No	Yes
B. Williams [23]	2011	WIP-WII	Nintendo Wii fit balance board	N/A	Yes
D. J. Zielinski [24]	2011	Shadow Walking	Floor screen, under floor projectors, and camera	No	Yes
J. Kim [25]	2012	Sensor-fusion WIP	Magnet sensor and smartphone	Yes	No
L. Bruno [26]	2013	SAS-WIP	Large screen and optical tracker	N/A	No
Nilsson [27]	2013	Tapping in Place	HMD, marker, and camera tracker	Yes	No
Yang	2015	ZCB-WIP	Kinect camera and TV screen	No	Yes

Note: 1. N/A means calibration is not mentioned in the research.

3. Methods

The ZCB-WIP system developed in this study utilizes unique hardware and software components. The ZCB-WIP system hardware includes a Microsoft Kinect sensor, a commercial TV screen, and a PC. The Kinect is a line of motion sensing input devices developed by Microsoft for Xbox and Windows PCs (as shown in Figure 1). The ZCB-WIP system software creates a 3D virtual scenario based on a real suburban community, and was developed using the Unity3D® game engine.

3.1. Hardware

In this research, the Kinect is used to track the skeletal joints of a human standing in front of the sensor. There are 20 key joints that can be detected and tracked by the Kinect (as shown in Figure 2). Tracking these joints renders possible the detection of various human body movements (such as walking behaviors). With a capture rate of 30 frames per second, the trajectory of each joint is smoothly tracked in real time. In the Kinect system, tracking is performed by coupling RGB and depth sensors [28]. Because we have adopted non-immersive VR technology (i.e., a screen instead of a head mounted display (HMD)), we use a commercial level TV screen to provide the virtual display. As mentioned above, we are using skeletal joint data collected by the Kinect sensor instead of the raw image stream, which significantly reduces the computational load. The relatively inexpensive combination of commercial devices is sufficiently powerful to handle computational complexity while producing smooth visual feedback.

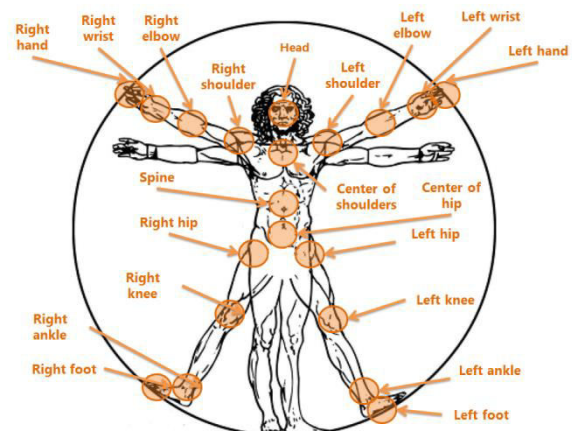Figure 1: System hardware (Microsoft Kinect, TV, and PC)

Figure 2: Vitruvian man and the 20 joints tracked by the Kinect

Because the subject must remain in the sensor's field of view (FOV), the WIP system requires a human to stay in a bounded physical space. To satisfy this requirement, feedback (the display sensor's FOV at the corner of the TV screen as shown in Figure 3) and feedforward (the placement of a cross mark sign on floor) were used to prevent users from leaving the sensor's FOV.

Figure 3: (1) Skeleton and (2) Kinect camera FOV
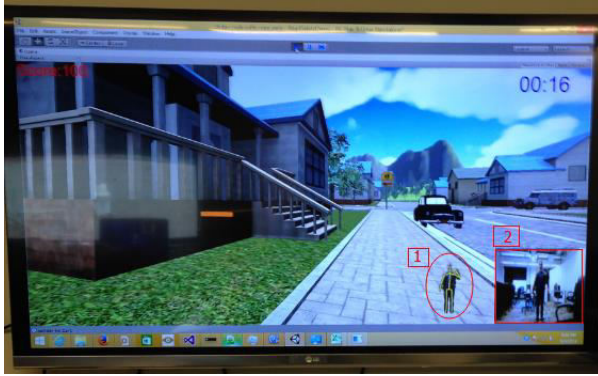
3.2. Software

Once tracking data are acquired through the hardware system, data are processed by the software system to generate smooth locomotion. Three components of the software system will be discussed in this section: the ZCB-WIP implementation, the speed-dampening algorithm, and rotation detection.

3.2.1. Zero crossing based algorithm

The joint trajectories tracked by the Kinect sensor are susceptible to variations caused by system and random errors. In this study, we apply the zero crossing algorithm to reduce this variation and accurately detect WIP steps. The zero crossing algorithm is commonly used in electronics, mathematics, sound and image processing. This algorithm is also used in pedestrian dead reckoning [29], [30], step length estimation [31] and step detection [32] in pedestrian tracking technologies. The zero crossing algorithm describes a point where the sign of a mathematical function changes. It is based on the zero crossing rate [33] (ZCR), at which the signal changes from positive to negative or vice versa. ZCR is defined as:

$$\text{ZCR} = \frac{1}{T-1} \sum_{\{t=1\}}^{\{T-1\}} \mathbb{I}\{s_t s_{\{t-1\}} < 0\}$$

$$s_t = KneeDiff = LeftKnee_t.Y - RightKnee_t.Y$$

Where $\mathbb{I}\{A\}$ is an indicator function. If the argument A is true, $\mathbb{I}\{A\}$ returns 1; otherwise, it returns 0. In this study, if $s_t s_{t-1} < 0$, then $\mathbb{I}\{s_t s_{t-1}\} < 0 = 1$, otherwise, $\mathbb{I}\{s_t s_{t-1} > 0\} = 0$. s_t is the knee difference at time t and s_{t-1} is the knee difference at time $(t-1)$ ($s_t = KneeDiff$ and $s_{t-1} = preKneeDiff$). When a human is walking, he/she will move by lifting and setting down each leg alternately. This locomotion will cause s_t to change sign for each step. The pseudo code snippet for this relationship is:

$$KneeDiff = LeftKnee.Y - RightKnee.Y$$
$$If\ (absolute\ value\ of\ KneeDiff$$
$$< threshold)$$
$$then: KneeDiff = 0$$
$$If\ (KneeDiff * preKneeDiff < 0)$$
$$then: KneeSwap\ increases\ by\ an\ increment$$

$$RawSpeed\ is\ proportional\ to\ KneeSwap$$
$$update\ preKneeDiff < - KneeDiff$$

This code snippet is placed in the $Update()$ function, which is then executed in each frame. The logic process followed by the ZCB algorithm in the $Update()$ function is demonstrated in Figure 4. First, the variable $KneeDiff$ is the difference between $RightKnee_t.Y$ and $LeftKnee_t.Y$. If the absolute value of $KneeDiff$ is larger than the predefined threshold value, it means the leg motion is obvious enough to be detected as a step event; otherwise, it will be treated as noise and filtered out. When the value of $KneeDiff * preKneeDiff$ is negative, crossing occurs and an increment Δ is added to the variable $KneeSwap$; accordingly, the variable $RawSpeed$ will increase proportionally because of $RawSpeed = KneeSwap * k$, where k is a scalar coefficient for speed tuning. The constant variable Δ

is a parameter used to adjust locomotion speed in the virtual world. In this study the values of constant variables are set as following: $threshold = 0.02$ and $\Delta = 1.2$ (both are determined empirically).
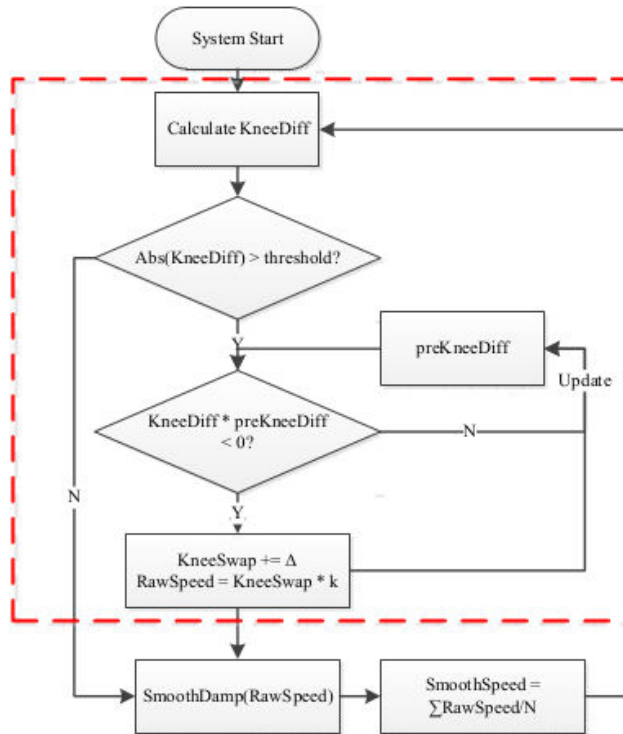

Figure 4: The logic loop of the ZCB algorithm

3.2.2. Speed-dampening algorithm

Whenever a step is detected by the Kinect sensor, a change in speed will be generated in the virtual world. In practice, there are two commonly used methods for determining forward speed [13], [34]. One method is to use body position as an input, and to produce keystroke and mouse events as outputs [34]. For example, when we press and hold the "forward" arrow key on the keyboard, the subject in the virtual world will keep moving forward until we release the "forward" arrow key. The advantage of this method is that it is simple and straightforward, and does not require changing the system configuration. With this approach, the stepping event is treated as a hardware interrupt event. The disadvantage of this method is that the frequency of the step event (about 2Hz) is much slower than the frequency of hardware interrupt

events (about 100 Hz). As a result, there are few speed impulses in each second, which will certainly result in severe jerkiness during walking. An alternative method is to use the box and the saw-tooth functions as applied in the LLCM-WIP system [13]. Using this method, the jerkiness between the two consecutive impulses can be smoothed. In this study, we use a revised saw-tooth function for speed smoothing. In each frame, the function $SmoothDamp()$ is called to dampen the speed from current value to 0 within a short period of time (e.g., 0.5 seconds). If the user stops generating new speed increments, the advancement of the viewpoint in the virtual world will stop after 0.5 seconds. If the user is continuously walking, the acceleration from the ZCB algorithm will counteract the deceleration from the speed-dampening algorithm, such that the speed of the subject in the virtual world is relatively stable and continuous. To summarize our speed-dampening algorithm, $KneeSwap$ increases while a step is detected and reaches 0 in 0.5 seconds if there is no step detected. The 0.5 seconds is also selected empirically.

$$KneeSwap = SmoothDamp(KneeSwap, 0, 0.5)$$
$$if\ (KneeSwap\ <\ 0.05)$$
$$\qquad then:KneeSwap\ =\ 0$$

Additionally, to avoid abrupt speed changes, a 4-period moving average speed is used to smooth the most recent speed values and reduce unwanted randomness and period-to-period speed variations. The speed changes before and after smoothing are described in Figure 5. We conclude that whenever the knee difference results in a zig-zag pattern (green dotted curve) indicating that the subject is walking, the raw speed will gain an increment (blue dashed curve). It is also worth mentioning that, because of the nature of the ZCB algorithm, the magnitude of the knee difference has no direct impact on the walking speed. By applying the 4-period moving

average, the variation of the smoothing speed becomes small (red solid curve). This smoothing speed will finally drive the advancement of the viewpoint and enables the subject to move in the virtual world.
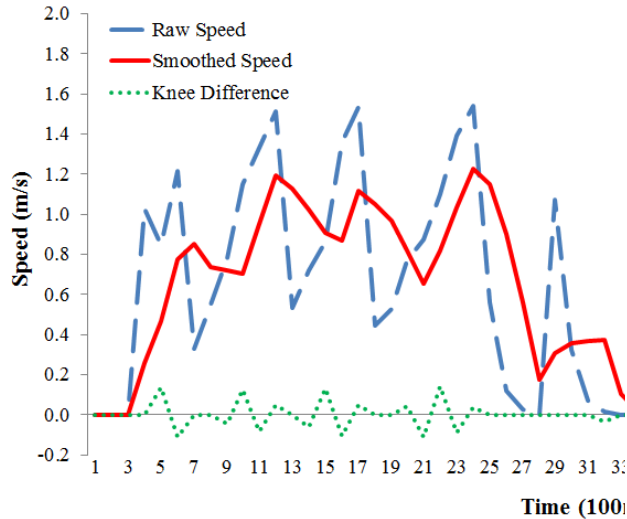

Figure 5: Knee difference and locomotion speed plot

3.2.3. Rotation detection

The joint position data of the left shoulder and right shoulder collected by the Kinect are used to track the subject's rotation. As discussed above, the Kinect sensor can also capture the depth value of each pixel as well as each body joint. As can be seen in Figure 6, when a human turns left or right, the depth values of the left and right shoulder joints will increase and decrease, respectively. When turning left, the difference between the depth values of the left and right shoulder joints ($LeftShoulder.Z - RightShoulder.Z$) will change from 0 to a positive value. Similarly, if the subject turns to the right, this depth difference will change from 0 to a negative value. In order to tell the real turning behavior, we set another threshold value. If the absolute value of the depth difference is larger than the threshold value, it means the turning behavior is sufficiently obvious to be quantified by our system; otherwise, we consider this depth change as noise. In addition, the turning speed of the avatar is directly proportional to the depth between the left and right shoulders.
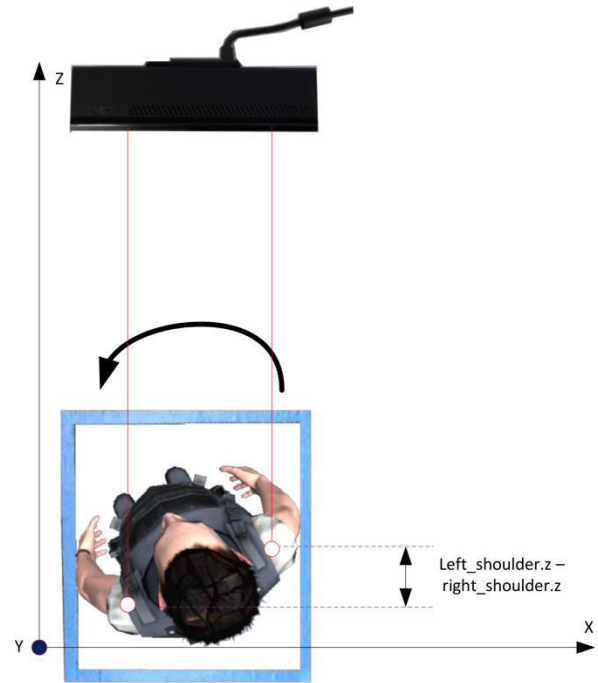

Figure 6: Bird eye view of making a left turn

3.3 System evaluation

To evaluate the performance and features of our ZCB-WIP system, an objective experiment and a subjective survey are conducted.

3.3.1 Experiment One: Objective Evaluation

We use a simple evaluation program based on our ZCB-WIP system to evaluate the actual latency from the participants' performances. We ask study participants to follow the instructions on the screen, such as "GO" and "STOP" with a downward counting timer (see Figure 7). During the experiment session, three variables are recorded: (1) the value of knee difference captured by the system; (2) the immediate locomotion speed before smoothing; and (3) the locomotion speed after smoothing. The sampling rate for these variables is 10Hz; thus, each data point represents 100ms. We choose a moderate sampling rate instead of a higher value mainly for performance considerations. Variables are stored in a

local file for post-processing and statistical analysis; thus, increasing the sampling rate results in I/O operations that may bring extra load on the computer and adversely impact the framerate of the visual feedback. Also, according to the result of our analysis, 100ms is an acceptable level of granularity for our study. The latencies can be calculated through counting the number of data points. As shown in Figure 8, we focus on two types of latencies: the starting latency is the number of non-zero knee swap values before the smoothed locomotion speed becomes non-zero during the "GO" period; the stopping latency is the number of zero knee swap values before the smoothed locomotion speed becomes zero during the "STOP" period. Even though there is an instruction variable indicating the current instruction ("GO" or "STOP"), this variable is ignored when calculating the latencies because participants do not always cope with the pace indicated on screen. Instead, we use the value of $Kneediff$ to serve this purpose (when $Kneediff$ becomes zero, it means the participant stopped walking).

Knee Difference	Locomoting speed before smoothing	Locomoting speed after smoothing
0.0000	0.0000	0.0000
-0.0849	0.0000	0.0000
-0.0566	0.0000	0.0000
0.0000	0.0000	0.0000
0.1050	1.0586	0.2836
0.1299	0.9282	0.5482
...	...	...
0.0704	0.9146	0.8037
0.0714	0.6822	0.8904
0.0000	0.4914	0.9888
0.0000	0.3512	0.7870
0.0000	0.2534	0.6099
0.0000	0.1143	0.4446
0.0000	0.0345	0.3026
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000

Starting Latency = 0.3S

Stopping Latency = 0.5S

Figure 8: Example of the representation of starting and stopping latencies in the data

Seventeen participants (11 male; 6 female) were recruited to participate in the evaluation experiment. These participants were selected from a convenience sample of students and staff at the University of Tennessee. Their ages range from 21 to 38, and their height varies from 65 to 74 inches. The majority of participants had little or no prior VR gaming experience before using our system, and some had only limited knowledge of Microsoft Kinect gaming. Prior to using our system, participants were not informed of the purpose of the experiment. Each participant was required to finish 7 starting instructions and 7 stopping instructions. The participants were instructed to walk intermittently to facilitate collection of latency data.

3.3.2. Experiment Two: Subjective Comparison Survey

In addition to the aforementioned objective experiments, we recruited a second group of participants for a subjective system evaluation. This group included 35 participants (29 male; 6 female), aged 13 to 17 years. The group of participants was asked to experience two VR systems: one is our ZCB-WIP system and the other is a demo program using the Oculus Rift HMD and traditional keyboard/mouse control. Eight subjective survey

Figure 7: Instruction text for users to start or stop walking in place

questions were answered by the participants after they tried both VR systems to rate their subjective experiences while using each VR system. These eight survey questions were selected from well-known VR evaluation questionnaires [35], with proper modification and rewording. The specific questions are listed in Table 2.

Table 2: Survey questions for subjective evaluation:

Q1: Walking is natural or not? Scale: 1 is most artificial and 5 is most natural.

Q2: System is responsive or not? Scale: 1 is not responsive and 5 is most responsive.

Q3: How much fatigue do you feel during the experiment session? Scale: 1 is least fatigue and 5 is most fatigue.

Q4: How much motion sickness do you feel during the experiment session? Scale: 1 is least motion sickness and 5 is most motion sickness.

Q5: How much latency (lag) do you feel during the experiment session? Scale: 1 is least latency and 5 is most latency.

Q6: How much immersion (being there) do you feel? Scale: 1 is least immersive and 5 is most immersive.

Q7: How much easiness is the virtual system to you? Scale: 1 is very easy and 5 is most complicated.

Q8: How much comfort do you feel when experiencing the system? Scale: 1 is not comfortable and 5 is most comfortable.

4. Results and Discussion

To evaluate the performance of the objective experiment, we calculate the mean and standard deviation of the starting latency and the stopping latency. We also listed the adjusted latency after we found that there is extra room for improvement on the stop latency. Furthermore, to test the calibration free feature, we conducted Mann-Whitney U tests on those latency data between two subgroups of participants with different heights. We are using the Mann-Whitney U test instead of the T test to validate the hypotheses because the sample size does not meet the required level for parametric analysis. Finally, we use Mann-Whitney U test on the Likert subjective response data to evaluate the fatigue and the motion sickness, as well as other subjective feelings between the ZCB-WIP system and the HMD system.

4.1 Objective Experiments Analysis

The average starting latency is 287ms (standard deviation: 121ms); and the average stopping latency is 781ms (standard deviation: 44ms). The longer stopping latency is due to the speed smoothing method; as mentioned above, in order to reduce the jerkiness caused from sudden changes in walking speed, the speed is smoothed by averaging the value in four frames. To overcome this issue, we modified the speed smoothing method, setting the smoothed speed to zero if the speed before smoothing is zero. After this improvement, the stopping latency is reduced from 781ms to 474ms (standard deviation: 35ms). Thus, the mean starting latency (287ms) and mean stopping latency (474ms) in our research are under acceptable levels, as compared to the value reported in previous studies (500ms) [15], [27]. Theoretically, the starting latency in our system begins when the foot of the subject is first raised above the ground, elevating to the maximum height, and ending when this foot touches the ground again. During this period, there will be no change of sign for the variable $Kneediff$ (will remain positive or negative depending on which foot raised first). But when the other foot begins rising, the sign of $Kneediff$ will change, and consequently result in advancement in the virtual world. The period between steps usually takes a relatively short time. Thus, the starting latency is negligible. In addition, the $SmoothDamp()$ function enables the speed of locomotion to decrease to 0 gradually while no step is detected. Because we are able to tell whether the

participant is walking or static within a few frames, we can dynamically change the deceleration rate of the algorithm.

We also found that the variability of the starting latency is relatively high.

This is due to the lack of calibration of the system. Beside the latency, there was no apparent jerkiness reported by participants during the experiment. Advantages of the ZCB algorithm include the lack of a calibration requirement and the ability to work with various body sizes. To evaluate these characteristics in our system, we conducted a Mann-Whitney U test on the starting latency and the stopping latency for two subgroups' data (with significance α set at 0.05). Participants are divided into two groups by the median of the population heights. One group includes taller participants (8 participants, higher than 68

inches). The other group includes shorter participants (9 participants, shorter than 68 inches). The null hypothesis is that there is no difference on the starting latency or stopping latency between two groups. Because the p-values of the test on starting latency and stopping latency are 0.1453 and 0.1181 respectively, we cannot reject the null hypothesis. Therefore, we claim that the height of the participant does not affect the starting or stopping latency in our WIP system.

4.2 Subjective Survey Analysis

After collecting the response data from the survey questions, we conduct Mann-Whitney U tests on each question item. First, we compare the basic statistics. Table 3 shows a list of non-parametric statistics for two systems on each question.

Table 3: Basic quartile statistics of comparison between ZCB-WIP and HMD systems

	ZCB-WIP	HMD
Q1: natural	[2.00, 2.00, 3.00]	[2.25, 4.00, 5.00]
Q2: responsive	[2.00, 2.00, 3.00]	[4.25, 5.00, 5.00]
Q3: fatigue	[1.00, 1.00, 2.00]	[1.00, 2.00, 3.00]
Q4: sickness	[1.00, 1.00, 1.00]	[1.00, 2.00, 4.00]
Q5: latency	[2.00, 3.00, 4.00]	[1.00, 1.00, 1.75]
Q6: being there	[1.00, 2.00, 3.00]	[4.00, 5.00, 5.00]
Q7: easiness	[2.00, 3.00, 3.75]	[1.00, 2.00, 4.00]
Q8: comfortable	[2.00, 3.00, 3.00]	[3.00, 4.00, 5.00]

Note: numbers in square bracket are: [25% quantile, median, 75% quantile]

From Table 3, we can see that based on the participants' rating, the HMD system is favored on most of the question items, like naturalness, responsiveness, immersion and so on. This is as expected, since the HMD can output stereoscopic image that provides more visual immersion to the users than other less immersive system. Besides, the keyboard controlling interface has undoubtedly lower latency. From Questions 3 and 4, we can find that in our ZCB-WIP system resulted in less fatigue and motion sickness when compared to the HMD system. To determine whether those differences are

significant, we conducted Mann-Whitney U test on those two items. The hypotheses include:

H_{a0} : The ZCB-WIP system has equal or higher fatigue than the HMD system

H_{a1}: The ZCB-WIP system has lower fatigue than the HMD system

H_{b0} : The ZCB-WIP system has equal or higher motion sickness than the HMD system

H_{b1} : The ZCB-WIP system has lower motion sickness than the HMD system

We performed a one-tailed Mann-Whitney U test on the two pairs of hypotheses and found that $p = 0.0432$ for H_a . Thus, we can reject the null hypothesis and conclude that the ZCB-WIP system results in lower fatigue than the HMD system. Similarly, $p < 0.01$ for H_b. We can reject the null hypothesis and conclude that the ZCB-WIP system results in lower motion sickness than the HMD system. The conclusion that the VR with HMD will cause more motion sickness and fatigue is consistent with prior studies [37]-[39].

5. Conclusion

This paper develops a WIP system that uses the zero crossing algorithm and Microsoft Kinect sensor. By using our ZCB-WIP system, latency is reduced while jerkiness is kept to a quasi-null state. According to the 500ms latency threshold reported in previous studies [15], [27], there is a noticeable reduction of latency in our ZCB-WIP (287ms for starting latency and 475ms for stopping latency). Also, the participants did not report any jerkiness while using our system.

One of the advantages of our ZCB-WIP system is that there is no calibration required for system users. When compared to distance based motion tracking algorithms, the ZCB algorithm is simpler and more reliable in detecting and tracking the user walking behaviors, regardless of an individual height. Furthermore, Kinect devices have embedded calibration algorithms in internal memory and the algorithms developed using the Microsoft Software Development Kit do not require a calibration to activate skeletal tracking [39], [40], even though there is certain level of distortion of the depth data [41]. Therefore, calibration has little impact on the experimental system accuracy. Thus, system calibration and training are not required for our system.

The objective experiment involving 17 participants was conducted to test the performance of the system.

The results of the experiment validate the advantages of the system. Moreover, according to the survey data from the group of participants who compared the ZCB-WIP and HMD systems, we found out that the ZCB-WIP system developed in this research is free of burden and causes limited motion sickness, despite that there is still distance from the fully immersive 3D VR in some extent.

Our system still has some limitations. One limitation is the technique used to track rotations. Microsoft Kinect requires the users to face the sensor and TV screen at all times; thus, once the user makes a rotation, the user is no longer facing the sensor and TV screen anymore. To fix this, the user needs to restore his/her facing direction. This "restore after turn" action may bring confusion to the users and limit the ecological validity of the virtual task. The ZCB-WIP system developed in this study used the first generation Microsoft Kinect. Currently, the second generation Microsoft Kinect has been released in the market. The resolution of the second generation Microsoft Kinect is much higher than the first generation. It also purports to be able to track the rotation angle of the head joint as well as the facial features of the user. One future direction of our research is to incorporate the second generation Microsoft Kinect into the ZCB-WIP system.

Another limitation is the inability to walk backwards. Because it is difficult to differentiate forward and backward walking in a WIP setting without physical displacement, we use leaning back as a proxy for walking backward. If the horizontal displacement of the spine is bigger than that of the feet ($spineZ - \frac{FootLeft.Z + FootRight.Z}{2} > delta$, we apply a speed opposite to the current forward direction. Although this behavior is not as natural as walking forward, considering the limited need for walking backwards in our task, we decide to choose this implementation so that the full navigation ability is covered.

One more limitation is the display interface. In our research we are using a commercial TV screen in front of the user, which can provide only limited immersion. An HMD that outputs a stereoscopic image stream is a much better solution in creating a fully immersive experience. This conclusion is supported by the subjective survey study on a group of participants who tried both the ZCB-WIP system and the HMD system. However, currently available HMDs all have wires connecting the device to the server [42], [43] for image streaming and sensor data communication. This tethered connection may be problematic for various users, and may interfere with natural walking and turning, entangle the users, and become uncomfortable [44]. Further, from the results of the subjective survey, we also found that participants feel more motion sickness and fatigue when using the HMD system. Thus, depending on application and target population, the ZCB-WIP and HMD systems each have advantages and disadvantages that must be taken into consideration.

ACKNOWLEDGMENT

REFERENCE

[1]   D. Reid, "The influence of virtual reality on playfulness in children with cerebral palsy: a pilot study," *Occup. Ther. Int.*, vol. 11, no. 3, pp. 131–144, Aug. 2004.

[2]   M. Gutierrez, F. Vexo, and D. Thalmann, *Stepping into Virtual Reality*. Springer Science & Business Media, 2008.

[3]   E. Wade and C. J. Winstein, "Virtual reality and robotics for stroke rehabilitation: where do we go from here?," *Top. Stroke Rehabil.*, vol. 18, no. 6, pp. 685–700, 2011.

[4]   D. A. Guttentag, "Virtual reality: Applications and implications for tourism," *Tour. Manag.*, vol. 31, no. 5, pp. 637–651, 2010.

[5]   B. Schlake and M. Narayanan, "Virtual reality applications," in *WESCON/94. Idea/Microelectronics. Conference Record*, 1994, pp. 76–81.

[6]   S. Ji, Z. Zhang, H. Yang, D. Liu, and R. Sawhney, "Simulation of Coastline's Echo on Marine Radar Simulator Based on Fractal Theory," in *Transportation Research Board 94th Annual Meeting*, 2015.

[7]   M. Mine, "Towards Virtual Reality for the Masses: 10 Years of Research at Disney's VR Studio," in *Proceedings of the Workshop on Virtual Environments 2003*, New York, NY, USA, 2003, pp. 11–17.

[8]   T. Abacı, R. de Bondeli, J. Cíger, M. Clavien, F. Erol, M. Gutiérrez, S. Noverraz, O. Renault, F. Vexo, and D. Thalmann, "Magic wand and the Enigma of the Sphinx," *Comput. Graph.*, vol. 28, no. 4, pp. 477–484, Aug. 2004.

[9]   F. Multon and A.-H. Olivier, "Biomechanics of Walking in Real World: Naturalness we Wish to Reach in Virtual Reality," in *Human Walking in Virtual Environments*, F. Steinicke, Y. Visell, J. Campos, and A. Lécuyer, Eds. Springer New York, 2013, pp. 55–77.

[10]   J. Templeman, P. Denbrook, and L. Sibert, "Virtual locomotion: Walking in place through virtual environments," *Presence*, vol. 8, no. 6, pp. 598–617, 1999.

[11]   L. Yan, R. Allison, and S. Rushton, "New simple virtual walking method-walking on the spot," presented at the Proceedings of the IPT Symposium, 2004, pp. 1–7.

[12]   J. Psotka, "Immersive training systems: Virtual reality and education and training," *Instr. Sci.*, vol. 23, no. 5–6, pp. 405–431, 1995.

[13]   J. Feasel, M. C. Whitton, and J. D. Wendt, "LLCM-WIP: Low-latency, continuous-motion walking-in-place," presented at the 3D User Interfaces, 2008. 3DUI 2008. IEEE Symposium on, 2008, pp. 97–104.

[14]   L. E. Sibert, J. N. Templeman, R. C. Page, J. T. Barron, and J. A. McCune, "Initial Assessment of Human Performance Using the Gaiter Interaction Technique to Control Locomotion in Fully Immersive Virtual Environments," Jun. 2004.

[15]   M. Usoh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. 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, USA, 1999, pp. 359–364.

[16]   Q. Huynh-Thu and M. Ghanbari, "Impact of jitter and jerkiness on perceived video quality,"

presented at the Proceedings of the Workshop on Video Processing and Quality Metrics, 2006.

[17] S. Borer, "A model of jerkiness for temporal impairments in video transmission," in *2010 Second International Workshop on Quality of Multimedia Experience (QoMEX)*, 2010, pp. 218–223.

[18] W. Lin and C.-C. Jay Kuo, "Perceptual visual quality metrics: A survey," *J. Vis. Commun. Image Represent.*, vol. 22, no. 4, pp. 297–312, May 2011.

[19] F. Stakem, G. AlRegib, and B.-H. (Fred) Juang, "Towards Modeling Human Arm Movement in a CVE," in *Proceedings of the First International Conference on Immersive Telecommunications*, ICST, Brussels, Belgium, Belgium, 2007, pp. 8:1–8:5.

[20] J. D. Wendt, M. C. Whitton, and F. P. Brooks Jr, "Gud wip: Gait-understanding-driven walking-in-place," presented at the Virtual Reality Conference (VR), 2010 IEEE, 2010, pp. 51–58.

[21] D. Swapp, J. Williams, and A. Steed, "The implementation of a novel walking interface within an immersive display," presented at the 3D User Interfaces (3DUI), 2010 IEEE Symposium on, 2010, pp. 71–74.

[22] K. K. Patel and S. K. Vij, "Unconstrained Walking Plane to Virtual Environment for Spatial Learning by Visually Impaired," *Ubiquitous Comput. Commun. J.*, 2010.

[23] B. Williams, S. Bailey, G. Narasimham, M. Li, and B. Bodenheimer, "Evaluation of walking in place on a wii balance board to explore a virtual environment," *ACM Trans. Appl. Percept. TAP*, vol. 8, no. 3, p. 19, 2011.

[24] D. J. Zielinski, R. P. McMahan, and R. B. Brady, "Shadow walking: An unencumbered locomotion technique for systems with under-floor projection," presented at the Virtual Reality Conference (VR), 2011 IEEE, 2011, pp. 167–170.

[25] J. Kim, D. Gracanin, and F. Quek, "Sensor-fusion walking-in-place interaction technique using mobile devices," presented at the Virtual Reality Short Papers and Posters (VRW), 2012 IEEE, 2012, pp. 39–42.

[26] L. Bruno, J. Pereira, and J. Jorge, "A new approach to walking in place," in *Human-Computer Interaction–INTERACT 2013*, Springer, 2013, pp. 370–387.

[27] N. C. Nilsson, S. Serafin, M. H. Laursen, K. S. Pedersen, E. Sikstrom, and R. Nordahl, "Tapping-in-place: Increasing the naturalness of immersive walking-in-place locomotion through novel gestural input," presented at the 3D User Interfaces (3DUI), 2013 IEEE Symposium on, 2013, pp. 31–38.

[28] R. J. Schalkoff, *Digital image processing and computer vision*. John Wiley & Sons Australia, Limited, 1989.

[29] W. Chen, R. Chen, Y. Chen, H. Kuusniemi, and J. Wang, "An effective Pedestrian Dead Reckoning algorithm using a unified heading error model," in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, 2010, pp. 340–347.

[30] S. Beauregard, "A Helmet-Mounted Pedestrian Dead Reckoning System," in *2006 3rd International Forum on Applied Wearable Computing (IFAWC)*, 2006, pp. 1–11.

[31] S. H. Shin, C. G. Park, J. W. Kim, H. S. Hong, and J. M. Lee, "Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors," in *IEEE Sensors Applications Symposium, 2007. SAS '07*, 2007, pp. 1–5.

[32] M. Alzantot and M. Youssef, "UPTIME: Ubiquitous pedestrian tracking using mobile phones," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, 2012, pp. 3204–3209.

[33] C. H. Chen, *Signal Processing Handbook*. CRC Press, 1988.

[34] H. Istance, A. Hyrskykari, S. Vickers, and T. Chaves, "For Your Eyes Only: Controlling 3D Online Games by Eye-Gaze," in *Human-Computer Interaction – INTERACT 2009*, T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. O. Prates, and M. Winckler, Eds. Springer Berlin Heidelberg, 2009, pp. 314–327.

[35] B. G. Witmer and M. J. Singer, "Measuring Presence in Virtual Environments: A Presence Questionnaire," *Presence Teleoperators Virtual Environ.*, vol. 7, no. 3, pp. 225–240, Jun. 1998.

[36] P. A. Howarth and P. J. Costello, "The occurrence of virtual simulation sickness symptoms when an HMD was used as a personal viewing system," *Displays*, vol. 18, no. 2, pp. 107–116, Dec. 1997.

[37] J. Moss, J. Scisco, and E. Muth, "Simulator Sickness during Head Mounted Display (HMD) of Real World Video Captured Scenes," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 52, no. 19, pp. 1631–1634, Sep. 2008.

[38] J. Kuze and K. Ukai, "Subjective evaluation of visual fatigue caused by motion images,"

Displays, vol. 29, no. 2, pp. 159–166, Mar. 2008.

[39] C. C. Martin, D. C. Burkert, K. R. Choi, N. B. Wieczorek, P. M. McGregor, R. A. Herrmann, and P. A. Beling, "A real-time ergonomic monitoring system using the Microsoft Kinect," in *2012 IEEE Systems and Information Design Symposium (SIEDS)*, 2012, pp. 50–55.

[40] Z. Zhang, "Microsoft Kinect Sensor and Its Effect," *IEEE Multimed.*, vol. 19, no. 2, pp. 4–10, Feb. 2012.

[41] D. Herrera C., J. Kannala, and J. Heikkilä, "Joint Depth and Color Camera Calibration with Distortion Correction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 2058–2064, Oct. 2012.

[42] P. Willemsen, M. B. Colton, S. H. Creem-Regehr, and W. B. Thompson, "The Effects of Head-mounted Display Mechanics on Distance Judgments in Virtual Environments," in *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization*, New York, NY, USA, 2004, pp. 35–38.

[43] J. M. Plumert, J. K. Kearney, J. F. Cremer, and K. Recker, "Distance Perception in Real and Virtual Environments," *ACM Trans Appl Percept*, vol. 2, no. 3, pp. 216–233, Jul. 2005.

[44] B. S. Santos, P. Dias, A. Pimentel, J.-W. Baggerman, C. Ferreira, S. Silva, and J. Madeira, "Head-mounted display versus desktop for 3D navigation in virtual reality: a user study," *Multimed. Tools Appl.*, vol. 41, no. 1, pp. 161–181, Aug. 2008.

BIOGRAPHIES

Hongbiao Yang is a PhD student and works as a graduate research assistant at the University of Tennessee in the Department of Industrial and Systems engineering. His research interest focuses on applying Virtual Reality to train pedestrians in road safety skills, and analyzing the corresponding learning effectiveness. He works on various projects in the Natural Interaction Lab in the Department of Industrial and Systems Engineering.

Email: hyang22@ulk.edu

Rupy Sawhney was nominated for 2009 Chancellor's Outstanding Academic Outreach Award. He serves as the Associate Professor and Associate Head of the Department of Industrial and Systems Engineering at the University of Tennessee. His Outreach work combines assisting business and industry with educating students while at the same time pursuing scholarly publications and funded University research.

Email: sawhney@utk.edu

Shuguang Ji holds a PhD in Civil Engineering from the University of Tennessee at Knoxville. He is currently a research assistant professor in the Department of Industrial and Systems Engineering at the University of Tennessee, Knoxville. Dr. Ji's research interests include transportation and energy policy, transportation safety, and statistical analysis.

Email: sji1@utk.edu

Eric Wade holds a PhD in Mechanical Engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently an assistant professor at the University of Tennessee in the department of Mechanical, Aerosapace, and Biomedical Engineering. His research is focused on the application of engineering technologies to the domain of motor, neurological, and behavioral health issues.

Email: erwade@utk.edu