Strolling in Room-Scale VR: Hex-Core-MK1 Omnidirectional Treadmill

Ziyao Wang, Chiyi Liu, Jialiang Chen, Yao Yao, Dazheng Fang, Zhiyi Shi, Rui Yan, Yiye Wang, KanJian Zhang, Hai Wang, and Haikun Wei

Abstract—The natural locomotion interface is critical to the development of many VR applications. For household VR applications, there are two basic requirements: natural immersive experience and minimized space occupation. The existing locomotion strategies generally do not simultaneously satisfy these two requirements well. This article presents a novel omnidirectional treadmill (ODT) system named Hex-Core-MK1 (HCMK1). By implementing two kinds of mirror-symmetrical spiral rollers to generate the omnidirectional velocity field, this proposed system is capable of providing real walking experiences with a full-degree of freedom in an area as small as 1.76 m², while delivering great advantages over several existing ODT systems in terms of weight, volume, latency and dynamic performance. Compared with the sizes of Infinadeck and HCP, the two best motor-driven ODTs so far, the 8 cm height of HCMK1 is only 20% of Infinadeck and 50% of HCP. In addition, HCMK1 is a lightweight device weighing only 110 kg, which provides possibilities for further expanding VR scenarios, such as terrain simulation. The system latency of HCMK1 is only 9 ms. The experiments show that HCMK1 can deliver a starting acceleration of 16.00 m/s² and a braking acceleration of 30.00 m/s².

Index Terms—Omnidirectional treadmill, locomotion devices, locomotion interfaces, room-scale VR

1 INTRODUCTION

With the development of virtual reality technology in recent years, the science fiction world in movies has gradually become a reality [47]. The virtual reality (VR) industry pursues the immersive user experience (UX) [6], [25], [41]. The natural locomotion interface (NLI) [18], [22], [51], [56] is one of the subdivisions of the user experience in VR. Currently, limited by natural locomotion, most of the VR scenes are restricted to a fixed location or a small-scale local area, also compromising the richness of content. Problems with NLI have become a bottleneck for the development of many VR applications. The natural user experience of locomotion is a comprehensive goal, where nuances in walking style, degree of freedom, delay, and continuity of the movement will affect the user’s immersion and make them feel uncomfortable. In addition, household VR applications often have to achieve natural locomotion in VR within a small physical space when trying to immerse users in the VR world completely. There is an urgent need for an effective solution that can meet both the requirements of physical space occupancy and the comprehensive goal of natural UX when NLI problems for room-scale VR are taken into consideration.

The NLI problem has been extensively studied. Several strategies based on different principles have been proposed, such as the controller-based methods, the walking in large space methods, the walking in place methods, the redirected walking methods and the omnidirectional treadmill (ODT) methods [2]. Although these different strategies can realize movement in VR, these solutions have difficulty meeting both the natural UX and requirements of space occupation in the face of room-scale NLI problems. Each of these methods has its own pros and limitations. For instance, controller-based and WiP methods are both lower-cost and do not require much space, but offer some disadvantages on UX (e.g., WiP usually has a low degree of freedom and high delay); while RDW and walking in large place methods are natural in terms of UX, but require large space and will be limited by physical boundaries; ODT, as an auxiliary device, can effectively balance UX and space requirements through various types of unique design, despite the need for additional mechanical structures and high hardware costs.

Specifically, for controller-based methods, a simple way is to use the controller such as the touchpad, to control the movement in the virtual environment (VE), just like the classic control method “WASD” or joystick in computer games.

1. https://en.wikipedia.org/wiki/WASD

Authorized licensed use limited to: University of Cambridge. Downloaded on December 23, 2023 at 10:27:12 UTC from IEEE Xplore. Restrictions apply.
It can lead to serious 3D motion sickness [28], [29], [30] due to the conflict between visual perception and inner ear perception [31]. Another controller-based method is teleportation [29]. The limitation includes the lack of awareness of the intermediate path, which will affect the body’s positioning of the current location and lead to getting lost. Besides, the user may have to repeat additional operations to arrive at a proper position when they are reaching for an object.

The Walk-in-place (WiP) methods replace natural walking with some specified actions, such as arm swinging [40], [64], jogging [17], [34], [38], etc. Since the WiP methods have the process of simulating walking, their UX is better than that of the handheld controller-based methods [17], [55], [64]. Generally, each proxy action can represent only one movement direction. Basic movements like forward, backward and lateral movements (right and left) [59] will need 4 proxy actions. Proxy actions are usually accompanied by a reduction in the degree of freedom (DOF) during movement. For instance, most WiP methods apply different proxy actions based on the direction of the head or torso [15], [17], [34], [54], [55], [62], which makes it difficult for the user to continuously walk in the original straight line when turning or looking around, i.e., the movement direction is restricted by the head or torso. In ArmSwing [40], although such a restriction is eliminated by the arm swing direction, the decrease in DOF is reflected by the fact that the regular hand interaction conflicts with the arm swing. It is difficult to perform other forms of interactions that require hand gestures while navigating. In addition, the system latency in starting and stopping is a challenge for the WiP methods [15], [53], i.e., how long it takes the VR locomotion to start or stop after the user takes a step or stops their steps. Latency in WiP is the result of adding a low-frequency filter or increasing the detection threshold in the data process, which aims to improve detection accuracy and reduce false triggers [15], [62]. For example, the first proposed WiP method [48] does not start the VR locomotion until 4 consecutive steps have been taken. VR-STEP [55] averages every 5 samples to filter the signal, and the latency is about \( 100 \text{ms} \) to \( 200 \text{ms} \). LLCM-WIP [15] chooses a cut-off frequency of \( 5Hz \), which adds about \( 100 \text{ms} \) of latency to both starting and stopping. After applying the offsetting operation, the final starting latency reaches \( 138 \text{ms} \) and the stopping latency of \( 96 \text{ms} \). Generally, the WiP methods seek a compromise between false triggers and latency, which are inevitable and difficult to eliminate.

For the walking in large space methods and redirected walking methods, moving in VE with our own legs is the most natural way of locomotion interfaces [18], [22], [51]. Based on low latency tracking technology, the walking in large space method [58] is an intuitive solution. Generally, the navigation space in VE is equal to the physical space. The limitation of this method is that the real-world space will limit the virtual-world space. If users move close to the boundary of the physical space, they cannot take more steps. In order to make full use of the limited space, space compression technology, such as the redirected walking (RDW) approach, has been extensively studied in recent years. Based on the human body’s insensitivity to slight rotation and translation, the RDW approach can leverage visual dominance to subtly manipulate the user’s physical path [5], [7]. When the user walks in a straight line in VE, the RDW approach guides the user to walk along a circle in the physical space. This is an effective strategy for compressing the limitless virtual space into a limited physical space [32], [38], [44]. Generally, the RDW approach uses unperceived curvature gains to manipulate the user’s path [50]. It has been shown that the minimum demand for space is about \( 7m \) in radius or \( 200m^2 \) [7]. To further reduce the space required, Telewalk [43] uses perceivable gains deliberately, highly reducing the space requirement to \( 3m \times 3m \). Since the Telewalk approach uses very high RDW gains, which can lead to motion sickness symptoms [43], it sacrifices UX for space reduction. In addition, a common problem faced by RDW is that the moving direction needs to be reset when approaching the physical boundary. Freeze-Backup, Freeze-Turn and 2:1-Turn [63] are still the mainstream solutions, and they interrupt the continuity of movement.

Strategies with auxiliary equipment usually have a much better performance in terms of the UX and space requirements. Omnidirectional treadmills (ODT) are a representative type of auxiliary equipment. The main idea is to keep the users’ bodies stationary when walking [2]. Compared with RDW and walking in large space approaches, ODT could provide a similar walking experience, but requires much less space and does not have any boundary restrictions. Compared with the WiP methods, ODT occupies a slightly larger area but eliminates the limitations of the WiP strategy. For example, the VR locomotion of ODT is independent of the view and torso direction, and the user can move in any direction without constraints. Hence, ODT achieves a higher degree of freedom than the WiP methods. In addition, ODT usually doesn’t need to consider the false trigger problem associated with the WiP methods, so the starting and stopping latency is much lower. In general, as an auxiliary device, the ODT is capable of simultaneously ensuring UX and space occupation. It has been demonstrated to be an effective solution to the NLI problem in room-scale VR [2].

Currently, a lot of different ODT design schemes have been proposed, and they can preliminarily achieve the basic functions. However, these design schemes have different limitations, such as being laborious to use, or having a dead zone, low degree of freedom, or bulkiness. These limitations are inherent shortcomings of the designs and are difficult to eliminate through optimization of the manufacturing process.

To overcome these aforementioned limitations, and simultaneously meet the requirements of user experience and space occupation for room-scale NLI problem, this paper proposes a novel ODT design scheme, i.e., the 45-degree wheel-based scheme with the spiral rollers as the carrier, which has significant advantages in terms of UX and volume. Based on this design scheme, we created the HexCore-MK1 (HCMK1) system shown in Fig. 1. On the premise of providing the most natural locomotion experience, HCMK1 is a miniaturized device that only occupies \( 1.76m^2 \) area and \( 8.0cm \) height. The small volume indicates that it is suitable for room-scale VR. The \( 110kg \) weight ensures high dynamic performance and provides capabilities for many VR applications like terrain simulations. In addition, the system latency of HCMK1 is only \( 9ms \), and the end-to-end (ETE) latency [35] is \( 31.22ms \) to \( 49.4ms \). Low latency ensures
the VR locomotion can be consistent with the user's actual intention in real-time. This paper further analyzes the main factors affecting the UX, including the OVF (omnidirectional velocity field) working delay (the delay after the user stops walking until the platform stops working) and the height of the user's center of gravity, which may guide the design of the controller to further improve the UX.

The main contributions of this paper are as follows:

- Proposing a novel design scheme of driven-based ODT.
- Developing the HCMK1 system based on the proposed scheme, which has state-of-the-art performance in terms of volume, weight, latency and dynamic performance.
- Analyzing the main factors affecting the UX based on the survey and the experimental data.

The remainder of the paper is organized as follows. Section 2 describes the related work. Section 3 presents the proposed novel design scheme of the HCMK1 system. Section 4 compares the proposed HCMK1 system with other driven-based ODTs. Section 5 presents an experimental study, and Section 6 presents the limitation of the proposed HCMK1 system, respectively. Finally, Section 7 summarizes the results and highlights future work. In addition, several videos about an application demo and part of the experiments are attached in the supplementary material, available online.

2 RELATED WORK

The main feature of ODT is to keep the user stationary in physical space when the user is walking. A simple method is sliding in place, i.e., the low-friction surface scheme [3], [10], [19]. Generally, the user's waist is tied to the machine. The user needs to wear a pair of roller shoes or stand on a special surface with suitable low-friction, such as a polytetrafluoroethylene (PTFE) surface or a ball-bearing surface. When walking forward, the user's feet will slide on the surface with the assistance of their waist. The commercial products based on this principle include Kat Walk, Virtuix Omni, Cyberith Virtualizer, etc. The main problem of this strategy is that overcoming the friction is too laborious; imagine the user needs to moonwalk on the surface like Jackson all the time. Users can slide forward laboriously but it is hard to slide backward or sideways [10]. The walking tutorial of Kat Walk shows the user needs to combine the WiP proxy actions to realize the backward and sideways. Besides, since the gait action is abnormal, it is hard to track the gait distance accurately. Kat Walk and Virtuix Omni use the inertial measurement unit (IMU) to count the number of steps and indirectly calculate the movement distance, which further reduces the accuracy of gait matching or even appears the incorrectly estimated gaits.

Like the traditional treadmill in the gym, if ODT can automatically carry users back to the center, the UX will be much better. This kind of ODT is also called driven-based ODT. CirculaFloor [23] uses 4 independent, autonomous circulating robots to carry the user back. The user always stands on 2 robots and another 2 robots move to the position of the user's next step. String Walker [24] uses several strings to pull the shoes back to the center. CyberCarpet [13], [46] and StriderVR are two similar ODTs. They all lay a layer of steel balls on a traditional treadmill. The main difference is that when the user turns, CyberCarpet rotates the traditional treadmill in the same direction to provide velocity in a proper direction, but StriderVR rotates the steel-ball layer in the inverse direction to eliminate the user's rotation. The main problems of these ODTs are the low dynamic performance and the low degree of freedom.

The fast-responding and continuous omnidirectional velocity field (OVF) is the key to driven-based ODTs [61]. Omnideck [8] can generate a fixed inwardly contracting OVF based on a number of inwardly rotating rollers. The fixed OVF means the user can only walk on the outer ring of the surface, which leads to low area utilization and large area occupation. An ideal OVF should be parallel, continuous, and able to respond to direction changes quickly. A traditional scheme is the belt-based ODT [11], which can be simply understood as a big treadmill (x-axis) carrying several small treadmills (y-axis). This scheme has been applied in several previous works, such as the Torus[21], Cyberwalk [45], [49] and F-ODT [33], [42]. The commercial product Infinadeck [36] is also based on this scheme. This scheme could provide a real walking experience with full DOF. The only drawback is that it is hard to miniaturize due to its double-layer structure. As a
commercial product, Infinadeck has been optimized for several years and still has 40 cm height and 225 kg weight. The large volume limits its application areas. Another scheme is the 45-degree wheel-based scheme [61]. Hex-Core-Prototype (HCP) is a much thinner and smaller ODT based on this scheme. The main components of HCP are the mirror-symmetrical chain and the small wheels arranged at 45 degrees on the chain. It can generate the ideal OVF based on the principle of the decomposition and composition of the velocity. HCP has reduced the height to 16 cm and the weight to 150 kg, which proves that the 45-degree wheel-based scheme has more competitive in terms of miniaturization. However, the chain usually stretches soon, which leads to the tooth-jumping problem. Taking the chains as carrier still is a double-layer structure, therefore, the volume could be further reduced.

3 SYSTEM DESIGN

HCMK1 is also based on the 45-degree wheel-based scheme. The main improvement is on the carriers of the 45-degree wheels. We designed a pair of spiral rollers to support the 45-degree wheels. Compared with the mirror-symmetrical chains, the spiral rollers have a more compact structure and more stable performance. It reduces the weight of the rotation components and greatly reduces the moment of inertia through a small rotation radius of only 1.71 cm. Therefore, HCMK1 has a much better dynamic performance.

Fig. 2 presents the workflow of HCMK1. As a closed-loop control system, HCMK1 consists of 4 main parts, i.e., the omnidirectional velocity field, the positioning system, the system controller and the locomotion calculation process. This section discusses these four parts as well as the analysis of the latency.

3.1 Notations

In this paper, we use \( x, y \) to denote the scalar and vector values respectively. The subscript of a notation indicates the source of the symbol, which is usually the first letter or abbreviation. Table 1 summarizes the notations used in this paper.

3.2 Design of Omnidirectional Velocity Field

The advantages of HCMK1 in terms of UX and volume depend on the design of OVF. The left part of Fig. 3 shows the main components: the mirror-symmetrical spiral rollers. The roller body is embedded with a number of small wheels, which are arranged at an angle of ± 45 degrees. Compared with the mirror-symmetrical chain in the HCP system [61], this is a more compact structure with a radius of only 1.71 cm. Therefore, the overall volume and weight of the HCMK1 system can be further reduced.

To put it simply, OVF is composed of two types of rollers, i.e., Roller1 type and Roller2 type in the left image of Fig. 3, which are alternately densely arranged. All rollers of the same type are connected by timing belts and driven by the same motor, so they have the same speed. The main difference between these two types of rollers is the direction of the wheel embedded on the surface. None of the wheels are powered and all can rotate around their axes freely. Therefore, for the external speed, the wheel can counteract the speed component in the direction of rotation, and only the axial speed component is retained. This is similar to the principle of the Mecanum wheel [20]. After such a decomposition process, the remaining speed components of these two types of rollers are perpendicular to each other. By adjusting the relative speeds of the two types of rollers, the OVF speed can be composited to move in any direction.

The following discussion is based on the right-handed coordinate system in the figure. When the Roller1 rotates around the \( z \)-axis at the angular velocity of \( \omega_{1_z} \), the linear velocity on the surface, i.e., the \( x-z \) plane, is \( v_{1_z} \). Since the 45-degree wheel rotates freely, \( v_{1_z} \) can be decomposed into two mutually perpendicular velocities, where the velocity perpendicular to the wheel’s axis will be counteracted, and another velocity along the wheel’s axis, i.e., the blue dash
Table 1: Notations in This Paper

| Notation | Meaning |
|----------|---------|
| \(w_{r1}, v_{r1}, v_{r1}^*\) | \(\circ\) Roller1’s angular velocity, linear velocity, and the retained velocity along the wheel axis. |
| \(w_{r2}, v_{r2}, v_{r2}^*\) | \(\circ\) The same meanings as above but of Roller2. |
| \(\theta_{r1}, \theta_{r2}\) | \(\circ\) The angle of the wheels on Roller1 and Roller2. |
| \(d_r\) | \(\circ\) The diameter of Roller1 and Roller2. |
| \(v_{ovf}, \theta^*, \alpha\) | \(\circ\) The velocity of OVF, with the direction of \(\theta^*\) and the amplitude of \(\alpha\). |
| \(P_{url}\) | \(\circ\) User’s position in the tracker’s coordinate system, i.e., \(X_c, Y_c, Z_c\) coordinate system. |
| \(P_{kg}, \mathbf{q}_{kg}\) | \(\circ\) Tracker’s position and spatial attitude in the global coordinate system. |
| \(P_{ug}\) | \(\circ\) User’s position in the global coordinate system. |
| \(P_{off}\) | \(\circ\) A preset point on the X-Z plane. The user will be sent to this point. |
| \(\sigma, n_{1}(t), n_{2}(t)\) | \(\circ\) The number of revolutions of servo motor1 and motor2. |
| \(\lambda\) | \(\circ\) The reduction ratio of servo motors. |
| \(D_{ovf}(t)\) | \(\circ\) The locomotion distance provided by the OVF. |
| \(D_{ovf}(t)\) | \(\circ\) The user’s locomotion distance that offset by the OVF. |
| \(D_{pe}(t), D_{ovf}(t)\) | \(\circ\) The user’s locomotion distance in the physical environment and virtual environment. |
| \(D_{rm}(t)\) | \(\circ\) The user’s local locomotion distance that has not been offset by the OVF. |
| \(T_{me}, T_{ref}, T_{mov}\) | \(\circ\) The latency of the total measurements, the positioning measurement, and the motor measurement. |
| \(T_{c}\) | \(\circ\) The latency of serial communication. |
| \(T_{s}\) | \(\circ\) The system latency of the HCMK1 system. |
| \(\beta_{1r}, \beta_{2}\) | \(\circ\) Gains of the platform OVF locomotion and the user’s local locomotion. |

line, \(v_{r1}^*\) is retained. Similarly, when the Roller2 rotates at the angular velocity of \(w_{r2}\), the surface’s linear velocity is \(v_{r2}\), only \(v_{r2}^*\), i.e., the red dash line, is retained. Based on the spiral rollers, the originally two parallel velocities, \(v_{r1}\) and \(v_{r2}\), are decomposed into two mutually perpendicular velocities, \(v_{r1}^*\) and \(v_{r2}^*\), which could be calculated by

\[
\begin{align*}
\mathbf{v}_{r1} &= \mathbf{w}_{r1} \times \left(0, \frac{d_r}{2}, 0\right); \quad \mathbf{v}_{r2} = \mathbf{w}_{r2} \times \left(0, \frac{d_r}{2}, 0\right); \\
\mathbf{v}_{r1}^* &= \left(\cos(\theta_{r1}), 0, \sin(\theta_{r1})\right); \quad \mathbf{v}_{r2}^* = \left(\cos(\theta_{r2}), 0, \sin(\theta_{r2})\right).
\end{align*}
\]

Here \(d_r = 3.42 cm\) denotes the diameter of the rollers, \(\theta_{r1} = \frac{\pi}{4}\) and \(\theta_{r2} = -\frac{\pi}{4}\) denote the angle of the wheels on Roller1 and Roller2. \(\mathbf{w}_{r1}\) and \(\mathbf{w}_{r2}\) denotes the angular velocities which point along the \(z\)-axis, the preset conditions are \(\mathbf{w}_{r1} = (0, 0, w_{r1})\) and \(\mathbf{w}_{r2} = (0, 0, w_{r2})\). 

Arranging these two kinds of rollers alternately could construct a surface shown in the middle of Fig. 3. The same type of rollers are connected in series using a row of timing belts and driven by one servo motor. Therefore, HCMK1 only needs two motors to construct the OVF, and the same type of rollers all have the same velocity.

The right part of Fig. 3 presents the composition process of OVF. \(v_{r1}^*\) and \(v_{r2}^*\) are staggered on the plane. When these two velocities act on the same object and ignore the torque caused by the non-coincidence of the acting positions, it is easy to get the composited velocity \(v_{ovf} = v_{r1}^* + v_{r2}^*\). Conversely, when a velocity with the direction of \(\theta^*\) and an amplitude of \(\alpha\) is required, i.e., \(v_{ovf} = (\alpha \cos(\theta^*), 0, \alpha \sin(\theta^*))\), these two type spiral rollers need to be set at the angular velocities

\[
\begin{align*}
\mathbf{w}_{r1} &= \left(0, 0, -\frac{2\alpha}{d_r} \cos(\theta^*) - \frac{2\alpha}{d_r} \sin(\theta^*) \tan(\theta_{r1})\right); \\
\mathbf{w}_{r2} &= \left(0, 0, -\frac{2\alpha}{d_r} \cos(\theta^*) - \frac{2\alpha}{d_r} \sin(\theta^*) \tan(\theta_{r2})\right).
\end{align*}
\]
3.3 Positioning System

High accuracy and low-latency positioning signals are important to control performance. The HCMK1 system uses the Vive Tracker\(^5\) to track the user’s position. Vive Tracker is based on the SteamVR tracking technology, which has a latency of fewer than 5 milliseconds and millimeter-scale positioning accuracy [39].

Since the active area of the platform is about 1.2m in diameter, it is essential to use an accurate method to measure the user’s position. As shown in Fig. 4, we chose the center point to represent the whole body. Because the balance of the human body is determined by the position of the center of the mass, measuring the position of the center portion is sufficient to represent the position of the human body. Though introducing the position information of the knees and feet could help to analyze the users’ intention, it is much more complicated, and this could be taken as an improvement of the positioning system in future work.

The tracker is worn around the waist to track the center point’s real-time position in the global coordinate system, i.e., in the XYZ-coordinate system. In this way of wearing, the center point is approximately fixed in the local coordinate system of the tracker, i.e., the \(X_t, Y_t, Z_t\)-coordinate system. We record the local position of the center point in \(X_t, Y_t, Z_t\)-coordinate system as \(\mathbf{p}_{\text{ctr}}\), which is a constant. We record the global spatial position of the tracker as \(\mathbf{p}_{\text{tg}}\) and use the quaternion to record the global spatial attitude as \(\mathbf{q}_{\text{tg}}\). Here we chose the quaternion to avoid the gimbal lock of Euler angles and to get a more stable and more efficient calculation process than rotation matrices. Based on the arithmetic rules of quaternion, the global spatial position of the center point is denoted by

\[
\mathbf{p}_{\text{tg}} = \mathbf{p}_{\text{tg}} + \text{vec}o\text{r}(\mathbf{q}_{\text{tg}} * \text{Quaternio}n(0, \mathbf{p}_{\text{ctr}}) * \mathbf{q}_{\text{tg}}^{-1}),
\]

where the \(\text{Quaternio}n(w, (x, y, z))\) means generating a quaternion, i.e., \(w + xi + yj + zk\) and \(\text{vec}o\text{r}()\) means to extract the vector part of the quaternion. \(\mathbf{p}_{\text{ctr}}\) is an approximate value, and preset as \(\mathbf{p}_{\text{ctr}} = (0, -0.1, 0)\).

3.4 Basic Controller

An ideal experience of ODTs is when the user abruptly starts walking or stops walking at a normal speed, the body always keeps stationary. This is a complex human-computer interaction problem that has a high requirement for the platform’s dynamic performance and it also needs to incorporate the kinematic model of the human body.

To preliminary verify this design scheme, we designed a basic controller. This is a simple proportional controller with an offset. When the user is at the position \(\mathbf{p}_{\text{tg}}\), set the OVF at the velocity

\[
\mathbf{v}_{\text{ovf}} = \alpha \cdot \frac{\mathbf{p}_{\text{off}}}{||\mathbf{p}_{\text{off}}||_2}.
\]

\[
\mathbf{p}_{\text{off}} = \begin{bmatrix}
1 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1
\end{bmatrix} \cdot (\mathbf{p}_{\text{ref}} - \mathbf{p}_{\text{tg}}),
\]

\\[\alpha = \begin{cases}
0 & \text{for } ||\mathbf{p}_{\text{off}}||_2 < D_{th} \\
K_p \cdot ||\mathbf{p}_{\text{off}}||_2 - D_{th} & \text{for } ||\mathbf{p}_{\text{off}}||_2 \geq D_{th}.
\end{cases}\]

Here \(||\cdot||\) denotes the euclidean norm of vectors, \(\mathbf{p}_{\text{off}}\) denotes the offset position between the projection of the user’s position on the X-Z plane and a preset reference point, i.e., \(\mathbf{p}_{\text{ctr}}\). In this basic controller, we set \(\mathbf{p}_{\text{ref}}\) as the center of the OVF. \(\alpha\) denotes the amplitude of the velocity. It is set as a piecewise function to avoid frequent adjustments when the user is close to the reference point. This is based on the control threshold \(D_{th}\). According to the experiments, setting \(D_{th} = 0.08\) is sufficient to avoid frequent adjustments. Considering the safety factor, we set \(K_p = 2\), which denotes the proportional gain. This is a fixed control strategy. The velocity of the OVF always points to the reference point, and the amplitude is determined only by the user’s position.

3.5 Locomotion Calculation

The servo motors can measure the number of revolutions, i.e., \(n_1(t)\) and \(n_2(t)\), based on a rotary encoder. The locomotion distance provided by the OVF is denoted by \(\mathbf{D}_{\text{ovf}}(t)\), which is a linear function of \(n_1(t)\) and \(n_2(t)\):

\[
\mathbf{D}_{\text{ovf}}(t) = \begin{bmatrix}
\frac{d\theta_1}{\lambda} \\
\frac{d\theta_2}{\lambda}
\end{bmatrix} \begin{bmatrix}
0 & 0 \\
0 & \frac{2\pi}{\lambda}
\end{bmatrix} \begin{bmatrix}
n_1(t) \\
n_2(t)
\end{bmatrix}.
\]

Here \(\lambda\) denotes the reduction ratio of the servo motors.

We record the cumulative locomotion distance of the user in the physical environment and in VE as \(\mathbf{D}_{\text{pe}}(t)\) and \(\mathbf{D}_{\text{ve}}(t)\). Since the user walks on the OVF, part of \(\mathbf{D}_{\text{pe}}(t)\) is the user’s locomotion distance that is offset by the OVF, which is denoted by \(\mathbf{D}_{\text{ovf}}(t)\), and another part is the remaining distance, which is denoted by \(\mathbf{D}_{\text{rm}}(t)\). Therefore,

\[
\mathbf{D}_{\text{pe}}(t) = -\mathbf{D}_{\text{ovf}}(t) + \mathbf{D}_{\text{rm}}(t).
\]

\(\mathbf{D}_{\text{ovf}}(t)\) is mainly affected by the slipping during translation on the surface. Introducing a matrix, denoted by \(\mathbf{S}_t(t)\) to represent the impact of the sliding process, i.e.,

\[
\mathbf{D}_{\text{ovf}}(t) = \int_0^t \mathbf{S}_t(x) \mathbf{v}_{\text{ovf}}(x) dx.
\]

\(\mathbf{v}_{\text{ovf}}(t)\) denotes the velocity of OVF and satisfies \(\int_0^t \mathbf{v}_{\text{ovf}}(x) dx = \mathbf{D}_{\text{ovf}}(t)\). \(\mathbf{S}_t(t)\) is a time-varying variable that
is related to various factors, such as speed, direction and the material in contact with the platform, e.g., the bottom of the user’s footwear. Since the maximum acceleration of humans in daily life is about \(1.44 \text{ms}^{-2} [52]\), even for the world 100m dash champion like Usain Bolt\(^6\) is about \(3.09 \text{ms}^{-2}\), an assumption in this paper is that the user has no slippage when walking on the OVF, and the following discussions are based on this assumption.

Algorithm 1. HCMK1 Program Algorithm

\textbf{Input:} functions: \(S_L(x), \psi(x)\); parameters: \(\theta_{t1}, \theta_{t2}, d_L, \lambda, p_{\text{ref}}, p_{\text{ref}}, K_p, D_{th}, T\)

\textbf{Output:} locomotion distance in VE: \(D_{ovf}(t)\)

1. Initialize servo motors and tracker.
2. \(t = 0\)
3. \textbf{for} each cycle period with an interval of \(T\), \textbf{do}
4. \hspace{1em} Receive \(p_{\text{q1}}(t)\) and \(q_{\text{ref}}(t)\) from tracker
5. \hspace{1em} Receive \(n_1(t)\) and \(n_2(t)\) from servo motors
6. \hspace{1em} Calculate \(p_{\text{g1}}(t)\) based on Equation (3)
7. \hspace{1em} Calculate \(D_{ovf}(t)\) based on Equations (8)–(11)
8. \hspace{1em} Map \(D_{pe}(t)\) to \(D_{ovf}(t)\) based on Equation (12)
9. \hspace{1em} Output \(D_{ovf}(t)\)
10. \hspace{1em} Calculate \(v_\tau\) based on Equations (4)–(6)
11. \hspace{1em} Calculate \(w_{\tau1}\) and \(w_{\tau2}\) based on Equation (2)
12. \hspace{1em} Send the target rotation speed \(\lambda w_{\tau1}\) to servo motor1 and \(\lambda w_{\tau2}\) to servo motor2
13. \(t = t + T\)
14. \textbf{if} Stop \textbf{then}
15. \hspace{1em} Quit

In this case, \(S_L(x)\) could be simplified as a unit matrix, i.e., \(S_L(x) \approx I_{3 \times 3}\), and

\[
D_{ovf}(t) \approx \int_0^t I_{3 \times 3} v_{\text{ref}}(x)dx = D_{ovf}(t). \tag{10}
\]

\(D_{vm}(t)\) can be measured by tracking the user’s position, i.e.,

\[
D_{vm}(t) = p_{\text{q1}}(t) - p_{\text{ref}}. \tag{11}
\]

According to the control strategy, \(D_{vm}(t)\) can be kept within a certain range, but it does not need to be zero when stable.

Equation (8) gives the locomotion distance in the physical environment, and the locomotion distance in VE can be obtained by the mapping process, i.e.,

\[
D_{ve}(t) = \psi(D_{pe}(t), \Theta(t)), \tag{12}
\]

Here \(\psi(x)\) denotes a preset function and \(\Theta(t)\) denotes the external parameters and variables. \(\psi(x) = x\) is the simplest function, which means 1:1 mapping from \(D_{pe}(t)\) to \(D_{ovf}(t)\). In addition, the \(\psi(x)\) could have some other formations, such as adding proportional gain, non-linear function, or even introducing the redirected-walking algorithm to get a better experience.

When the system is working, the control frequency is 20Hz. Algorithm 1 shows the framework of the HCMK1 system. Future work to improve the performance could be based on this framework.

6. https://www.wired.com/2012/08/maximum-acceleration-in-the-100-m-dash/
involves the user moving at a certain speed, then decelerating, and finally stopping; the data is generated by iterative calculation. The green dash line denotes the user’s actual intended locomotion, while the red solid line denotes the user’s local locomotion relative to the platform, i.e., \( D_{\text{ovf}}(t) \). In addition, the blue solid line denotes the locomotion of the OVF on the platform, i.e., \( D_{\text{pe}}(t) \). The black dash line is the sum of the red solid line and the blue solid line, denoting the calculated locomotion in the physical environment, i.e., \( D_{\text{rm}}(t) \).

As shown by the green dash line, the user first walks on the platform at a constant velocity of 1 and starts to stop at the 5th cycle of the simulation process. After a deceleration process, the user stops at approximately the 10th cycle, and then the actual intended velocity remains 0. The actual intended distance first increases in a linear manner, then gradually slows down, and finally remains unchanged.

Generally, this process can be divided into 4 stages, as observed in the subsequent experiment data:

1) Since the platform offsets the user’s velocity, the user’s local velocity is 0, and the local distance remains 1. The platform distance increases linearly.

2) At the 5th cycle, the user’s actual intended velocity is decreased. Since the platform velocity is still kept at a high value, the user’s local velocity will increase, but the local distance will decrease, resulting in a diminished platform velocity; accordingly, the increase of platform distance is slowed down.

3) At the 15th cycle, the user’s local velocity equals the platform’s velocity in terms of amplitudes but with opposite signs. The user’s actual intended velocity decreases to 0, and the actual intended distance remains unchanged. The calculated velocity drops to 0, and the calculated distance no longer increases.

4) Afterward, the user is gradually carried back to the center by the OVF, while the user’s local distance is slowly reduced, but the platform distance is slowly increased. Finally, the platform stops working, and the user’s local velocity is reduced to 0.

Assuming that the user has no slippage when walking on the OVF, a part of the user’s actual locomotion is eliminated by the OVF and the remaining part is the user’s local locomotion relative to the platform, which can be measured with millimeter accuracy by the positioning system. Therefore, the calculated locomotion is accurate, meaning that the black and green dash lines are equal in value. However, due to the system latency, the calculated locomotion always lags behind the user’s actual intended locomotion. In the worst case, the maximum latency would reach \( T = 9\text{ ms} \).

Different from the around 100ms system latency [15], [55], which is perceivable in the WiP strategy, the 9ms system latency of HCMK1 ensures the calculated VR locomotion synchronizes with the user’s actual intention. However, it is worth noting that the working delay caused by the controller, as shown in Fig. 5 for the machine stopping stage, is the delay after the user stops walking until the platform stops working. And the working delay is also caused by the inability of the simple proportional controller to accurately track the user’s actual intended velocity. Although this delay does not affect the correct and real-time mapping of the user’s actual intended locomotion to the VR scene, it will result in some discomforts, such as body sway [60], i.e., the user tries to compensate for the unwanted motion, resulting in multiple under-correction-overcorrection swings, thus leading to feelings of insecurity. In a word, the problem of how to diminish the working delay is a challenge for the control strategy.

### 3.6.2 End-to-End Latency

When a user is walking on HCMK1, another important latency is the end-to-end (ETE) latency, i.e., the delay between the user’s movement and changes in the display of the HMD reflecting the user’s motion. The ETE latency totals up the delay of motion tracking, transmission, computation, and finally rendering images on display [35]. In addition, it is closely related to system latency, the data refresh rate, and the refresh rate of a screen, etc.

As discussed previously, the system latency of HCMK1 is about 9ms. Restricted by the RS485 serial communication bus, the HCMK1 system could only control and calculate locomotion sequentially as shown in Algorithm 1. During the debugging process, it is necessary to record the process data through RS485, such as simultaneously recording the velocity and distance of the platform, which increases the total communication time and reduces the refresh rate of the calculated locomotion. And when it is not necessary to record these process data, according to the measurement, the maximum refresh rate of the calculated locomotion, which is also the control frequency, is 55Hz.

The timing chart starting from the user’s movement on the HCMK1 to when it is visible in the HMD is shown in Fig. 6. Since the VR system adopted is HTC Vive, the first three rows of the timing chart are based on reference [57]: where signals from the application program to the final display were synchronized through Vsync and refreshed at a rate of 90Hz. Besides, the HCMK1 operates at a rate of 55Hz as a standalone system. It takes 8ms for the HCMK1 to measure and calculate the user’s locomotion at the outset of each cycle, and then the calculated locomotion will be sent to the receiving thread of the application program; the receiver successfully receives the data after about 1ms and updates the user’s locomotion data; the user’s locomotion data in the application is updated every 18.18ms, and the updated values are those from 9ms ago, such as the data represented by green, purple and red.

For instance, in Frame#2, i.e., the green frame, when Vsync #1 arrives, the CPU waits after preparation until 2ms before Vsync #2 arrives. The application submits rendering operations related to Frame #2 to the GPU through the CPU, and it starts to Spin on Event Query [57]; after Vsync #2 arrives, the CPU terminates the Event Query and assigns a command to the GPU to start rendering Frame #2; the user’s locomotion data will be reflected in the followed GPU rendering if they are updated before the event query ends; the GPU usually finishes rendering Frame #2 within 11.11ms and sends Frame #2 to HMD panels when Vsync #3 arrives; upon the arrival of Vsync #4, the HMD panels are lit and Frame #2 is finally presented to the user. Therefore, a rendering and display delay of 22.22ms is undergone from the time when the current data are handed over to the GPU via the CPU for rendering to the final display on the HMD.
panels. In addition, since the update rate of the user’s locomotion data is lower than the refresh rate of HMD frames, as the Frame#3, the user’s locomotion data are not updated and the data used in Frame#2 is used again when rendering Frame#3.

Based on the above information, in the best case, the user’s locomotion data are updated just before the event query ends. At this time, the ETE latency is 31.22 ms, which includes the system latency of 9 ms and the rendering and display delay of 22.22 ms. While in the worst case, the user’s locomotion data are not updated in the current frame, and only reused the last updated data. At this time, the ETE latency is at its maximum of 49.4 ms, which contains an additional 18.18 ms of HCMK1’s data refresh period.

In this section, the system latency and the ETE latency of HCMK1 were discussed in detail. Although the system latency of HCMK1 is 9 ms, which is low, its ETE latency is 31.22 ms to 49.4 ms. To reduce the ETE latency, the HCMK1 system currently has a maximum working frequency of 55 Hz and sends the user’s locomotion data at an interval of 18.18 ms. Due to the frequency difference, the user’s locomotion data is not updated when HMD renders some frames, such as Frame#3, i.e., the orange part.

4 High-Level Comparisons With Other Systems

A working HCMK1 system is shown in Fig. 7. Since the OVF should be isotropic, a circular structure is more reasonable than a square structure. The mirror-symmetrical spiral rollers make it easier to design the platform close to a circle. In HCMK1, the OVF is designed as a regular octagon, and it saves the waste area on the diagonal of the square structure. Fig. 8 shows the comparison of HCMK1 with the Infinadeck and the HCP system. It shows that HCMK1 is a much lighter device and is more suitable for room-scale VR.

The detailed parameters of HCMK1 are shown in Table 3. We applied two 0.6 kW servo motors and the gear ratio is 3:1. The rotation part of HCMK1, i.e., the spiral rollers and synchronous wheels, is about 48 kg total weight and the rotation radius is 1.71 cm. Therefore, the moment of inertia is only 0.007 kg·m². The small moment of inertia ensures that even applying low-power motors can obtain sufficient acceleration to track the user’s movement. Theoretically, when assuming the transmission efficiency is 90%, the
maximum acceleration can reach 25.00 \(m/s^2\), and for a user with 100kg weight, the acceleration can reach 4.84 \(m/s^2\). The experiment in the next section shows the maximum starting acceleration is about 16.00 \(m/s^2\) and the maximum braking acceleration can reach 30.00 \(m/s^2\). The transmission efficiency is about 88% and the torque caused by rotating friction is about 2.5NM.

Table 2 demonstrates the detailed parameters of several systems that can generate parallel OVF. Cyberwalk [45], [49], F-ODT [33], [42], and Infinadeck [36] are three belt-based ODTs. Huge volume and weight lead to large inertia, which puts much more pressure on the actuator and reduces the dynamic performance. To produce enough power, Cyberwalk and F-ODT set several motors at one axis. Therefore, they need to solve synchronization errors [42] between different motors. In addition, the motors on X-axis and Y-axis are different and usually have different electrical characteristics, such as the rated speed, the torque, etc. All these will bring difficulties to the controller.

HCP [61] applied the 45-degree wheel-based scheme and reduced the volume and weight a lot based on the mirror-symmetrical chains. HCMK1 proposes a novel carrier, i.e., the mirror-symmetrical spiral rollers, to replace the chains in HCP. It significantly reduces the device weight and moment of inertia. Therefore, although the total power is only 1.2\(kW\), the dynamic performance can be greatly improved. Furthermore, the mirror-symmetrical structure ensures that identical motors can be applied on different axes.

5 EXPERIMENT RESULT

HCMK1, as an ODT with a new design scheme, has primarily demonstrated its superiority in volume and weight through the comparison in Section 4. In this section, a series of experiments were conducted around the performance of the HCMK1 in practical applications.

First, the basic performance of the platform was evaluated. Given that the HCMK1 is an active ODT, its exceptional dynamic performance provides the foundation for achieving fast tracking of the user’s motion. The amplitude and controllable range of accelerations provided by the platform were tested and are reported in Section 5.1.

Second, since the HCMK1 is a new kind of ODT, its core function is to achieve omnidirectional movement of the user, and experiments on the effectiveness of this function were conducted and are reported in Section 5.2.

In addition, the user’s movement in VR was obtained by calculation, so the inappropriate mathematical model in Equation (12), e.g., the gains of the platform OVF locomotion and the user’s local locomotion are not equal, will introduce unnecessary latency, which should be avoided in practical VR applications. In Section 5.3, experiments were conducted to illustrate this issue.

| Device | Motor          | Active Area | Height | Weight   | Maximum Speed | Maximum Acceleration |
|--------|----------------|-------------|--------|----------|---------------|---------------------|
| CyberWalk | X-axis        | 40kW(4EA)   | 6.5*6.5m | 1.45m    | 12000kg       | 2m/s                | 0.75m/s^2           |
|         | Y-axis        | 37.7kW(25EA)|        |          |               |                     |                     |
| F-ODT   | X-axis        | 8.8kW(2EA)  | 2.5*2.5m | 0.64m    | 576kg         | 2.5m/s              | 3m/s^2              |
|         | Y-axis        | 3.6kW(2EA)  |        |          |               |                     |                     |
| Infinadeck | X-axis      | -           | 1.2*1.5m | 0.4m     | 225kg         | >2m/s               | -                   |
|         | Y-axis        | -           |        |          |               |                     |                     |
| HCP     | X-axis        | 0.5kW(1EA)  | 1*1.2m  | 0.16m    | 150kg         | 1.6m/s              | 1.3m/s^2            |
|         | Y-axis        | 0.5kW(1EA)  |        |          |               |                     |                     |
| HCMK1   | X-axis        | 0.6kW(1EA)  | Radius 0.575m, 1.10m^2 | 0.08m    | 110kg         | 1.78m/s             | 25.00m/s^2          |
|         | Y-axis        | 0.6kW(1EA)  |        |          |               |                     |                     |
The experiments in the above subsections illustrate the basic performance, effectiveness, and precautions of the HCMK1 system in practice. As a human-computer interaction VR device, an excellent UX is crucial. Further improving the UX of the HCMK1 on the existing basis requires an understanding of the main factors affecting the UX. In Section 5.4, experiments and studies were conducted on the main factors affecting the UX.

5.1 Acceleration Experiment

As an active ODT, sufficient dynamic performance means that the platform can provide enough acceleration to quickly track and counteract changes in speed when the user suddenly starts or stops moving, which is the basis for designing a good controller. Given that HCMK1 has a small moment of inertia, as described in Section 4, the platform can theoretically achieve an acceleration of 25.00 \( \text{m/s}^2 \). In this subsection, several experiments are made to test the actual acceleration performance. The experiments record the starting and braking process from 0 to the maximum speed, i.e., 1.78 \( \text{m/s} \), without additional load imposed, but with different upper limits of the motor’s acceleration applied. At each control cycle, the computer control program recorded the current system status via logs, including the time stamp, the OVF locomotion distance, velocity, and acceleration. The OVF locomotion distance and velocity were obtained by the control program accessing the servo motors and then performing a coordinate transformation through Equation (7), while the acceleration is the result of applying the first-order differential in the OVF velocity.

Fig. 9 shows the results of these experiments. The first one mainly tests the maximum acceleration of the platform. The upper limit is set to 65535 \((r/min)/s\), corresponding to 39.32 \( \text{m/s}^2 \) on the surface of the OVF, a figure that is much greater than the theoretical value of 25.00 \( \text{m/s}^2 \). As indicated in Fig. 9a, the starting acceleration is about 16.00 \( \text{m/s}^2 \) and the braking acceleration is about 30.00 \( \text{m/s}^2 \). It may be caused by rotating friction. From this result, it can be calculated that the transmission efficiency is about 88% and the torque generated by rotating friction is about 2.5 \( \text{NM} \).

In the second experiment, as shown in Fig. 9b, when the upper limit is set to 20000 \((r/min)/s\), i.e., 12.00 \( \text{m/s}^2 \), the starting and braking accelerations of OVF are almost the same, about 12.5 \( \text{m/s}^2 \), indicating that the OVF is fully controllable within this range, that is, the speed change of the platform can be controlled at any acceleration. For the design of high-level controllers, the driving force is sufficient to directly control the acceleration, while ignoring the rotating friction.

The third experiment implements the upper limit of 6000 \((r/min)/s\), i.e., 3.6 \( \text{m/s}^2 \). Fig. 9c indicates that the starting and braking process from 0 to the maximum speed, i.e., 1.78 \( \text{m/s} \), without additional load imposed, but with different upper limits of the motor’s acceleration applied. At each control cycle, the computer control program recorded the current system status via logs, including the time stamp, the OVF locomotion distance, velocity, and acceleration. The OVF locomotion distance and velocity were obtained by the control program accessing the servo motors and then performing a coordinate transformation through Equation (7), while the acceleration is the result of applying the first-order differential in the OVF velocity.

Fig. 9 further shows that the X and Z axes have the same dynamic performance, which results from the mirror-symmetric spiral roller structure of HCMK1.

5.2 Locomotion Experiment

Based on Section 5.1, the HCMK1 has outstanding dynamic performance. However, as the ODT with a novel design scheme, its core function is to provide omnidirectional movement. The experiments in the above subsections illustrate the basic performance, effectiveness, and precautions of the HCMK1 system in practice. As a human-computer interaction VR device, an excellent UX is crucial. Further improving the UX of the HCMK1 on the existing basis requires an understanding of the main factors affecting the UX. In Section 5.4, experiments and studies were conducted on the main factors affecting the UX.

| Motor Model | Device Parameter |
|-------------|------------------|
| Rated Power | 0.6 kW | Total Power 1.2kW |
| Torque      | 1.9 NM | Total Area 1.76 \( \text{m}^2 \) |
| Gear Ratio  | 3:1   | Height 0.08m |
| RPM         | 3000  | Weight 110kg |
| Maximum Speed | 1.78m/s | Maximum Acceleration 25.00 \( \text{m/s}^2 \) |
| Active Area | 1.10 \( \text{m}^2 \) | User Acceleration (100kg) 4.84 \( \text{m/s}^2 \) |

Table 3: The Detail Parameters of HCMK1 System
movement for users to extend space. In this section, the effectiveness of this core function was verified through experiments. A skilled male experimenter with a height of 174 cm and a weight of 72 kg participated in this test. The locomotion experiment includes two sub-experiments, in which the user will try to walk by intuition along a circular trajectory and a square trajectory on the platform without visual cues. At each control cycle, the computer control program recorded the system status via logs, including the time stamp, the user’s local distance and velocity, the locomotion distance and velocity of the OVF, and the user’s locomotion distance, velocity and acceleration in VR. The user’s local distance and velocity were obtained from the coordinate transformation through Equation (3) after the application program accessed the positioning system; the OVF distance and velocity were obtained from the coordinate transformation by Equation (7) after the application accessed the servo motor; the user’s distance and velocity in VR were calculated by superimposing the user’s local and OVF distances and velocities, i.e., by Equation (8), while the acceleration is the result of the first derivative of the velocity.

The recorded data are shown in Fig. 10, where the first row shows the circle trajectory’s experiment result and the second row shows the square trajectory’s experiment result. The user starts to move at the green point and stops at the red point. Since the experiment did not use visual path guidance, the user came out of the above trajectory independently with their intuition, so the start and end points did not completely overlap. The fluctuations in the trajectory are caused by the swings of the human body when moving. The results in all of the experiments suggest that the platform can effectively expand the limited physical locomotion to limitless VR locomotion.

5.3 Locomotion Gains and End-to-End Latency

In the HCMK1 system, the user’s movement in the VR is obtained by superimposing the user’s local locomotion and the OVF’s locomotion, followed by the mapping in Equation (12). Although this mapping is not constrained, inappropriate computation, especially the inappropriate mathematical model in Equation (12), e.g., the gains of the platform OVF locomotion and the user’s local locomotion are not equal, will introduce unnecessary delay, which should be noticed and avoided. Experiments in this section illustrate how to avoid this problem.

It is found in the latency analysis that the VR scenes and the user’s actual intentions are almost synchronized. That is based on Equation (8), i.e., the gains of the platform OVF locomotion and the user’s local locomotion are 1:1. The mutual elimination of these two sets of locomotion ensures that the working delay would not affect the ETE latency. As described in Equation (12), changing the mapping function can introduce different algorithms or achieve special effects. However, the ratio of gains should be 1:1 in this mapping function; otherwise, the working delay will increase the ETE latency.

In this experiment, we use

$$D_{pe}(t) = -\beta_1 \cdot D_{ovf}(t) + \beta_2 \cdot D_{ru}(t).$$

(13)

to analyze the influence of different gains on latency. Based on the simulation result in Fig. 11, the following simple inferences can be acquired:

1) When the user stops, if $\beta_1 < \beta_2$, the locomotion of platform OVF cannot completely eliminate the user’s local locomotion; therefore, the VR scene would slide in the backward direction;

2) If $\beta_1 = \beta_2$, the locomotion of platform OVF would eliminate the user’s local locomotion exactly; therefore, the VR scene will stop synchronously;

3) If $\beta_1 > \beta_2$, the locomotion of platform OVF would excessively eliminate the user’s local locomotion, therefore, the VR scene will continue to slide in the forward direction.

The VR scene in the experiments is built in Unity3d. One participant in this experiment has a height of 174 cm and a weight of 72 kg. The HMD is HTC Vive. Since human bodies

Fig. 10. User’s locomotion experiments. Each row represents an experiment in which the user walks along a certain trajectory. The first column is the user’s local position relative to the platform. The second column is the user’s VR position relative to the coordinate system in VE. The green point denotes the start point, and the red point denotes the endpoint. The third and fourth columns represent the corresponding spatial information in the VE coordinate system along the X and Z axis, respectively.
will inevitably sway when the platform is working, and holding the security handrail will change the user’s gaits, in order to reduce the impact of body sway and holding the security handrail on VR scenes, the experimenter is required to keep as stable as possible without the aid of the security handrail after stopping his moves. At each control cycle, the control program accessed the servo motor to calculate the OVF distance and speed and forwarded them to the VR application program. Before rendering each frame of the VR scene, the OVF velocity is used for integration to compensate for the OVF distance when it is not updated; the VR application program directly accesses the positioning system to obtain the user’s local distance and velocity. After that, the user’s locomotion in VR was calculated using different gains based on Equation (13). The three experimental ratios between \( \beta_1 \) and \( \beta_2 \) is set as 0.5:1, 1:1 and 5:1.

By selecting the data with less body sway, the results in Fig. 12 are obtained. The experimental results are presented in the form of synthetic video frames, including the third perspective of the physical environment and the first perspective in the VR scene, which were recorded at a frame rate of 30 fps by video camera and screen recording, respectively. Thenceforth, the two videos were edited synchronously based on audio. The frame rate of the synthetic video is 29.97 fps. The 1st, 5th and 10th frames of the user’s stopping process are extracted. The continuous experimental video is attached in the supplementary material, available online.

Different gain ratios are applied in different columns in Fig. 12. The positions of the user on the platform and the salient reference objects in the scenes are zoomed in. It can be clearly observed that when \( \beta_1 < \beta_2 \), the VR scenes would slide backward; when \( \beta_1 > \beta_2 \), the VR scenes would continue to slide forward; and when \( \beta_1 = \beta_2 \), the VR scenes would stop immediately. This experiment demonstrates that when calculating a user’s locomotion or introducing mapping functions as Equation (12), it is necessary to ensure that the gains of the platform OVF locomotion and the user’s local locomotion will inevitably sway when the platform is working, and holding the security handrail will change the user’s gaits, in order to reduce the impact of body sway and holding the security handrail on VR scenes, the experimenter is required to keep as stable as possible without the aid of the security handrail after stopping his moves. At each control cycle, the control program accessed the servo motor to calculate the OVF distance and speed and forwarded them to the VR application program. Before rendering each frame of the VR scene, the OVF velocity is used for integration to compensate for the OVF distance when it is not updated; the VR application program directly accesses the positioning system to obtain the user’s local distance and velocity. After that, the user’s locomotion in VR was calculated using different gains based on Equation (13). The three experimental ratios between \( \beta_1 \) and \( \beta_2 \) is set as 0.5:1, 1:1 and 5:1.

By selecting the data with less body sway, the results in Fig. 12 are obtained. The experimental results are presented in the form of synthetic video frames, including the third perspective of the physical environment and the first perspective in the VR scene, which were recorded at a frame rate of 30 fps by video camera and screen recording, respectively. Thenceforth, the two videos were edited synchronously based on audio. The frame rate of the synthetic video is 29.97 fps. The 1st, 5th and 10th frames of the user’s stopping process are extracted. The continuous experimental video is attached in the supplementary material, available online.

Different gain ratios are applied in different columns in Fig. 12. The positions of the user on the platform and the salient reference objects in the scenes are zoomed in. It can be clearly observed that when \( \beta_1 \neq \beta_2 \), the VR scenes’ stops always lag significantly behind the user’s stop actions. This finding is consistent with the inference above: when \( \beta_1 < \beta_2 \), the VR scenes would slide backward; when \( \beta_1 > \beta_2 \), the VR scenes would continue to slide forward; and when \( \beta_1 = \beta_2 \), the VR scenes would stop immediately.

This experiment demonstrates that when calculating a user’s locomotion or introducing mapping functions as Equation (12), it is necessary to ensure that the gains of the platform OVF locomotion and the user’s local locomotion...
are equal. Otherwise, it will bring about a significant latency between the VR scene and the user’s intention.

5.4 Main Factors Affecting Users’ Experience

The above experiments and discussions show that HC MK1 is an effective solution to realize VR space expansion. However, as a human-computer interaction VR device, the realization of basic functions is far from enough; a natural and immersive UX in practice is crucial. The analysis of the UX of the current system can effectively guide the improvement resulting from future work. This subsection first conducts a qualitative analysis based on the self-reported feelings of participants, and then analyzes the main factors affecting the UX combined with a set of typical data collected in the trials.

5.4.1 User Discomfort Survey

At present, a total of 30 participants (including 5 females and 25 males) have experienced VR interaction on this system. Age range: 20 to 45 (27.67±7.68). Height range: 155 to 198 cm (172.03±8.01 cm). Weight range: 45 to 99 kg (67.82±12.73 kg). All of the participants were first-time users. After walking on the treadmill for about 1 min without wearing HMD, they all then entered the same VR scenes to have a free interaction.

Aside from reminding the participants about the security handrail, no other guidance or instructions are given. The average experience time is 10 minutes. An example of the interaction process is shown in the supplemental application demo video, available online. After completing the experience, each of the participants was asked whether they had experienced motion sickness or any other discomforts.

The inquiry finds that 29 participants had good experiences without motion sickness, with only one, who is 198 cm tall and weight 99 kg, reporting slight motion sickness. Besides, during the experiments, 4 participants attempted to release the security handrail when walking. Among them, two found it was hard to stabilize their bodies when stopping and turning around, with a feeling of falling down; another two indicated no abnormalities. None of these 4 participants experienced motion sickness.

According to the above results, the UX needs to be further improved. Although HC MK1 has many advantages in its design, like other driven-based ODTs, the UX would ultimately depend on the performance of the controller [4], [12]. This is a human-computer interaction problem. The ideal UX is that no matter when a user starts or stops moving at any speed, their body can always keep stationary at the center of the platform. Obviously, there is still a certain gap between the current experiences and the ideal cases. To further pinpoint the causes of the above results to boost the controller’s performance, this study collects and analyzes the actual data through the following start-stop experiments.

5.4.2 Start-Stop Experiment

The experiment includes two independent start-stop trials, where the participants performed several start-stop motions along the X and Z axes, respectively, wearing HTC Vive HMD and Tracker. The participants were asked to accelerate from a stationary state, walk at a constant speed for a while, and then immediately stop stepping and remain stationary until the platform stopped running. During the whole process, the user moved without the help of handrails. Unlike the latency experiment in Section 5.3, in this experiment, the participant was only asked to walk naturally, without deliberately keeping the body stable. At each control cycle, the computer control program recorded the current system status via logs, including the time stamp, the user’s local distance and velocity, the locomotion distance and velocity of the OVF, and the user’s locomotion distance, and velocity in VR. The user’s local distance and velocity were obtained from the coordinate transformation through Equation (3) after the application program accessed the positioning system; the OVF distance and velocity were obtained from the coordinate transformation by Equation (7) after the application accessed the servo motor; the user’s distance and velocity in VR were calculated by superimposing the user’s local and OVF distances and velocities, where the calculation process is based on Equation (8), i.e., β₁ = β₂ = 1.0.

A total of 6 people participated in data collection, and for different users, most of the data are repeated and similar. An example of a typical user’s data is shown in Fig. 13. The selected data is from a user with a height of 174 cm and a weight of 72 kg. Other experimental data are attached in the supplementary data, available online. As shown in Fig. 13, the start-stop process corresponds to the three dark-red dash-line boxes, i.e., 1, 2, 3. Since the results of the X and Z axes are almost the same, the discussion will mainly focus on the X-axis.

In the development stage of walking [16], i.e., Box 1, when the user starts walking but has not moved beyond the threshold of the controller, the platform keeps stationary. The user’s local velocity increases until the user’s position exceeds the threshold. Then, the platform starts to carry the user backward and thus slows down the user’s local velocity; however, the user’s VR velocity still rises as indicated by the actual intention, therefore leading to the peak of the red dash-line.

When the user’s local velocity decreases to approach 0, the rhythmic stage [16], i.e., Box 2, starts and the user’s actual intended velocity is eliminated by the platform velocity. When the user steps, the local velocity fluctuates, but the average value is approximately 0.

The decay stage [16], i.e., Box 3, starts when the user suddenly stops. The local velocity is lowered to a negative value due to the influence of the platform velocity. Then the user’s local position falls below the threshold and the platform starts working. It is worth noting that, although the peak of VR velocity, as shown by the blue dashed line in Box 3, resembles the case of β₁ < β₂ in Fig. 11, when the OVF velocity decreases to 0, the user’s local velocity is not 0. From this finding, it can be inferred that the peak is not caused by the unmatched gains in the calculation process, but because of the swaying of the body. In contrast, in Box 1, the user keeps the body swaying at a low level during the stop stage. When the platform OVF velocity falls to 0, the user’s local velocity becomes substantially 0, while the calculated VR velocity drops to 0 earlier than the platform.

In both X-axis and Z-axis, it can be observed that the swaying of the body during the decay stage is a common problem. Due to the working delay, the platform OVF velocity fails to track the user’s intended velocity quickly. After the user stops, the platform takes some time to send the
user back to the center. Meanwhile, the center of gravity of the user is far above the ground. As a result, after the platform sends the user back to the center and stops working, the body still moves at a certain speed due to inertia, which leads to the swaying of the body. Besides, the higher the user’s center of gravity, the more intense the swaying phenomenon. Once the sway reaches a certain extent, a sense of motion sickness will appear. The fall-down feeling appearing when stopping and turning without holding the handrail can also be explained by the same reason. In addition, physiological studies reveal that humans are much more sensitive to lateral acceleration [12]; therefore, body sway caused by the working delay will become even more serious when turning around.

Although previous research has found that the main factor affecting the UX is users’ position errors in the abrupt stop stage [4], this study maintains that such errors are just one of the main reasons affecting the UX, and other reasons also include the system’s working delay and the height of the center of gravity. The distance for OVF to send users back to the center can be shortened by cutting users’ local distance in the rhythmic stage. Therefore, minimizing the initial position errors of users during the decay stage can lower the working delay for a while. Another method to reduce the working delay is to predict users’ stopping action in advance and set the platform OVF velocity to 0, so that the platform OVF can quickly follow the users to stop. In addition, in order to obtain a good UX, body sway caused by inertia is unacceptable. Therefore, when designing the controller, it is necessary to consider inhibiting the influence of the height of the center of gravity. For example, limiting the jerk, i.e., the first-time derivative of acceleration, of the platform based on the height makes the process of transporting the user smoother, thereby reducing body sway.

6 LIMITATIONS

Although HCMK1 is superior to other similar systems, it still has some limitations with respect to the mechanism, algorithms and metrics of UX. These limitations may limit the possible application scenarios of HCMK1.

We noticed that the working noise of HCMK1 was relatively large, which might have a negative effect on the UX. The noise was mainly caused by the power supply fan and mechanical transmission. Moreover, grasping the security handrail might negatively affect the user’s ability to use the hand-held controller for VR interactions.

The current system uses the RS485 to collect the essential data, which greatly increases the time of data measurement, reduces the control frequency, and further increases the system latency and ETE latency. A new method to measure the data is important to improve system performance.

The control algorithm is the key problem that needs to be studied and solved urgently in the current system. This paper uses proportional control for verification, whose final performance is not satisfactory. Because the current working delay of about 0.5s might have a negative impact on the UX, the proportional control algorithm that we used in HCMK1 should be replaced by better control algorithms for optimal performance.

In addition, the current evaluation of the UX is only subjective: the data is collected from users’ self-reported answers to queries. The lack of a quantitative metric makes it difficult to distinguish between different UXs created with different algorithms.

7 CONCLUSIONS AND FUTURE WORK

This paper has proposed and developed a novel ODT system, namely HCMK1, suitable for household VR applications. The design scheme not only provides a natural walking experience but also has great advantages with respect to weight, volume, dynamic performance, and latency, making it possible to achieve the natural locomotion experience in room-scale VR. With the lightweight, HCMK1 is easier to integrate with other devices, such as installing the HCMK1 on a miniaturized 2DoF-6DoF motion simulator platform that could provide tilt, vibration, and
simulate terrains like uphill and downhill. Through the experiments and analysis, this paper has systematically validated that when the gains of OVFs’ locomotion and the user’s local locomotion match, HCMK1 has only a minor system latency of 9ms and ETE latency of 31.22ms to 49.4ms. By analyzing the results of the UX, this paper has identified several main factors, including the working delay and the height of the center of gravity. These results may help design better controllers to improve UX in the future.

Our study opens new directions for future work. The controller design of ODT is a typical human-computer interaction problem. Due to the randomness of human motion and ODT directly affecting human gait, the research in this area will involve the estimation of human motion and complex control theory. We plan to introduce the height of the center of gravity into the controller design to achieve a better control effect. Moreover, in order to improve the prediction performance of the controller, the estimation of human motion will be added to the controller as auxiliary information. Furthermore, we will consider applying reinforcement learning strategies to solve such human-computer interaction problems. As for the UX evaluation, in order to be distinguish and improve the UX brought by different controllers, we will build a single quantitative metric that can directly reflect the UX. The metric can be used to objectively guide the design of the controller and help to formulate a reward function in the reinforcement learning strategy.

REFERENCES
[1] B. D. Adelstein, T. G. Lee, and S. R. Ellis, “Head tracking latency in virtual environments: Psychophysics and a model,” in Proc. Hum. Factors Ergonom. Soc. Annu. Meeting, 2003, pp. 2083–2087.
[2] M. Al Zayer, P. MacNeilage, and E. Folmer, “Virtual locomotion: A survey,” IEEE Trans. Vis. Comput. Graph., vol. 26, no. 6, pp. 2315–2334, Jun. 2020.
[3] 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 Computations, Cham, Switzerland: Springer, 2018, pp. 367–373.
[4] H. J. Asl, S.-H. Pyo, and J. Yoon, “An intelligent control scheme to facilitate abrupt stopping on self-adjustable treadmills,” in Proc. IEEE Int. Conf. Robot. Autom., 2018, pp. 1639–1644.
[5] E. R. Bachmann, E. Hodgson, C. Hoffbauer, and J. Messinger, “Multi-user redirected walking and resetting using artificial potential fields,” IEEE Trans. Vis. Comput. Graph., vol. 25, no. 5, pp. 2022–2031, May 2019.
[6] A. Benjamin, “Why virtual reality lack of content?,” EB/Ol, Aug. 2021. [Online]. Available: https://filmona.wondershare.com/ virtual-reality/why-virtual-reality-lack-of-content.html
[7] L. Bölling, N. Stein, F. Steinicke, and M. Lappe, “Shrinking circles: Adaptation to increased curvature gain in redirected walking,” IEEE Trans. Vis. Comput. Graph., vol. 25, no. 5, pp. 2032–2039, May 2019.
[8] H. Brinks and M. Bruins, “Redesign of the Omnideck platform: With respect to DfA and modularity,” EB/Ol, 2016. [Online]. Available: http://lnu.diva-portal.org/smash/get/diva2:932520/ FULLTEXT02.pdf
[9] T. J. Baker, D. A. Vincenzi, and J. E. Deaton, “The effect of apparent latency on simulator sickness while using a see-through helmet-mounted display: Reducing apparent latency with predictive compensation,” Hum. Factors, vol. 54, no. 2, pp. 253–249, 2012.
[10] T. Cakmak and H. Hager, “Cyberth virtualizer: A locomotion device for virtual reality,” in Proc. ACM SIGGRAPH Emerg. Technol., 2014, Art. no. 6.
[11] R. P. Darken, W. R. Cockayne, and D. Carmein, “The Omni-directional treadmill: A locomotion device for virtual worlds,” in Proc. 10th Annu. ACM Symp. User Interface Softw. Technol., 1997, pp. 213–221.
[12] A. De Luca, R. Mattone, P. R. Giordano, and H. H. Bültzho, “Control design and experimental evaluation of the 2D cyberwalk platform,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2009, pp. 5051–5058.
[13] A. De Luca et al., “Motion control of the CyberCarpet platform,” IEEE Trans. Control Syst. Technol., vol. 21, no. 2, pp. 410–427, Mar. 2013.
[14] S. R. Ellis, K. Mania, B. D. Adelstein, and M. I. Hill, “Generalizability of latency detection in a variety of virtual environments,” in Proc. Hum. Factors Ergonom. Soc. Annu. Meeting, 2004, pp. 2632–2636.
[15] 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, 2008, pp. 97–104.
[16] J. G. Gamble, J. Rose, and J. G. Gamble, Human Walking, 3rd ed. Philadelphia, PA, USA: Lippincott Williams & Wilkins, 2006.
[17] S. Hanson, R. A. Paris, H. A. Adams, and B. Bodenheimer, “Improving walking in place methods with individualization and deep networks,” in Proc. IEEE Conf. Virtual Reality 3D User Interfac- es, 2019, pp. 367–376.
[18] J. M. Hollerbach, “Simulation interfaces,” in Handbook of Virtual Environments. Boca Raton, FL, USA: CRC Press, 2002, pp. 279–294.
[19] J.-Y. Huang, “An omnidirectional stroll-based virtual reality interface and its application on overhead crane training,” IEEE Trans. Multimedia, vol. 5, no. 1, pp. 39–51, Mar. 2003.
[20] B. E. Ilon, “Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base,” U.S. Patent 3 876 255, Apr. 08, 1975.
[21] H. Iwata, “Walking about virtual environments on an infinite floor,” in Proc. IEEE Virtual Reality, 1999, pp. 286–293.
[22] H. Iwata, “Locomotion interface for virtual environments,” in Robotics Research. Berlin, Germany: Springer, 2000, pp. 275–282.
[23] H. Iwata, H. Yano, H. Fukushima, and H. Noma, “CirculaFloor [locomotion interface],” IEEE Comput. Graph. Appl., vol. 25, no. 1, pp. 64–67, Jan./Feb. 2005.
[24] H. Iwata, H. Yano, and M. Tomiyoishi, “String walker,” in Proc. ACM SIGGRAPH Emerg. Technol., 2007, pp. 20–es.
[25] A. Jenkins, “The fall and rise of VR: The struggle to make virtual reality get real,” EB/Ol, Jun. 2019. [Online]. Available: https://finance.yahoo.com/news/fall-rise-vr-struggle-virtual-103009561. html
[26] J. Jerald and M. Whitton, “Relating scene-motion thresholds to latency thresholds for head-mounted displays,” in Proc. IEEE Virtual Reality Conf., 2009, pp. 211–218.
[27] J. A. Jones, E. Luckett, T. Key, and N. Newsome, “Latency measurement in head-mounted virtual environments,” in Proc. IEEE Conf. Virtual Reality 3D User Interfaces, 2019, pp. 1000–1001.
[28] E. M. Kolasinski, “Simulator sickness in virtual environments,” US Army Res. Inst. Behav. Social Sci., Alexandria, VA, USA, vol. 1027, pp. 1–2, 2005.
[29] E. Langbehn, P. Lubos, and F. Steinicke, “Evaluation of locomotion techniques for room-scale VR: Joystick, teleportation, and redirected walking,” in Proc. Virtual Reality Int. Conf.–Laval Virtual, 2018, Art. no. 4.
[30] J. J. Lapiola Jr., “A discussion of cybersickness in virtual environments,” ACM SIGCHI Bull., vol. 32, no. 1, pp. 47–56, 2000.
[31] B. Lawson, “Motion sickness symptomatology and origins,” in Handbook of Virtual Environments: Design, Implementation, and Applications. Boca Raton, FL, USA: CRC, 2014.
[32] D.-Y. Lee, Y.-H. Cho, and I.-K. Lee, “Real-time optimal planning for redirected walking using deep Q-learning,” in Proc. IEEE Conf. Virtual Reality 3D User Interfaces, 2019, pp. 63–71.
[33] H. Lee, S. Pyo, S. Park, and J. Yoon, “Design of the omnidirectional treadmill based on an Omni-pulley mechanism,” in Proc. 13th Int. Conf. Ubiquitous Robots Ambient Intell., 2016, pp. 889–894.
[34] 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, 2018, Art. no. 2832.
[35] P. Lincoln et al., “From motion to photons in 80 microseconds: Towards minimal latency for virtual and augmented reality,” IEEE Trans. Vis. Comput. Graph., vol. 22, no. 4, pp. 1367–1376, Apr. 2016.
[36] V. Metsis, K. S. Smith, and D. Gobe, “Integration of virtual reality with an omnidirectional treadmill system for multi-directional balance skills intervention,” in Proc. Int. Symp. Wearable Robot. Rehabil., 2017, pp. 1–2.
[37] D. C. Niehorster, L. Li, and M. Lappe, “The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research,” i-Perception, vol. 8, no. 3, 2017, Art. no. 20141669517708205.

[38] N. C. Nilsson, S. Serafin, F. Steinicke, and R. Nordahl, “Natural walking in virtual reality: A review,” Comput. Entertainment, vol. 16, no. 2, pp. 1–22, 2018.

[39] Okreylos, “Lighthouse tracking examined,” EB/OL, May 2016. [Online]. Available: http://doc-ok.org/?p=1478

[40] 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.

[41] K. Farrish, “Pricing and lack of content are still barriers against the adoption of VR,” EB/OL, Aug. 2018. [Online]. Available: https://www.digitaltrends.com/computing/vr-pros-see-pricing-and-content-as-mainstream-barriers/

[42] S. H. Pyo, H. S. Lee, B. M. Phu, S. J. Park, and J. W. Yoon, “Development of a fast-omnidirectional treadmill (F-ODT) for immersive locomotion interface,” in Proc. IEEE Int. Conf. Robot. Autom., 2018, pp. 766–772.

[43] M. Steinicke, M. Deubzer, T. Dreja, and E. Rukzio, “Telewalk: Towards free and endless walking in room-scale virtual reality,” in Proc. CHI Conf. Hum. Factors Comput. Syst., 2020, pp. 1–9.

[44] P. Schmitz, J. Hildebrandt, A. C. Valdez, L. Kobbelt, and M. Ziefle, “You spin my head right round: Threshold of limited immersion for rotation gains in redirected walking,” IEEE Trans. Vis. Comput. Graph., vol. 24, no. 4, pp. 1623–1632, Apr. 2018.

[45] M. C. Schaewiger, T. Thummel, and H. Ulbrich, “Cyberwalk: An advanced prototype of a belt array platform,” in Proc. IEEE Int. Workshop Haptic Audio Vis. Environ. Games, 2007, pp. 50–55.

[46] M. C. Schaewiger, T. Thummel, and H. Ulbrich, “A 2D-motion platform: The cybercarpet,” in Proc. 2nd Joint EuroHaptics Conf. Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst., 2007, pp. 415–420.

[47] M. Slater and M. V. Sanchez-Vives, “Enhancing our lives with immersive virtual reality,” Front. Robot. AI, vol. 3, 2016, Art. no. 74.

[48] M. Slater, M. Usoh, and A. Steed, “Steps and ladders in virtual reality,” in Virtual Reality Software and Technology, World Scientific: Singapore, 1994, pp. 45–54.

[49] J. L. Souman et al., “CyberWalk: Enabling unconstrained omnidirectional walking through virtual environments,” ACM Trans. Appl. Percep., vol. 4, no. 4, 2011, Art. no. 25.

[50] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe, “Estimation of detection thresholds for redirected walking techniques,” IEEE Trans. Vis. Comput. Graph., vol. 16, no. 1, pp. 17–27, Jan./Feb. 2010.

[51] F. Steinicke, Y. Visell, J. Campos, and A. Lécuyer, Human Walking in Virtual Environments, vol. 2. Berlin, Germany: Springer, 2013.

[52] K. Teknomo, “Microscopic pedestrian flow characteristics: Development of an image processing data collection and simulation model,” PhD dissertation, Mar. 2002.

[53] J. N. Templeman, P. S. Denbrook, and L. E. Sibert, “Virtual locomotion: Walking in place through virtual environments,” Presence, vol. 8, no. 6, pp. 598–617, 1999.

[54] J. N. Templeman, L. E. Sibert, R. C. Page, and P. S. Denbrook, Immersive Simulation to Train Urban Infantry Combat. Washington, DC, USA: Naval Res. Lab Washington DC Information Technology Div., 2006, pp. 23:1–23:16.

[55] 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.

[56] M. Usoh et al., “Walking walking-in-place > walking-in-place > flying, in virtual environments,” in Proc. 26th Annu. Conf. Comput. Graph. Interact. Techn., 1999, pp. 359–364.

[57] A. Vlachos, “Advanced VR rendering,” EB/OL, 2015. [Online]. Available: https://www.gdcvault.com/play/1021771/Advanced-VR

[58] D. Waller, E. Bachmann, E. Hodgson, and A. C. Beall, “The hive: A huge immersive virtual environment for research in spatial cognition,” Behav. Res. Methods, vol. 39, no. 4, pp. 835–843, 2007.

[59] J. Webster, V. A. Dogaru, P. L. Strandholt, N. C. Nilsson, R. Nordahl, and S. Serafin, “Step aside: An initial exploration of gestural input for lateral movement during walking-in-place locomotion,” in Proc. 24th ACM Symp. Virtual Reality Softw. Technol., 2018, pp. 1–5.

[60] F. Wang, M. Skubic, C. Abbott, and J. M. Keller, “Body sway measurement for fall risk assessment using inexpensive webcams,” in Proc. IEEE Annu. Int. Conf. Eng. Med. Biol., 2010, pp. 2225–2229.

Ziyao Wang is currently working toward the PhD degree with the School of Automation, Southeast University. He is interested in mechanical manufacturing, control, intelligent algorithms, visualization, and human-computer interaction. He strives to be an inventor. Maybe one day he will be able to launch a rocket to Mars.

Chiyi Liu received the master's degree in control engineering, focusing on the study and research of the Internet of Things from Southeast University. He is a senior engineer with ByteDance Technology Co., Ltd., mainly engaged in international payment-related business.

Jialiang Chen received the bachelor's degree from Southeast University, Nanjing, China. Now, he is the leader of Myths EU Corporate, which is a multinational business entity engaging in the import, distribution, and retail of Chinese goods in the Single Euro Payments Area (SEPA). Currently, they are aiming to build a distribution network in the whole Europe with a fast expanding strategy.

Yao Yao received the bachelor's degree from Southeast University, Nanjing, China, in 2022. She is currently working toward the PhD degree with the Center for Brain-like Computing and Machine Intelligence, Shanghai Jiao Tong University, Shanghai, China. Her research interests mainly include natural language processing, especially lie in information extraction, coreference resolution, and semantic role labeling.

Dazheng Fang received the bachelor's degree from Southeast University, Nanjing, China. Now, he is a software development engineer with SYH Digital Technology Co., Ltd. focusing on HEX omnidirectional treadmill detecting algorithm and Product marketing. He studies human-computer interaction, virtual reality development.
Zhiyi Shi received the bachelor’s degree from Southeast University, Nanjing, China, in 2022. He is currently working toward the master degree with Carnegie Mellon University, Pittsburgh, US. His research interests mainly include image generation and image registration especially applied in medical images.

Rui Yan is currently working toward the master’s degree with Southeast University. Her fields of research are robotics engineering and intelligent systems, particularly how to conduct inspection in an unattended environment.

Yiye Wang is currently working toward the PhD degree with Southeast University, Nanjing, China. Her current research interests include human-computer interactive virtual reality, neural networks, and saliency analysis.

KanJian Zhang received the BS degree in mathematics from Nankai University, China, in 1994, and the MS and PhD degrees in control theory and control engineering from Southeast University, China, in 1997 and 2000, respectively. He is currently a professor with the School of Automation, Southeast University. His research is in nonlinear control theory and its applications, with particular interest include robust output feedback design and optimization control.

Hai Wang received the BSc degree in computer science from the University of New Brunswick, in 1995, and the MSc and PhD degrees in computer science from the University of Toronto, in 1997 and 2004, respectively. He is a professor with the Department of Finance, Information Systems and Management Science, Sobey School of Business, Saint Mary’s University. He was an assistant professor and then associate professor with the Saint Mary’s University from 2004 to 2016. He also holds an adjunct professor appointment with the Faculty of Computer Science, Dalhousie University. His research is currently funded by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Haikun Wei received the BS degree from the Department of Automation, North China University of Technology, China, in 1994, and the MS and PhD degrees from the Research Institute of Automation, Southeast University, China, in 1997 and 2000, respectively. He was a visiting scholar with RIKEN Brain Science Institute, Japan from 2005 to 2007. He is currently a professor with the School of Automation, Southeast University. His research interest include real and artificial in neural networks and industry automation.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.