Smooth transitioning between two walking metaphors for Virtual Reality applications

I. Salvetti and A. Ríos and N.Pelechano

ViRVIG, Universitat Politècnica de Catalunya

Abstract

Virtual navigation should be as similar as possible to how we move in the real world, however the limitations of hardware and physical space make this a challenging problem. Tracking natural walk is only feasible when the dimensions of the virtual environment match those of the real world. The problem of most navigation techniques is that they produce motion sickness because the optical flow observed does not match the vestibular and proprioceptive information that appears during real physical movement. Walk in place is a technique that can successfully reduce motion sickness without losing presence in the virtual environment. It is suitable for navigating in a very large virtual environment but it is not usually needed in small virtual spaces. Most current work focuses on one specific navigation metaphor, however in our experience we have observed that if users are given the possibility to use walk in place for large distances, they tend to switch to normal walk when they are in a confined virtual area (such as a small room). Therefore, in this paper we present our ongoing work to seamlessly switch between two navigation metaphors based on leg and head tracking to achieve a more intuitive and natural virtual navigation.

CCS Concepts
•*Software and its engineering* → *Virtual worlds training simulations;* •*Human-centered computing* → *Virtual reality;*

1. Introduction

Walking normally is by far the most natural and intuitive way of moving in the virtual world. However natural walk is only possible when the size of the virtual room matches the size of the physical space (and tracked area). Sometimes, if the difference between the real and physical space is not too large, it is possible to fool the user by altering the ratio between real and physical displacement [PSAB08].

A successful way to allow the user to move in an almost natural manner is known as walk-in-place (WIP). WIP requires the user to move the legs up and down as if he was walking but without changing location. WIP techniques either detect discrete steps or equip the user with sensors to track the continuous movement of the feet. This second case results in smoother velocity, since it computes the camera velocity from feet vertical movement.

Head Mounted Displays with 6DOF allow the user to move naturally within a small area. Typically, if the virtual environment is larger, a hand held controller is employed to move further distances.The user ends up switching between the controller to move large distances and then walks normally within a small area to inspect its surroundings.

In this work, we were interested in replacing the controller by a WIP technique using two HTC trackers located at the user's feet and the VIVE HMD. We observed that when users are familiar with the HTC VIVE, they would use the up/down leg movement for large distances but they would automatically start walking normally when entering a small room. Since WIP only tracks the displacement in the vertical axis, this would fail to move the camera accordingly when the foot would perform a large step over the horizontal plane without much change of the vertical value. This brought up the need to combine navigation metaphors to switch seamlessly between walk-in-place (WIP) and natural walk (NW) based on the user behavior.

2. Related work

Artificial Locomotion Techniques (ALT) are used to navigate a virtual environment without the user physically moving with his own body. Popular ALT's include vehicle movement and full locomotion with hand held devices. Both techniques allow the user to easily navigate through a virtual environment, but they produce motion sickness due to the difference between what the user sees and the proprioceptive and vestibular feedback [RO16]. Teleportation avoids motion sickness by simulating discrete transitions between locations, however, it tends to produce disorientation and an unwanted loss of presence due to the lack of optical cues [LPAZF18].

The best solutions to avoid motion sickness involve natural

movement in the real world. The WIP technique was first described by [SSU95]. They compared the "walking on the spot" method with the traditional hand-held pointing device navigation and showed that the first one improved the sense of presence. The Gaiter [TDS99] used sensors attached to the knees, waist, head and feet. It allowed users to walk straight and turn in a natural way over. Wilson et al. used Myo arm bands to run a comparison of two WIP techniques based on arm swinging and feet movement against normal walk [WKMW16].

Marchal et al. presented a device that would allow to use the body as a joystick, and movement was performed by leaning in a specific direction [MPL11]. Other have combined leaning techniques with custom-designed foot haptics [KMT*16] and observed that adding walking related cues (such as footstep sounds or simulating bobbing head-motions) could also enhance presence and reduce motion sickness.

WIP has also been used for mobile VR applications using inertial sensors [TF16]. The Legomotion [BTF17] was implemented using a wireless Android Tablet that integrates motion tracking. The system is designed to allow the users to switch between normal walking and walk-in-place through a real-time step detection algorithm.The steering direction is given by the head orientation, therefore it cannot handle looking around while walking. Also step detectors based on head movement provide exclusively a discrete input which is difficult to map to a continuous velocity function in a smooth and natural manner. There are several methods of redirected walking (to allow users to walk endlessly in a VE): scaling user motion and head rotation for the virtual camera [SBJ*10,RSS*02], warping the virtual scene [DXMZ*17] and take advantage of blinks and eye saccades to modify what the user sees while interacting with the surroundings [SPW*18].

3. Our system

The goal of our system is to identify when the user is switching between a WIP movement and a normal walk. The key element to differentiate those locomotion behaviors can be done by gathering information from the positional tracking of the feet or HMD. From our initial experiments, we found that the displacement of the HMD along the horizontal plane, provides enough information to differentiate between both walking metaphors. Our WIP method does not compute camera movement exclusively from the feet movement, but instead it obtains the final camera movement from the combination of the HMD displacement and a velocity vector computed from the feet tracking difference in position and orientation. This provides not only smooth transitions between walking metaphors, but it also allows the user to look around or even bend down while performing a WIP, and obtain correct visual cues from the virtual environment. This method also allows to keep the slight bouncing left/right and up/down of the head that appears while we walk.

We tested the system in a 1.5x2m indoor lab area. Position tracking was performed using an HTC VIVE immersive virtual display and two VIVE trackers were strapped to each foot. The complete setup can be seen in Figure 1 The trackers are rotated such that the forward direction of the tracker corresponds to the front of the foot. The surroundings can be inspected by moving the head and Unity Game Engine was used to render the environment and animations.

3.1. Navigation algorithm

The algorithm has several steps. The first one consists in differentiating between the two walking metaphors, and the second one is to find the right mapping function between feet movement and camera movement for the WIP. In order to determine the walking metaphor, we can use the fact that during normal walk, the user performs larger displacements over the floor, meaning that the difference in position over the horizontal plane (XZ) between consecutive frames will be larger than a given threshold δ.

Therefore, considering that p_t^H is the projection over the horizontal plane of the position of the HMD at time t and ε a small window to avoid constant flickering between the two walking states, we have:

$$S = \begin{cases} NW & \text{if } (p_t^H - p_{t-1}^H) > (\delta + \varepsilon) \\ WIP & \text{if } (p_t^H - p_{t-1}^H) < (\delta - \varepsilon) \end{cases} \quad (1)$$

During NW, the camera movement in both position and orientation is driven by the HMD movement. The complicated part is to compute camera movement for WIP, which implies finding a function that maps feet movement into walk velocity. Our method computes the steering direction of the camera based on the orientation of the feet trackers. So that the WIP velocity orientation is computed as the unit vector:

$$\hat{v} = \hat{v}^L + \hat{v}^R \quad (2)$$

Where $\hat{v}^L$ is the unit vector of the projection of the left foot orientation onto the XZ plane, and $\hat{v}^R$ is computed similarly for the right foot. This provides a natural steering direction based on feet orientation as it happens in the real world when walking forward. Note that WIP only simulates walking forward. The user can easily switch to stepping sideways or backwards as he would do in the real world, since the model would automatically detect this as normal walk, and thus change to NW state.

The magnitude of the walk velocity due to feet movement at instant t, v_t^{walk}, during WIP is computed with the sum of the magnitudes of velocities of the right, $\vec{v}_t^R$, and left, $\vec{v}_t^L$, feet trackers after applying filtering high frequencies to reduce noise from sensors.

$$v_t^{walk} = \left\| \vec{v}_t^R \right\| + \left\| \vec{v}_t^L \right\| \quad (3)$$

Computing the feet velocity at every frame as indicated above will produce a jerky walk velocity as we can see in Figure 2 (middle), with velocity dropping to zero magnitude when both feet are on the floor between steps, and also when one feet is on the floor while the other is changing from moving up to moving down. This velocity function does not correctly match the velocity of our head movement while walking (we do not abruptly stop between steps as our body keeps leaning forward while both feet are on the floor, and we also do not slow down in the middle of a step). In order to obtain a mapping function that better reproduces this behavior, our solution consists on computing the final walk velocity by doing a weighted sum that includes the last n velocities (we have empirically found that $n = 5$ is a good value from a perceptual perspective). Therefore, the final magnitude of the walk velocity is com-

(a)   **(b)**   **(c)**   **(d)**
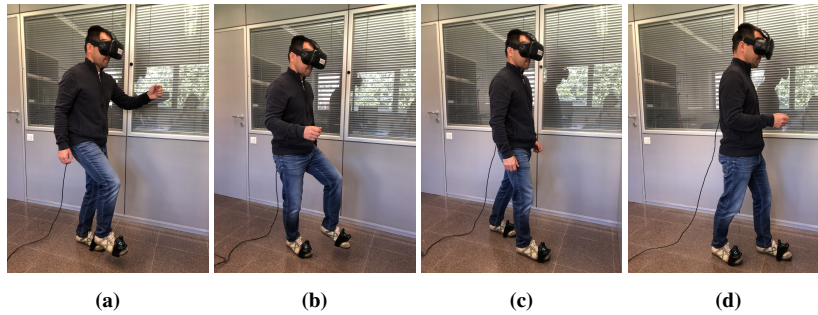
Figure 1: *User seamlessly switching between WIP (a,b) and Normal Walk (c,d).*

puted as:

$$v_t^{fwalk} = \alpha v_t^{walk} + \frac{\gamma}{n} \sum_{i=1}^{n} v_{t-i}^{walk} \tag{4}$$

where α and γ are parameters such that $\alpha + \gamma = 1.0$. Our empirically found best values are $\alpha = 0.25$ and $\gamma = 0.75$. Based on those values, we can decide how much influence we give to the current velocity extracted from the feet trackers, and how much to the previous n walk velocities. The final camera velocity is computed by summing the velocity extracted from the HMD positional trackers, $\vec{v}^H$, with the magnitude of the walk velocity computed above and the orientation given by the feet trackers.

$$\vec{v}^{cam} = v^{fwalk} \cdot \hat{v}^{walk} + \vec{v}^H \tag{5}$$
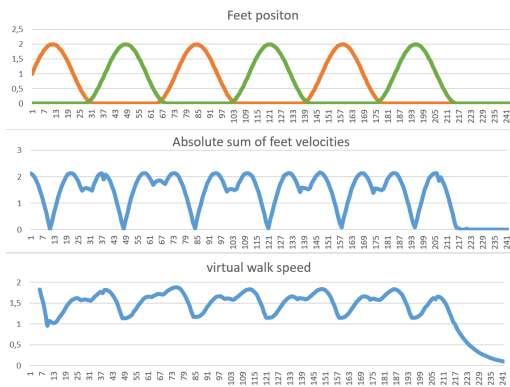
Figure 2: *Feet position during a walk in place (top), absolute sum of feet velocity in the Y axis computed by equation 3 (middle). Result of our mapping function from feet movement to navigation computed by equation 5 (bottom).*

We can see in Figure 2 (bottom), that this mapping function is smoother and will provide a closer match with our normal walk velocity function. Note that for the camera movement the final walk magnitude is used, but the direction is given by the current orientation of the trackers, so that turns are applied straight away.

If current v^{walk} results in 0, a diminishing fraction of the previous walk speed will be used instead for a small amount of frames. This keeps the walking fluid, as users tend to spend a few frames without moving between steps during the simulated walking. An important side effect of this induced fluid velocity is that it adds a small delay when the user decides to stop walking. Therefore it is important to find a good trade-off between obtaining a fluid velocity during movement, but reducing the latency when stopping.

As we have seen, the displacement of the headset is always being added to the character, even during the walk-in-place, which provides freedom of consistent head movements even while performing a WIP.

A virtual play-area is rendered around the player when the user gets close to the limits of the physical area. To maintain correct relative position between the player and play-area, the play-area receives only the displacement generated by the walk-in-place, while the player receives both the walk-in-place and headset displacements. Once the player approaches the edges of the play area, a rectangle is rendered to indicate a possible collision (see Figure3).

4. Results

The train station model introduced in [RMP18] was used to test our system. The user can move freely over a wide space from the main entrance to the platforms. An area with shops and cafeterias were placed to allow the users to walk inside and move within a small area (Figure 3). Over a hundred avatars walk around simulating an everyday situation to make the experience more realistic.

(a)   **(b)**   **(c)**
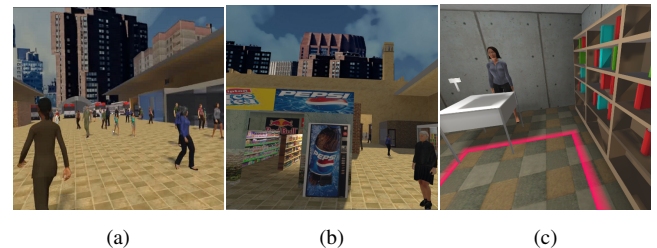
Figure 3: *Scenario of the train station, and rendered rectangle.*

We have run preliminary experiments with this scenario, and verify the correct and natural switching between the two walking

methods (WIP-NW). Figure 4 shows an example of a real trajectory followed by a user, with the state changes between WIP and NW. The graph shows the head position as the users navigates the virtual train station, and colors represent the state.
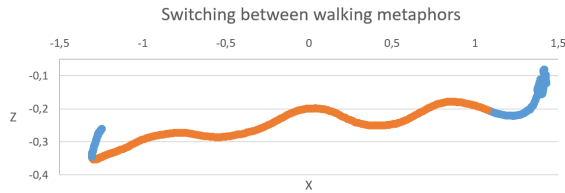
Figure 4: *State changes and real head position.*

The algorithm successfully changed states and thus velocity mapping function. Since head position is always considered to update virtual navigation (even during WIP), our method respects the natural head movement during walking which reduces motion sickness. It also allows for free head movement while moving in WIP. There is however one important element from this graph that needs to be further studied, and it is the fact that during WIP we can still observe that the user physically changes location without noticing. This means that, even though the user is trying to stay in the spot, there is a natural displacement that appears through all our initial experiments. Therefore it is important to study how to avoid the user from moving out of the tracked area. At this stage we are simply rendering a rectangle over the floor, which identifies the free movement area. Although this technique is easy to use, we would like to explore immersive techniques to achieve this goal, without the user noticing.

If we observe the head velocity as the user walks normally, we can see that it is not a uniform velocity, but it follows an ondulatory curve with the peaks happening right after switching feet. Figure 5 shows the feet height during a normal walk (note that since the tracker is above the foot, its resting position is around 0.03). We can observe that the height increases as we lift the foot, decreases a bit during the swing phase and increases just before landing the foot on the floor. Our mapping function also results in an ondulatory behavior with the peak happening a bit after the user changes standing feet during WIP.
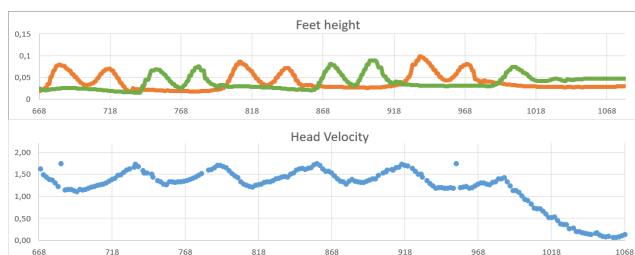
Figure 5: *Feet height during NW and head velocity from the HMD.*

5. Conclusions and Future Work

Our preliminary results show a seamless and natural transition between WIP and NW. Our system provides a velocity mapping function that closely resembles natural walk. We have observed through a pilot study that the system successfully reduces motion sickness and enhances immersion. Our next step will be to run a study with a large number of participants to check the usability of the method and evaluate efficiency, reliability, learnability and immersion. We also want to study how to adjust the parameters α and γ in equation 4 of our system to each user, meaning that the values for considering the time elapsed for stopping and the speed relationship between feet movement and navigation velocity should probably be user-specific and easily adjusted in a short training process.

One of the limitations of the system, is that the trackers must be visible from the HTC base stations, therefore it is important to have an empty space of the required dimensions, dedicated exclusively to our navigation system.

References

[BTF17] BHANDARI J., TREGILLUS S., FOLMER E.: Legomotion: Scalable walking-based virtual locomotion. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology* (New York, NY, USA, 2017), VRST '17, ACM, pp. 18:1–18:8. 2

[DXMZ*17] DONG Z., XIAO-MING F., ZHANG C., WU K., LIU L.: Smooth assembled mappings for large-scale real walking. *ACM Transactions on Graphics 36* (11 2017), 1–13. 2

[KMT*16] KRUIJFF E., MARQUARDT A., TREPKOWSKI C., LINDEMAN R. W., HINKENJANN A., MAIERO J., RIECKE B. E.: On your feet!: Enhancing vection in leaning-based interfaces through multisensory stimuli. In *Proceedings of the 2016 Symposium on Spatial User Interaction* (New York, NY, USA, 2016), SUI '16, ACM, pp. 149–158. 2

[LPAZF18] LIU J., PAREKH H., AL-ZAYER M., FOLMER E.: Increasing walking in vr using redirected teleportation. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2018), UIST '18, ACM, pp. 521–529. 1

[MPL11] MARCHAL M., PETTRÉ J., LÉCUYER A.: Joyman: A human-scale joystick for navigating in virtual worlds. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)* (2011), IEEE, pp. 19–26. 2

[PSAB08] PELECHANO N., STOCKER C., ALLBECK J., BADLER N.: Being a part of the crowd: towards validating vr crowds using presence. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1* (2008), International Foundation for Autonomous Agents and Multiagent Systems, pp. 136–142. 1

[RMP18] RIOS A., MATEU D., PELECHANO N.: Follower behavior in a virtual environment. *Virtual Humans and Crowds in Immersive Environments (VHCIE) . March 18. Reutlingen (Germany)* (2018). 3

[RO16] REBENITSCH L., OWEN C.: Review on cybersickness in applications and visual displays. *Virtual Reality 20*, 2 (2016), 101–125. 1

[RSS*02] RAZZAQUE S., SWAPP D., SLATER M., WHITTON M. C., STEED A.: Redirected walking in place. In *Proceedings of the Workshop on Virtual Environments 2002* (Aire-la-Ville, Switzerland, Switzerland, 2002), EGVE '02, Eurographics Association, pp. 123–130. 2

[SBJ*10] STEINICKE F., BRUDER G., JERALD J., FRENZ H., LAPPE M.: Estimation of detection thresholds for redirected walking techniques. *IEEE Transactions on Visualization and Computer Graphics 16*, 1 (Jan 2010), 17–27. 2

[SPW*18] SUN Q., PATNEY A., WEI L.-Y., SHAPIRA O., LU J., ASENTE P., ZHU S., MCGUIRE M., LUEBKE D., KAUFMAN A.: Towards virtual reality infinite walking: Dynamic saccadic redirection. *ACM Trans. Graph. 37*, 4 (July 2018), 67:1–67:13. 2

[SSU95] SLATER M., STEED A., USOH M.: The virtual treadmill: A naturalistic metaphor for navigation in immersive virtual environments. *Virtual Environments '95: Selected papers of the Eurographics Workshops in Barcelona, Spain, 1993, and Monte Carlo, Monaco, 1995* (01 1995), 135 –148. 2

[TDS99] TEMPLEMAN J., DENBROOK P., SIBERT L.: Virtual locomotion: Walking in place through virtual environments. *Presence 8* (12 1999), 598–617. 2

[TF16] TREGILLUS S., FOLMER E.: Vr-step: Walking-in-place using inertial sensing for hands free navigation in mobile vr environments. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), CHI '16, ACM, pp. 1250–1255. 2

[WKMW16] WILSON P. T., KALESCKY W., MACLAUGHLIN A., WILLIAMS B.: Vr locomotion: walking> walking in place> arm swinging. In *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry-Volume 1* (2016), ACM, pp. 243–249. 2