IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 19, 2019, accepted December 12, 2019, date of publication December 17, 2019, date of current version December 27, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2960409

Swing-In-Place (SIP): A Less Fatigue Walking-in-Place Method With Side-Viewing Functionality for Mobile Virtual Reality

YANGYI ANG, PUTERI SUHAIZA SULAIMAN, RAHMITA WIRZA O. K. RAHMAT, AND NORIS MOHD NOROWI

Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang 43400, Malaysia

Corresponding author: Yangyi Ang (racuss.ang@gmail.com)

This work was supported by the Ministry of Higher Education Malaysia under MyBrain15 Program.

ABSTRACT The concept of mobile Virtual Reality (VR) headset that utilize mobile smartphone as the main display and processing device has successfully achieved a low-cost but yet feasible method to implement VR for a mobile user. However, locomotion in mobile VR is still a challenge because the ways to interact with the smartphone are limited. Walking-in-place (WIP) can be used as a hands-free input method to control locomotion inside a mobile VR environment, at the same time, enhance user experience by increasing the sense of immersion. Commonly, WIP implementation in mobile VR uses the inertia sensors of smartphone to detect WIP, thus no additional apparatus is needed. However, common WIP implementation navigates based on user's gaze direction, but in reality, we may walk and look into different direction at the same time. Besides that, fatigue caused by WIP is another important issue to be considered for prolongs usage of VR application. So, we present a less fatigue WIP gesture, swing-in-place (SIP) with a WIP implementation that allows users to travel and look in different directions during locomotion. Our implementation used only the accelerometer and gyroscope in single smartphone. Evaluation results show that SIP able to provide a similar feeling of immersion and it is less fatigue when compared with a common WIP method used in mobile VR. The side-viewing functionality is liked by the users.

INDEX TERMS Mobile virtual reality, walking-in-place, locomotion technique, hands-free navigation, walking-in-place gesture, gestural interaction.

I. INTRODUCTION

Virtual Reality (VR) technology in recent years is undergoing a revival [1] along with the rapid development of mobile smartphone and the introduction of VR headset, which is a Ski-Masked Shaped Goggle device designed specifically for VR [2]. In recent years, many VR headsets which are portable and essentially rely on smartphone as the main unit for display and data processing have been released in the market, for example, Samsung Gear VR, Google Daydream, Pansonite VR, and etc. In this paper, we named this kind of mobile smartphone-dedicated VR products as mobile VR.

Mobile VR carries an important proportion in the VR industry, because it has the advantages of what a mobile application has, which are: anywhere, and anytime. Moreover, mobile VR can be realized with a relatively low cost because all we need for implementation is a smartphone and a smartphone-dedicated VR headset, while smartphone has already become one of our daily necessities, whereas the VR headset usually can be bought with a reasonable low price, range from $8 to $130 USD depending on the brand and specifications [3], and this kind of low cost smartphone VR headsets were actually can achieve similar immersive experience as those expensive head-mounted display such as Oculus Rift [4].

Locomotion, which is the ability to travel from one point to another point, is an important aspect for a VR application because locomotion is a fundamental human activity, and it provides a basic way for the users to explore the virtual environment. However, locomotion in mobile VR is

The associate editor coordinating the review of this manuscript and approving it for publication was Maurizio Tucci.

still a concern [5] of current mobile VR development environment since there are limited ways to interact with the smartphone [6] because the smartphone is being placed inside the VR headset when a user is exploring the virtual environment using the mobile VR application. Some of the common locomotion methods used in mobile VR including: using a controller, teleportation, auto navigation and walking-in-place [7], [8].

Using a controller for locomotion is reliable and easy to use, but it is less immersive than walking-in-place [8]. In fact, you will be pulling back to the reality every time you realize you have to press the button in order to move ahead. On the other hand, Teleportation required the user to point to a location to move on, and the user will "jump" to the location immediately after that. Teleportation usually use a controller or gaze selection as a pointer. The nature of teleportation, which is totally different from normal locomotion, can break the immersion feeling of users [8].

Next, auto navigation, or so-called "flying" is a method that follows user's gaze direction to travel forward automatically, regardless the actual intention of the user, for example, a user might not want to move at that moment. Hence, auto navigation method will lower down the feeling of immersion when compared to walking-in-place [9]. Nevertheless, auto navigation method will allow user under a hands-free condition. A navigation method with hands-free condition provides advantage for a VR application to maximize the ability of user interaction with the virtual environment [10], for instance, user can use their hands to do others operation such as interacting with objects in the virtual environment.

Walking-in-place is a better solution for locomotion in mobile VR because it can provide greater immersive feeling, at the same time, allows a hands-free situation. In addition, locomotion methods which using the user's own legs to perform a gesture similar to "walking" can reduce motion sickness [11]. Although real walking can definitely provide the greatest immersive feeling in a virtual environment, however, apply real walking in VR required complicated apparatus setup and preparation of a large area in the real world. Therefore, real walking is not suitable for mobile VR because of the spatial and apparatus constraints.

Implemented walking-in-place methods in most of the mobile VR applications rely on the inertia sensors such as accelerometer inside the smartphone, for example, the method used by VR-STEP [12]. By using the smartphone's built-in sensors, additional apparatus is not needed. However, this walking-in-place implementation can only navigate based on gaze direction, so user cannot travel towards a direction and look in a different direction simultaneously.

A research [13] proposed a method to track the sight view direction and body direction separately using an additional waist-worn sensor. However, this method required additional apparatus, which is not desirable for a normal mobile VR user. We want to enable users to look to the side while travelling forward during virtual locomotion without increasing the apparatus requirements.

Besides that, since walking-in-place method will definitely cause physical fatigue when compared to controller-based or teleportation method [14], less energy consumption [15] and low-fatigue [16] have become a vital concern for non-sport designed VR application which requires longer usage time because a general consumer just wants to experience VR leisurely.

This paper presents a walking-in-place implementation for mobile VR that can support different travel direction and sight view direction during the locomotion in VR environment, with a walking-alike gesture: Swing-In-Place (SIP) which is less fatigue than the gesture used in the common walking-in-place method for mobile VR. The SIP is a walking-alike gesture which required users to use their own legs to perform a walking-alike cycle, hence able to arise the feeling of "walking" when users are travelling in the virtual environment using the gesture. This is similar to the concept of walking-in-place method, thus it able to provide a similar sense of immersion as walking-in-place.

This implementation does not require additional apparatus except for the accelerometer and gyroscope available in a typical smartphone. In the rest of the paper, we discussed the related works and methodology of the proposed SIP implementation. Finally, a qualitative comparison between our SIP implementation with VR-STEP [12] method is presented.

II. RELATED WORKS

Walking-in-place implementations for locomotion in VR have been introduced in early years, [17] use an electromagnetic tracking device on the head-mounted devices to analyze the user's movement patterns for step recognition. Although their implementation has a limitation regarding step latency, which is an important aspect for virtual locomotion to maintain immersive user experience, but it is a good starting point for the following researchers.

Later, researchers strive to overcome the limitation of step latency by using more sensors placed on different parts of the user's body. Yan *et al.* [18] present an implementation that required users to place sensors on legs, back, and head for more accurate motion detection without step latency. Then, Feasel et al. [19] have introduced a method using two magnetic foot trackers placed on the user's legs and a chest tracker to enable tracking of body orientation. Wendt *et al.* [20] and Bruno *et al.* [21] both used optical motion tracking system to track the leg movement for step detection, which means that the users have to wear a motion suit or motion trackers on their body and stand in front of an optical tracking camera. Wendt *et al.* [20] apply biomechanics gait to further enhance the analysis of stepping motion, whereas Bruno *et al.* [21] use the footstep amplitude and speed metric instead of frequency metric to reflect the locomotion speeds.

By placing sensors or trackers on different parts of the body, step latency issues can be overcome [18], [19], and later, researchers [20], [21] change their focus on controlling locomotion speed. However, these implementations [18]–[21] have to be conducted with full apparatus setup supported by different devices or laboratory requirements, thus they are not feasible to be applied in mobile VR.

There are researches implement WIP method without placing sensors on users' body but with the support of physical interface, such as a pressure mat [22], [23], treadmill-like devices [24], [25], CAVE system [26], or special shoes [27]. These methods required support of the special devices, those devices are usually a prototype which is hard to obtain by a normal user, and hence, it is not suitable for mobile VR as well. Ease of implementation is one of the important issues to be considered when dealing with mobile VR.

So, moving forward to the issues of ease of implementation and low costing, the trend of researches on WIP implementations had moved into the utilization of build-in sensors in smartphones, or recent commercial gaming devices such as Wiimote [28], Nintendo Wii Balance Board [29], [30], and Microsoft Kinect [31]–[34] for tracking the WIP motion. Using the ready-made components inside our household digital devices, such as accelerometer in Wiimote, pressure sensors in Nintendo Wii Balance Board, or the infrared camera in Kinect taking the advantage to lower down the setup requirements and therefore able to reduce the implementation cost, however, the popularity of these household digital devices is not as wide as smartphone. Therefore, there is a high possibility that a mobile VR user does not possess the required devices.

Others than household digital devices, smartphone is another powerful device which built-in with several sensors that can be utilized to sense body motion. A research [35] used two smartphones and a magnet attached to the user's legs to detect the WIP motion. In this method, ease of implementation is still an issue because two smartphones and a magnet is required, and the smartphones have to be placed on the user's legs, so it is not feasible for mobile VR, because in mobile VR, the smartphone has to be placed inside the VR headset.

In recent years, researchers have started to explore on apparatus free WIP implementation for mobile VR due to the introduction of mobile VR headset. Tregillus and Folmer [12] have introduced a WIP implementation, VR-STEP. Their implementation does not require additional apparatus, and uses only the smartphone's sensors that placed inside the VR headset, thus is feasible for mobile VR. The researchers have released a plugin of VR-STEP for Unity [36]. This implementation senses the upward acceleration of the smartphone to detect a step, therefore, this method required users to perform a gesture which is relatively similar as jogging in order to create an upward motion. Hence, this gesture will cause fatigue when the usage time is too long. In addition, this method only supports navigation with gaze direction. Concurrently, [37] presents a similar WIP method, and they claim that their method uses a low pass filter to allow small movements of users' head, this will enable users to look to another direction during fast pace running. However, their method can only avoid changing of direction abruptly, the body orientation will follows back the head direction slowly afterwards, which means users cannot look to different directions freely. Besides that, their methodology explanation about the side view function is vague and no proper experimental results showing the user experience in regards the function to allow user look to the side.

Later, [38] presented a method that uses head-tilt to indicate the direction to travel and combine with the VR-STEP [12] WIP implementation to move forward. For example, when a user left-tilt his head will cause the navigation direction slide to the left horizontally, which is much more like sideway walking, but not controlling the view direction follows head rotation. In [39], similar walking-in-place method which supports sliding to different direction was proposed. They use the angle of head to indicate the direction of navigation, their method support forward, backward, and sideways walking. These implementations [38], [39] have an interesting way to control the moving direction, but the side direction they can control is the horizontal direction, which is either slide the sight view left or right, but not follows the direction of head rotation, which means that the user is still not able to walk and look to a different direction at the same time. However, we can see that researchers strive to introduce different ways to control the moving direction in mobile VR locomotion.

Lee *et al.* [40] have proposed a WIP method which is similar as VR-STEP [12], which their implementation used inertia sensors in a VR head-mounted device (HMD) to detect upward acceleration when user performing a jog-like gesture, with high step detection accuracy. Their evaluation shows that their step detection accuracy is 99.32%, since previous research [12] does not prove the accuracy of their implementation, this accuracy is not comparable. However, same with VR-STEP [12], their proposed method can only navigates based on gaze direction, and their jog-in-place gesture is still a gesture which will cause high level of physical strain when continue for a longer period of time.

In order to support different view and actual travel direction, Park et al. [13] have proposed to use the combination of sensors in HMD with a waist-worn sensor to capture the orientation of the user's head and pelvis. Their method utilizes the pelvis direction as the travel direction and head direction as the view direction, nevertheless, this method requires additional sensor to be placed on the user's waist.

When come to the issue of fatigue, [15] has proposed to use arm swing instead of walking-in-place for locomotion in order to reduce the physical fatigue and thus achieve prolong use of VR application. On the other hand, [16] proposed a locomotion technique which use two controllers on users' hand, and users can move forward by pulling the trigger of the controller, and control the direction of movement using the orientation of the controllers. Both techniques [15], [16] can greatly reduce the fatigue level as compared to walking-in-place since users need not move their legs continuously. Even though they show that sense of presence was preserved in their method as compared to walking-in-place, but their method required holding of two controllers on hand, which will not able to leave user on a hands free condition, and both methods are only applicable to those VR devices with two controllers, such as HTC Vive, which are not suitable to be used in mobile VR.

III. PROPOSED METHODOLOGIES

Common WIP implementations used in mobile VR, such as VR-STEP [12] uses inertia sensing to detect a step and navigate follow the gaze direction inside the mobile virtual environment. However, in reality, we may walk toward a direction, but at the same time, looking in another direction. VR-STEP use upward acceleration of the smartphone to detect a step, thus it is hard to realize locomotion which allows different view and travel direction using only the sensors in a smartphone. In order to differentiate the view and travel direction of the user with limited sensors available in a mobile smartphone, we proposed a walking-alike gesture which is similar to walking-in-place, named swing-in-place (SIP). The proposed SIP gesture can generate different acceleration patterns when a user is on different view direction and travel direction while performing the SIP gesture. Moreover, the fatigue level of the proposed SIP gesture is lower as compared to the common walking-in-place gesture used in mobile VR, for example, the jogging-alike gesture used in VR-STEP.

A. SWING-IN-PLACE (SIP) GESTURE

The proposed SIP gesture is illustrated in Fig. 1. SIP gesture generates acceleration by lift-up one leg from the floor, so that the body will be lean to the opposite side, this create sudden acceleration, which the scale will be significantly differ from small body movement while standing. The gesture for a continuous step is to land the floating leg and lift-up the opposite leg at the same time, this will make the body lean back to another side, and create the sudden acceleration to the opposite direction. Users can perform the gesture continuously to simulate the cycle of walking circumstance. SIP gesture minimizes the need of leg and body movement, therefore, able to reduce the physical fatigue caused by performing the gesture as compared to the common jogging gesture used in others WIP implementation for mobile VR.

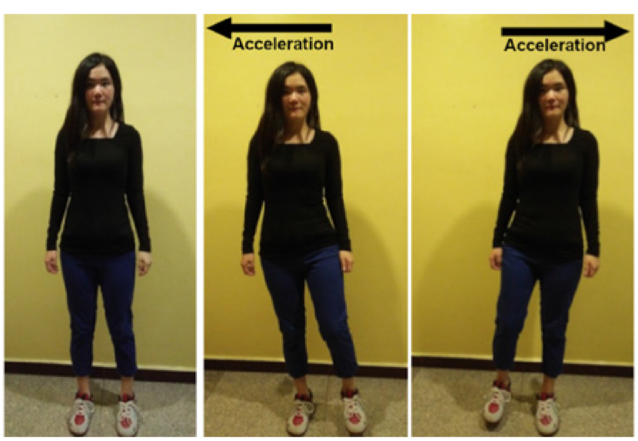
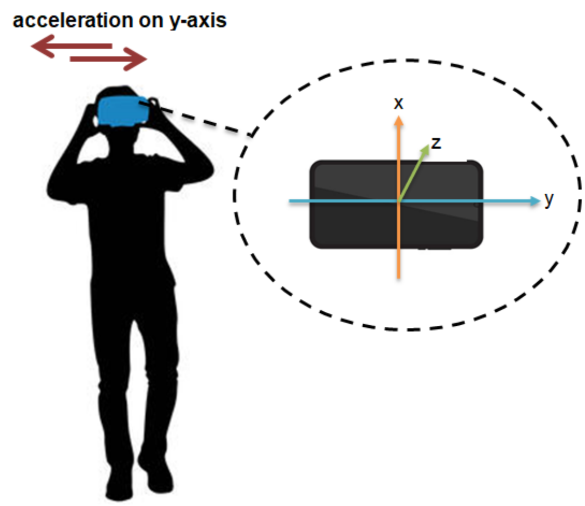
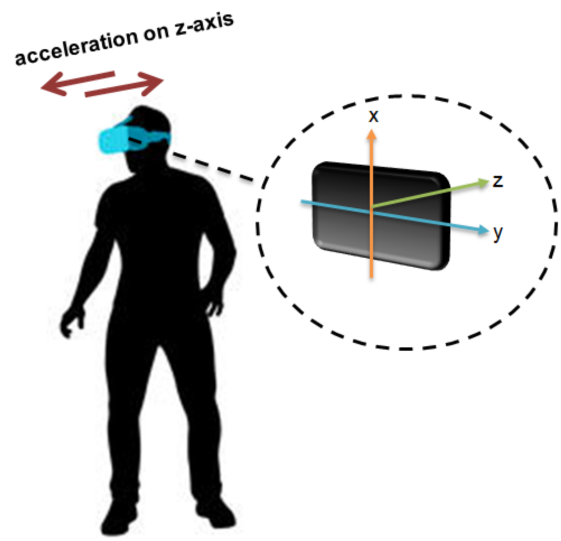


FIGURE 1. Swing-in-place gesture.

Swing-in-place (SIP) gesture able to generate different acceleration patterns for the two situations: (1) when a user is performing SIP gesture and look to the front, and (2) when a user is performing SIP gesture and look to side direction. This is because when a user is under the two different situations, the orientation of the smartphone inside the headset will change as illustrated in Fig.2.



(a)

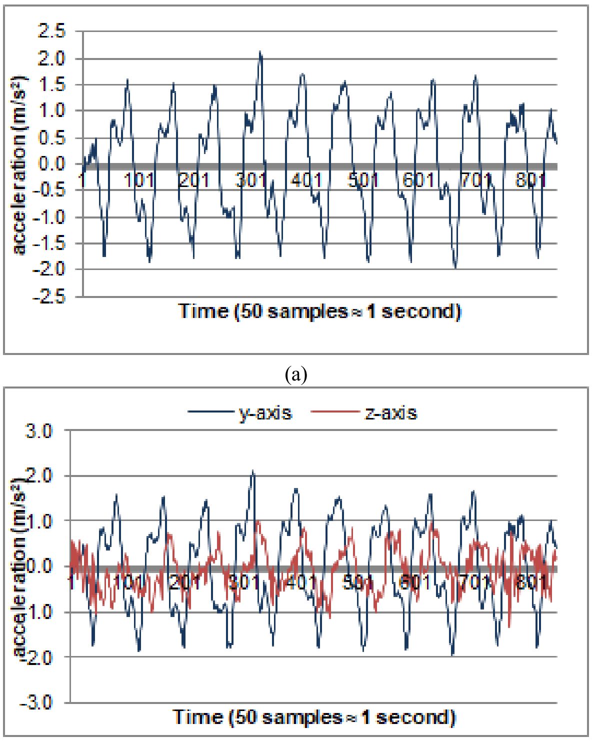


(b)

FIGURE 2. (a) Acceleration axes and orientation of smartphone when a user is looking to front direction. (b) Acceleration axes and orientation of smartphone when a user is looking to side direction.

As illustrated in Fig. 2(a), when a user is looking to the front and performing SIP, the smartphone inside the headset will accelerate horizontally, thus generates acceleration patterns on the y-axis of the accelerometer. On the other hand, when a user is looking to the side direction and performing SIP, the smartphone's orientation will change as illustrated in Fig. 2(b). When the smartphone is under side orientation, if the user is performing the SIP gesture, it will cause the smartphone accelerate forth and back, and a significant acceleration pattern on z-axis can be detected.

Fig. 3 and Fig. 4 show the acceleration patterns on both y-axis and z-axis for two situations. When a user is looking to front direction and performs SIP gesture, we can get a significant acceleration pattern on the y-axis as shown in Fig. 3 (a). During this situation, although there are accelerations on z-axis as well as illustrated in Fig. 3(b), however, the z-axis acceleration does not form an obvious pattern as compared to y-axis acceleration. Therefore, we can identify the "look forward" situation by analyzing the acceleration pattern on the y-axis.



(a)

(b)

FIGURE 3. **(a) Y-axis acceleration pattern for 10 steps when a user is looking to front direction and performing the SIP gesture. (b) Comparison of y-axis and z-axis acceleration when a user is looking to front direction and performing the SIP gesture.**

The acceleration patterns will change when a user is looking to the side direction and performs SIP gesture. Fig. 4 shows the acceleration patterns of y-axis and z-axis when a user is looking to the left direction and performing SIP gesture. Obviously, the y-axis acceleration pattern under this situation was distorted, but alternatively, we get a significant acceleration pattern on z-axis. This is because when a user turn his head to the left direction, the smartphone orientation was rotated, performing SIP gesture when the smartphone is in this orientation will cause the phone to accelerate forth

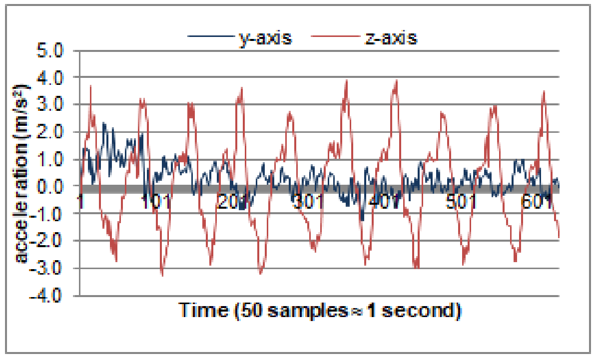


FIGURE 4. **Y-axis and z-axis acceleration for 10 steps when a user is looking to the left direction and performing the SIP gesture.**

and back continuously as illustrated previously in Fig. 2 (b). As the result, this situation will generating an acceleration pattern which is similar to the pattern when a user is looking to the front direction, but this time, the pattern is getting from the z-axis. Hence, we can utilize the acceleration pattern of these two axes to differentiate the situations and identify whether the user is looking to the front direction or looking to the side direction while the user is performing the SIP gesture.

B. STEP AND SITUATION DETECTION

The implementation of SIP step and situation detection required two sensors in a smartphone, which are: 3-axis accelerometer and gyroscope. These two sensors are very common and we can find it in most of the smartphones nowadays. Accelerometer readings are used to detect a step, whereas gyroscope readings are used to detect head turning. We average every 5 samples of the accelerometer and gyroscope readings to smooth the signals and reduce noise as the algorithm used by Zhao [41] and VR-STEP [12]. Our implementation uses several thresholds to detect different situations when the users performing the SIP gesture. The situations include: (1) Standing, (2) Step Initialization, (3) Swing-in-Place (SIP), (4) Swing-in-Place and look to the side (SIP + Side View), and (5) Turning Around.

We use two thresholds: a lower threshold (negative value) and an upper threshold (positive value) to recognize the acceleration patterns of different situations. This is because when a user is performing the SIP gesture stepping from left to right then left, the acceleration values will continuously create a cycle from negative to positive then to negative or vice versa.

The thresholds for Standing, Step Initialization, and Turning Around are calculated using mean $\pm$ constant(σ) to form a range with a lower boundary and an upper boundary. The mean and standard deviation (σ) is calculated from the data getting from accelerometer and gyroscope when the user is under the situation of "Standing". For the "Standing" situation, we used accelerometer y-axis readings and a constant of 3 to calculate the STABLE thresholds for accelerometer input event, readings which fall in between the range of the thresholds indicates the user is on the "Standing" situation. The thresholds for detecting a "Step Initialization" were calculated using a constant of 4. Accelerometer y-axis readings exceed the upper or lower threshold denote a left or right step initialization. For the ''Turning Around'' situation, we used gyroscope x-axis readings and a constant of 4 to calculate the STABLE thresholds for gyroscope input event. Gyroscope readings exceed the range denote a slightly larger head rotation as compared to the normal situation, thus can be perceived as an indication of the ''Turning Around'' state. Besides that, these STABLE thresholds were also used to detect a user's head turning while performing the SIP gesture. So, when a user turns his head during ''SIP'' situation and the head rotation angle is larger enough to create a rotational motion that exceeds the STABLE thresholds for the gyroscope input event, then it indicates a head turning event, and this indication will be used as a factor for the decision making process in the accelerometer input event algorithm.

The upper and lower thresholds for ''SIP'' and ''SIP + Side View'' states are calculated by mean $\pm$ constant(σ) as well, but the upper threshold uses the mean of positive values minus standard deviation of positive values multiplied by a constant, whereas the lower threshold use the mean of negative values plus standard deviation of negative values multiplied by a constant. The constant value is used to adjust the sensitivity of the step detection. We used a constant value of 1 in our implementation. The positive and negative values for means and standard deviations calculation are based on the acceleration data generated by users when users are performing SIP gesture. Firstly, we remove the values that are falling in between the range of ''Standing'', which are those values that closed to zero in order to remove noises. Then, the acceleration values were categorized into positive values and negative values for thresholds calculation. The thresholds for ''SIP'' state are calculated based on y-axis acceleration values, readings exceed the upper threshold or lower threshold denote a left or right step respectively. On the other hand, the thresholds for ''SIP + Side View'' situation are calculated based on the z-axis acceleration readings when users are looking to the side and performing SIP gesture. Two pairs of thresholds were calculated for the ''SIP + Side View'' situation, one for left viewing and one for right viewing.

Fig. 5 shows the flow chart of the algorithm for the accelerometer input event. For every accelerometer and gyroscope input event, 5 samples of y-axis and z-axis data from accelerometer, and x-axis data from gyroscope will be averaged for smoothing and noise removal purpose, then the averaged x-value, y-value and z-value will be used as the input of the algorithm. Basically, the algorithm detects the user's current situation based on current state and simple thresholding. The corresponding thresholds will be used depending on the user's current state and the situation to be tested. The user's states to be recognized included ''Standing'', ''Walking'', ''Side View'', ''Left Stepping'', ''Right Stepping'', ''Left View'', ''Right View'', and ''Turning''. First of all, the algorithm will check if the user is standing stably currently, if yes, a ''Standing'' state will be set. Then, the state checking process will proceed based on the following sequence: Side View > Walking > Standing. This sequence is designed in such a way that the former is predominant than the latter, for example, a ''Side View'' state will only be fired when the user is on ''Walking'' state, and a ''Walking'' state will only be fired if the user is on ''Standing'' state previously.

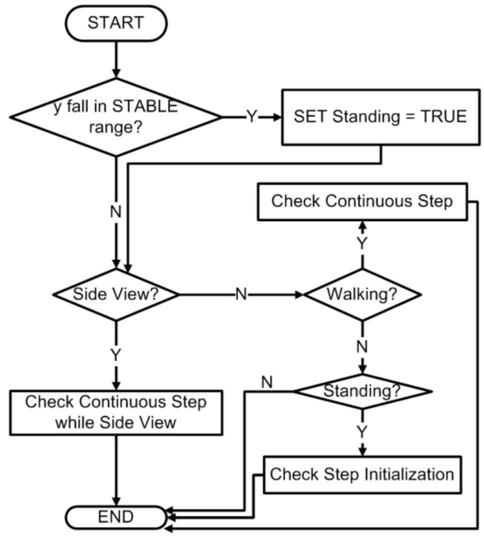


FIGURE 5. Accelerometer input event algorithm.

For instance, given user's current state is ''Standing'', the algorithm will check whether if the user has initiate a step. Step initialization is test using the corresponding thresholds. If the y-value has exceeds the thresholds ($y <$ negative threshold or $y >$ positive threshold), ''Walking'' state will be set, with an additional state of ''Left Stepping'' or ''Right Stepping'', depending on which threshold is passed.

So, during the next input event, given the user's current states are ''Walking'' and ''Left Stepping'', then the algorithm will test if the user continues the gesture by stepping to the right. There will be an interval in between one step and another step, the time interval in between steps was controlled by a sampling counter, the sampling counter will increase every time a sample data getting from the accelerometer. When the sampling counter has exceeds the allowed interval time, it will be reset. So, if in between the interval, no following step was detected, then the user's state of ''Walking'' will be discarded. On the other hand, if a following step was detected, the ''Walking'' state will continue and the state will change from ''Left Stepping'' to ''Right Stepping'' or vice versa.

Next, we use the combination of both accelerometer and gyroscope input event to detect the situation of ''Side View + SIP''. The ''Side View'' state will be initialized from the gyroscope input event as shown in Fig. 6. If the gyroscope x-value exceeds the STABLE thresholds, it means head turning is detected. If a head turning is detected but the user current's state is ''Standing'', then it indicates a ''Turning Around'' situation, which the user is just turning his head

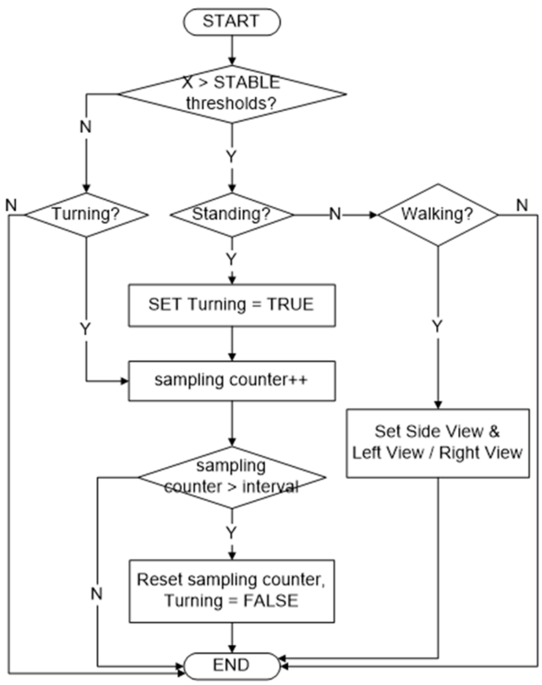


FIGURE 6. Gyroscope input event algorithm.

without walking. On the other hand, if head turning is detected and the user's current state is "Walking", which means that the user is walking and looking to the side direction, then two additional states: "Side View", and "Left View" or "Right View" will be set depending on the direction of the user's head rotation. When the user's current's states include "Walking" and "Side View", previous direction before head turning will be used as the travel direction, whereas current head direction will be used as the gaze direction, so that the user will be able to look to the side, but keep on travel with previous direction.

When the user is under the "Side View" state, accelerometer z-value will be used for continuous step checking in the accelerometer input event algorithm. The corresponding thresholds will be used depending on user's current state, whether is "Left View" or "Right View". An interval is given for detecting a following step, if no following step is detected within the interval, the "Side View" state will be discarded and left only "Walking" state, and the sampling counter for the "Walking" state will be increased to 3/4 of the total interval because the interval time for a step have been consumed during the "Side View" state step detection.

Once the "Side View" state is discarded, the user's travel direction and gaze direction will follow back the head direction as "Walking" state, and the continuous step checking will change back to base on y-axis of the accelerometer.

For the "Turning Around" situation, gyroscope x-axis readings were used to detect the user's head turning event as shown in Fig. 6. If head turning is detected and the user is on "Standing" state, "Turning" state will be fired. During "Turning" state, acceleration input event will be ignore to avoid step detection when the user is just trying to look around without travel. There will be an interval for the user to turn and look around. After the interval, "Turning" state will be discarded.

IV. EXPERIMENT

We evaluate SIP implementation by comparing it with VR-STEP [12], a common WIP method used for mobile VR locomotion. The evaluation compared the two WIP methods in terms of fatigue level and immersive feeling. We have a total of 20 volunteers participate in the user evaluation (Average Age = 20.55).

A. SETUP

We used Lenovo K6 Note smartphone and VR Shinecone headset to conduct the experiment. We developed a mobile VR Android application implemented with the SIP method for locomotion using Cardboard SDK and JPCT-AE to render a street view 3D virtual environment. We used a free 3D model available from [42] as shown in Fig. 7 as the virtual environment in this experiment, where Fig. 7 (a) shows the user's perspective of the virtual environment while Fig.7 (b) shows the bird view perspective of the virtual environment and the setting of the virtual space. As illustrated in Fig. 7(b), the street view virtual environment is a square space with a crossroads in the middle of the virtual space, where both sides of the roads are surrounded by buildings. Users can only travel on the roads and there are no obstacles on the roads. Users are not allowed to pass through the borders, thus they are not possible to enter the buildings area. The length of the virtual model of the road from one end to another end is 48 meters. The time taken for a user to travel from one end to another end is approximately 20 to 30 seconds. Since the time taken to travel is affected by the user's behavior when performing the gesture, for example, some users like to perform the gesture faster, and some users like to perform the gesture slowly, thus we provide an approximate range of time for reference.

For the VR-STEP application, we used Unity to render the same virtual environment and applied the VR-STEP plugin [36] for locomotion. The plugin is available for purchase from the Unity asset store. Both applications are under the same circumstance to avoid bias caused by factors such as graphic quality, 3D models' quality, and environment.

B. MEASURES

We used questionnaire to assess the subjective feeling of the users. A set of questions from Slater-Usoh-Steed Questionnaire [43], which is a questionnaire commonly used by researchers to measure the sense of presence were included in our questionnaire. The questions from Slater-Usoh-Steed Questionnaire including: sense of "being there", virtual environment (VE) becomes the reality, and VE perceived as images or place. We measure the following criteria using a rating scale from "1" to "7":

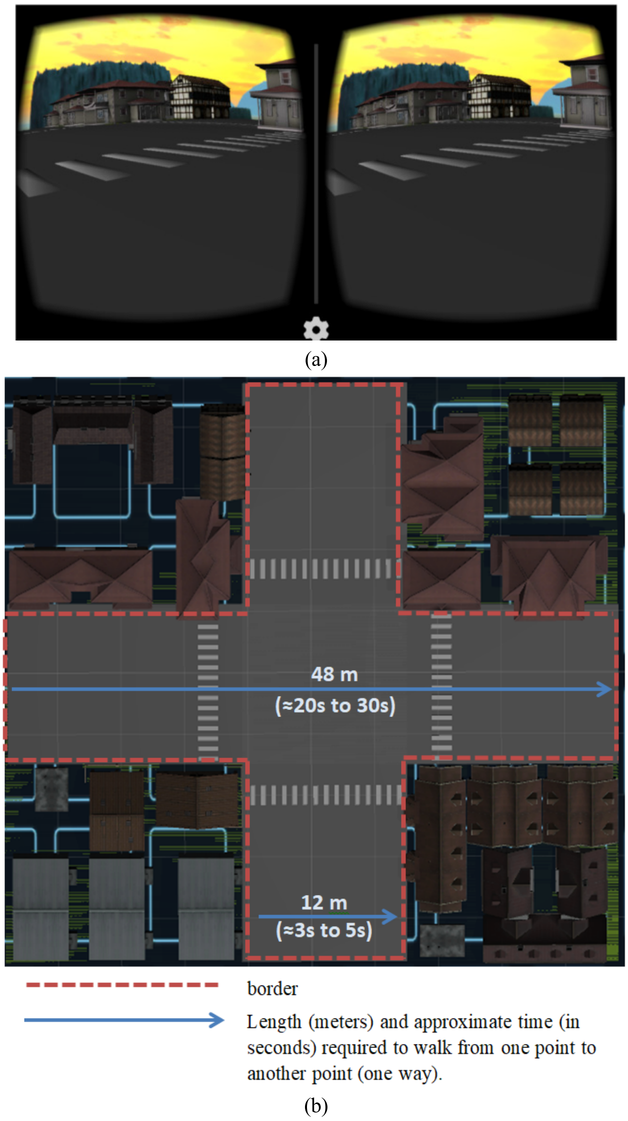


FIGURE 7. **Street view virtual environment used in the experiment. (a) User's perspective. (b) Bird view perspective and the setting of the virtual space.**

- **Fatigue Level**: to measure how fatigue the users feel when they use the walking-in-place method to navigate. Rating of "1" indicates low fatigue level while "7" indicates high fatigue level.
- **Sense of "being there"**: to evaluate how strong was the feeling of the users feel like they are "being there" when they use the walking-in-place method to navigate. Rating of "1" indicates low sense of "being there" while "7" indicates strong sense of "being there".
- **VE becomes the reality**: to measure if there was a time when the users feel that the virtual environment has become reality for them. Rating of "1" indicates "at no time" while "7" indicates "all the time".
- **VE perceived as images or place**: to measure if the virtual environment perceived as a set of images they have seen or perceived as a place which have been visited. Rating of "1" indicates "perceived as images" while "7" indicates "perceived as place".

Besides that, two additional questions regarding the "Side View" feature provided by the SIP method were asked. Users can answer the questions with "Yes", "No", or "Not sure". The questions are as follows:

- Does the "Side View" feature increase immersive feeling?
- Do you like the "Side View" feature?

C. PROCEDURES

Briefing and demonstration of both methods: SIP and VR-STEP were given before the experiment started. We used within-subject design for the experiment, every participant has to navigate in the virtual environment using both methods, and the sequence of which method to be used first was allocated alternately. During the experiment, participants were asked to travel to the four borders of the virtual environments as shown in Fig. 8. Participants are free to move and look around throughout the experiment. During the experiment session for the SIP method, we reminded every participant to try out the "Side View" feature while they are travelling to make sure every participant have tested the feature. Fig. 9 shows an example when a participant is performing the SIP gesture and look to the side direction.

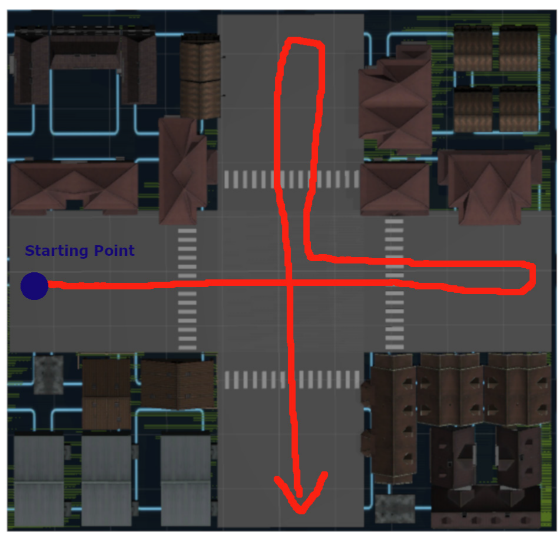


FIGURE 8. **The navigation path of the experiment.**

V. RESULTS AND DISCUSSION

Table 1 shows the means and standard deviations (sd) of the ratings for the subjective measures. Two-sample t-test was conducted to analyze if there is any significant different between the rating of SIP and VR-STEP. We found significant difference when comparing the fatigue level of SIP and VR-STEP ($t(19) = -5.5816$, $p < 0.001$), the result shows that SIP method is less fatigue than the VR-STEP method. Since the SIP gesture can be performed with a more moderate

FIGURE 9. A user performing the SIP gesture and testing the "Side View" feature.

TABLE 1. Results of the ratings of SIP and VR-STEP. T-values are obtained from two-tailed two-sample t-test. The value indicates significant difference was bold.

	SIP		VR-STEP		t-value
	mean	sd	mean	sd	
Fatigue level	2.75	0.97	5.00	1.52	**5.5816 (p<0.001)**
Sense of "being there"	4.05	0.999	4.05	1.19	0
VE become the reality	3.55	1.32	3.55	1.28	0
VE perceived as images or place	3.30	1.56	3.05	1.54	0.5105

body movement as compare to the VR-STEP gesture which required frequent head bouncing, thus the SIP method allow the user to complete the locomotion task in a less fatigue and less intense way than the VR-STEP method.

However, there were no significant differences found between SIP and VR-STEP when compared with the immersive feeling based on the Slater-Usoh-Steed Questionnaire's [43] questions. From our opinions, since SIP and VR-STEP are both a WIP method, therefore the senses of presence provided by these two methods are similar. We argue that Slater-Usoh-Steed questionnaire is more suitable to be used for comparison between different kinds of locomotion methods, for example, real walking, walking-in-place, and controller, rather than comparing the same type of locomotion method. Similar circumstance can be seen in previous research [44] which evaluate and compare different types of walking-in-place gestures. Their results show that there were also no significant differences found when they assess the sense of presence using the Slater-Usoh-Steed Questionnaire's questions.

Although SIP cannot achieve significant improvement in terms of immersive feeling, however, SIP implementation able to achieve same level of immersion as VR-STEP, this prove that the SIP gesture itself is acceptable as a WIP locomotion method because we actually get a very similar mean values of rating when measuring the immersive feeling of the two methods using the Slater-Usoh-Steed Questionnaire's questions. In addition, SIP has the extra feature to support different travel and view direction, and this feature is fairly liked by users, which can be seen in the following results' discussion.

The results of the two additional questions regarding the "Side View" feature of the SIP method were illustrated in Fig. 10 and Fig. 11. According to the results shown in Fig. 10, there are 55% of the participants agree that the ability to look to different directions during locomotion can increase the immersive feeling. There are 25% of participants disagreeing with this and 20% of participants choose "not sure". This might because it is hard to express the feeling of immersive, thus some of the participants choose to not give determination when answering this question. Nevertheless, we still get the majority of agreement on this question.

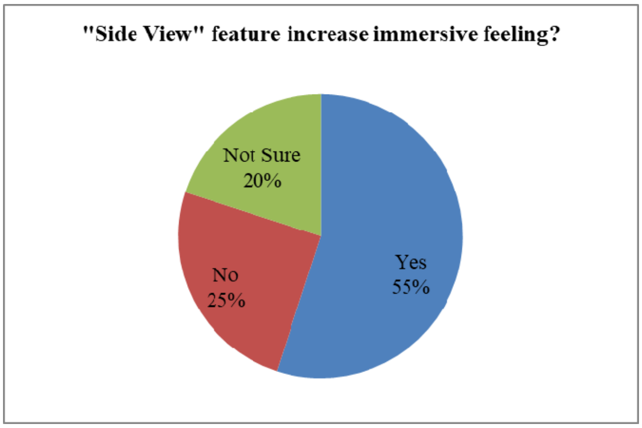


FIGURE 10. Results of the question asking if the "Side View" feature can increase immersive feeling.

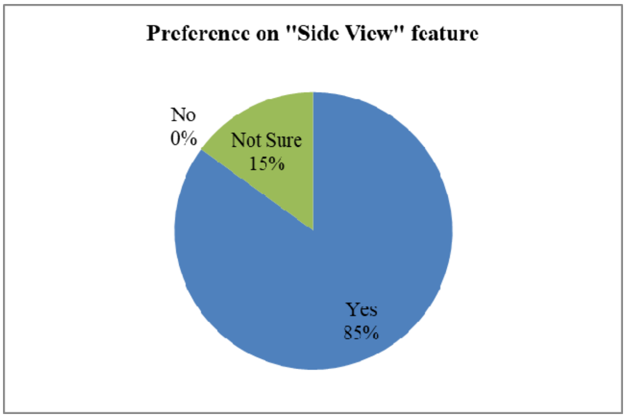


FIGURE 11. Results of the question asking preference on the "Side View" feature.

Next, Fig. 11 shows that 85% of the participants like the "Side View" feature. This is because the "Side View" feature is similar to the real world situation, which we can look around while walking, but the previous WIP method in mobile VR such as VR-STEP only allows navigating based on the gaze direction, thus the SIP method give the user a fresh experience. Even though there are 15% of participants who are not sure, but there is no participant dislikes the feature. Therefore, we can conclude that users like the ''Side View'' feature which allows them to look to different directions during locomotion in the mobile VR application.

VI. LIMITATION AND FUTURE WORKS

SIP implementation enables users to look to different directions while travelling in the virtual reality environment, however, this can only achieved during the moment when the user is travelling. For example, a user cannot initiate side view navigation when he is standing still with his body facing one direction and looking to another direction. Besides that, users have to stop the locomotion in order to make a rotation and turn to others direction. Next, current thresholds used in our implementation are based on fixed data samples. For future works, dynamic thresholds and user's calibration may be applied so that the thresholds value will be calculated from real time data based on the user's data. This may increase the effectiveness of the implementation because different users have different practice and body movement when they performing the gesture, and the acceleration might be affected by the user's height. For example, a user with longer legs might generate larger acceleration as compared to a user with shorter legs. In addition, we will explore a way to control the locomotion speed using the ''swing'' amplitude of the swing-in-place gesture instead of step frequency to reduce the fatigue level when the user want to navigate faster in the virtual environment.

VII. CONCLUSION

In this paper, we presented a walking-in-place (WIP) implementation that work with a walking-alike gesture, Swing-In-Place (SIP), which is less fatigue than common WIP locomotion in mobile VR. Our implementation has a feature to enable the users to travel and look to different directions simultaneously. This implementation utilizes only accelerometer and gyroscope in a smartphone to capture body movement and head rotation, and then analyze the acceleration patterns on different axes to identify different situations of the user when the user is performing the SIP gesture. The results of the user study reported that the SIP implementation is less fatigue and able to achieve same level of immersive feeling when compared to a common WIP implementation method for mobile VR. Users like the introduced ''Side View'' feature and majority agree that this feature will increase the level of immersion during locomotion in VR environment.

REFERENCES

[1] C. Boletsis, ''The new era of virtual reality locomotion: A systematic literature review of techniques and a proposed typology,'' *Multimodal technol. Interact.*, vol. 1, no. 4, p. 24, 2017.

[2] P. R. Desai, P. N. Desai, K. D. Ajmera, and K. Mehta, ''A review paper on oculus rift-a virtual reality headset,'' 2014, *arXiv:1408.1173*. [Online]. Available: https://arxiv.org/abs/1408.1173

[3] S. Noble, (2019). *The 16 Best Smartphone Virtual Reality Headsets 2019*. Accessed: Jun. 23, 2019. [Online]. Available: https://www.aniwaa.com/best-of/vr-ar/best-smartphone-vr-headset/

[4] N. M. Papachristos, I. Vrellis, and T. A. Mikropoulos, ''A comparison between oculus rift and a low-cost smartphone vr headset: Immersive user experience and learning,'' in *Proc. IEEE 17th Int. Conf. Adv. Learn. Technol. (ICALT)*, Jul. 2017, pp. 477–481.

[5] C. Boletsis, J. E. Cedergren, and S. Kongsvik, ''Hci research in virtual reality: A discussion of problem-solving,'' in *Proc. Int. Conf. Inter. Hum. Comput. Interact., (IHCI)*, Jul. 2017, pp. 21–23.

[6] C. Anthes, R. J. García-Hernández, M. Wiedemann, and D. Kranzlüller, ''State of the art of virtual reality technology,'' in *Proc. IEEE Aerosp. Conf.*, Mar. 2016, pp. 1–19.

[7] Y. Y. Ang, P. S. Sulaiman, R. W. O. Rahmat, and N. M. Norowi, ''Put down the controller, enable 'walking' in a virtual reality (VR) environment: A review,'' in *Ambient Communications and Computer Systems*. Springer, 2018, pp. 367–379. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-10-7386-1_32

[8] C. Boletsis and J. E. Cedergren, ''Vr locomotion in the new era of virtual reality: An empirical comparison of prevalent techniques,'' *Adv. Hum.-Comput. Interact.*, vol. 2019, Feb. 2019, Art. no. 7420781.

[9] 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 *Proc. 26th Annu. Conf. Comput. Graph. Interact. Techn.*, Boston, MA, USA: Addison-Wesley, 1999, pp. 359–364.

[10] J. J. LaViola, Jr., D. A. Feliz, D. F. Keefe, and R. C. Zeleznik, ''Hands-free multi-scale navigation in virtual environments,'' in *Proc. SI3D*, vol. 1, 2001, pp. 9–15.

[11] J. Lee, M. Kim, and J. Kim, ''A study on immersion and vr sickness in walking interaction for Immersive virtual reality applications,'' *Symmetry*, vol. 9, no. 5, p. 78, 2017.

[12] S. Tregillus and E. Folmer, ''Vr-step: Walking-in-place using inertial sensing for hands free navigation in mobile VR environments,'' in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2016, pp. 1250–1255.

[13] C. Park, K. Jang, and J. Lee, ''Walking-in-place for VR navigation independent of gaze direction using a waist-worn inertial measurement unit,'' in *Proc. IEEE Int. Symp. Mixed Augmented Reality Adjunct (ISMAR-Adjunct)*, Oct. 2018, pp. 254–257.

[14] E. Bozgeyikli, A. Raij, S. Katkoori, and R. Dubey, ''Point & teleport locomotion technique for virtual reality,'' in *Proc. Annu. Symp. Comput.-Hum. Interact. Play.*, 2016, pp. 205–216.

[15] Y. S. Pai and K. Kunze, ''Armswing: Using arm swings for accessible and immersive navigation in ar/vr spaces,'' in *Proc. 16th Int. Conf. Mobile Ubiquitous Multimedia.*, 2017, pp. 189–198.

[16] B. Sarupuri, S. Hoermann, F. Steinicke, and R. W. Lindeman, ''Trigger-walking: A biomechanically-inspired locomotion user interface for efficient realistic virtual walking,'' in *Proc. 5th Symp. Spatial User Interact.*, 2017, pp. 138–147.

[17] M. Slater, M. Usoh, and A. Steed, ''Taking steps: The influence of a walking technique on presence in virtual reality,'' *ACM Trans. Comput.-Hum. Interact. (TOCHI)*, vol. 2, no. 3, pp. 201–219, 1995.

[18] L. Yan, R. Allison, and S. Rushton, ''New simple virtual walking method walking on the spot,'' in *Proc. IPT Symp.*, 2004.

[19] J. Feasel, M. C. Whitton, and J. D. Wendt, ''LLCM-WIP: Low-latency, continuous-motion walking-in-place,'' in *Proc. IEEE Symp. 3D User Interfaces*, Mar. 2008, pp. 97–104.

[20] J. D. Wendt, M. C. Whitton, and F. P. Brooks, ''GUD WIP: Gait-understanding-driven walking-in-place,'' in *Proc. IEEE Virtual Reality Conf. (VR)*, Mar. 2010, pp. 51–58.

[21] L. Bruno, J. Pereira, and J. Jorge, ''A new approach to walking in place,'' in *Proc. IFIP Conf. Human-Comput. Interact.* Springer, 2013, pp. 370–387. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-40477-1_23

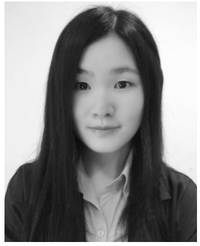
[22] W. Couvillion, ''The pressure mat: A new device for traversing virtual environments using natural motion,'' in *Proc. Interservice/Industry Training Simulation Edu. Conf.*, 2001, pp. 199–211.

[23] K.-R. Mun, H. Jeong, H. Oh, and J. Kim, ''A learning-based gait estimation during walking-in-place in VR locomotion system,'' *ISBS Proc. Arch.*, vol. 36, no. 1, p. 618, 2018.

[24] H. Iwata, ''The torus treadmill: Realizing locomotion in VEs,'' *IEEE Comput. Graph. Appl.*, vol. 19, no. 6, pp. 30–35, Nov. 1999.

[25] T. Cakmak and H. Hager, ''Cyberith virtualizer: A locomotion device for virtual reality,'' in *Proc. ACM SIGGRAPH Emerg. Technol.*, 2014, p. 6.

[26] D. J. Zielinski, R. P. McMahan, and R. B. Brady, "Shadow walking: An unencumbered locomotion technique for systems with underfloor projection," in *Proc. IEEE Virtual Reality Conf.*, Mar. 2011, pp. 167–170.
[27] D. Swapp, J. Williams, and A. Steed, "The implementation of a novel walking interface within an immersive display," in *Proc. IEEE Symp. 3D User Interfaves (3DUI)*, Mar. 2010, pp. 71–74.
[28] T. Shiratori and J. K. Hodgins, "Accelerometer-based user interfaces for the control of a physically simulated character," *Trans. Graph.*, vol. 27, no. 5, p. 123, 2008.
[29] A. Harris, K. Nguyen, P. T. Wilson, M. Jackoski, and B. Williams, "Human joystick: Wii-leaning to translate in large virtual environments," in *Proc. 13th ACM SIGGRAPH Int. Conf. Virtual-Reality Continuum Appl. Ind*, 2014, pp. 231–234.
[30] 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. Perception (TAP)*, vol. 8, no. 3, p. 19, 2011.
[31] P. T. Wilson, K. Nguyen, A. Harris, and B. Williams, "Walking in place using the microsoft Kinect to explore a large ve," in *Proc. 13th ACM SIGGRAPH Int. Conf. Virtual-Reality Continuum Appl. Ind. ACM*, 2014, pp. 27–33.
[32] Y. Zheng, M. McCaleb, C. Strachan, and B. Williams, "Exploring a virtual environment by walking in place using the microsoft Kinect," in *Proc. ACM Symp. Appl. Perception*, 2012, p. 131.
[33] B. Williams, M. McCaleb, C. Strachan, and Y. Zheng, "Torso versus gaze direction to navigate a VE by walking in place," in *Proc. ACM Symp. Appl. Perception*, 2013, pp. 67–70.
[34] V. K. Ketoma, P. Schäfer, and G. Meixner, "Development and evaluation of a virtual reality grocery shopping application using a multi-Kinect walking-in-place approach," in *Proc. Int. Conf. Intell. Hum. Syst. Integr.* Springer, 2018, pp. 368–374. [Online]. Available: https://link.springer.com/ chapter/10.1007/978-3-319-73888-8_57
[35] J.-S. Kim, D. Gračanin, and F. Quek, "Sensor-fusion walking-in-place interaction technique using mobile devices," in *Proc. IEEE Virtual Reality Workshops (VRW)*, Mar. 2012, pp. 39–42.
[36] VRMersive. (2016). *VR-Step*. Accessed: Jul. 15, 2018. [Online]. Available: https://assetstore.unity.com/packages/tools/input-management/vr-step-60450
[37] T. Pfeiffer, A. Schmidt, and P. Renner, "Detecting movement patterns from inertial data of a mobile head-mounted-display for navigation via walking-in-place," in *Proc. IEEE Virtual Reality (VR)*, Mar. 2016, pp. 263–264.
[38] S. Tregillus, M. Al Zayer, and E. Folmer, "Handsfree omnidirectional VR navigation using head tilt," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2017, pp. 4063–4068.
[39] J. Lee, G. J. Kim, S. C. Ahn, and J.-I. Hwang, "MIP-VR: An omnidirectional navigation and jumping method for VR shooting game using IMU," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–3.
[40] J. Lee, S. Ahn, and J.-I. Hwang, "A walking-in-place method for virtual reality using position and orientation tracking," *Sensors*, vol. 18, no. 9, p. 2832, 2018.
[41] N. Zhao, "Full-featured pedometer design realized with 3-axis digital accelerometer," *Analog Dialogue*, vol. 44, no. 6, pp. 1–5, 2010.
[42] pabong. (2014). *Environment 3D Model*. Accessed: Sep. 12, 2018. [Online]. Available: https://free3d.com/3d-model/environment-23742.html
[43] M. Slater, M. Usoh, and A. Steed, "Depth of presence in virtual environments," *Presence, Teleoperators Virtual Environ.*, vol. 3, no. 2, pp. 130–144, 1994.
[44] N. C. Nilsson, S. Serafin, M. H. Laursen, K. S. Pedersen, E. Sikström, and R. Nordahl, "Tapping-In-Place: Increasing the naturalness of immersive walking-in-place locomotion through novel gestural input," in *Proc. IEEE Symp. 3D User Interfaces (3DUI)*, Mar. 2013, pp. 31–38.



YANGYI ANG received the master's degree in computer science from Universiti Putra Malaysia (UPM), Malaysia, in 2015, where she is currently pursuing the Ph.D. degree in multimedia computing. Her research interests include virtual reality, augmented reality, and mobile application development.

PUTERI SUHAIZA SULAIMAN received the degree in science (computer) and the M.Sc. degree in computer science from Universiti Teknologi Malaysia (UTM), and the Ph.D. degree in computer graphics from Universiti Putra Malaysia (UPM). She is currently a Senior Lecturer with UPM. She has more than 15 years of experience in the field of computer graphics, computer vision, information visualization and advanced interaction. Her current research explores the realm of virtual and mixed reality, focusing on the interaction and visualization elucidation. She has been involved in designing and developing computer games for children and people with special needs.

RAHMITA WIRZA O. K. RAHMAT received the B.Sc. and M.Sc. degrees in science mathematics from University Science Malaysia, in 1989 and 1994, respectively, and the Ph.D. degree in computer assisted engineering from University of Leeds, U.K. From 1989 to 1990, she worked as a Research Assistance with the Department of Physics, University Science Malaysia, experimenting on Ozone layer measurement at the Equatorial region, before working as a Tutor in Universiti Putra Malaysia. She is currently a Professor with the Faculty of Computer Science and Information Technology, UPM. Among her focus research areas are computer graphics and applications, computer assisted surgery, and computational geometry. Until now she has lead ten research grants, published her work in nearly 100 journals, 80 proceedings, five book chapters, and two books. 28 of her research students have graduated where she as the main supervisor for 13 of them. She has supervised at least 40 M.Sc. (without thesis) project and at least 100 B.Sc. final year project. She also managed to patented six and numbers copyrighted of her research work. Few years back, she has organized eight conferences and two workshops. Few of her research work have successfully worn medals in national and UPM research exhibition.

NORIS MOHD NOROWI received the degree in computer science (multimedia) and the master's degree in multimedia systems from Universiti Putra Malaysia (UPM), and the Ph.D. degree in artificial intelligence in computer music from the University of Plymouth, U.K. She is currently an Academic Staff with the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia. Her research interests include human–computer interaction, sound synthesis, sound cognition, and interactive technologies.

• • •