Walking World Model for Visually Impaired Path Following

Haokun Ju, *Student Member, IEEE*, Lixuan Zhang, *Student Member, IEEE*, Xiangyu Cao, Meina Kan, *Member, IEEE*, Shiguang Shan, *Fellow, IEEE* and Xilin Chen, *Fellow, IEEE*

***Abstract*—Guiding visually impaired individuals (VI) walking along planned paths is essential for enabling independent long-distance mobility. Current reactive approaches only correct deviations after they occur. These methods ignore VI's walking dynamics (e.g., reaction latency and heading drift), resulting in frequent interventions that increase cognitive load, reduce walking efficiency, and may lead to missed turns. To address these limitations, we propose a predictive path-following approach enhanced by a walking world model to enable proactive guidance through vibrotactile guidance commands. Specifically, our walking world model is used to predict the future state of users after receiving specific commands. To mitigate the inefficiency in collecting action-annotated walking data, we exploit unannotated free-walking data to enhance model generalization. Specifically, the model first undergoes self-supervised pre-training on a large unannotated dataset to learn general gait patterns, and then is fine-tuned on annotated data with action labels to model the walking dynamics of users given guidance commands. Integrated with model predictive control (MPC) specially considering cognitive load for the human, our method proactively optimizes instructions to minimize deviation, ensure safety, and reduce cognitive load. Experiments show significant improvements in walking speed and cognitive load over reactive baselines.**

***Index Terms*—Design and human factors, wearable robotics, human-centered robotics.**

I. INTRODUCTION

As urban environments grow increasingly crowded and complex, navigation poses significant challenges for visually impaired individuals (VI). Traditional aids such as white canes and guide dogs often prove insufficient in dynamic or cluttered environments [1]. To enhance mobility of VI, modern navigation systems usually employ intelligent techniques to provide more reliable guidance. These systems typically comprise three core components: [2]–[5] : an environmental perception and mapping module, a path planning module and a user interface. While substantial progress has been achieved in perception and planning, interface design remains a major bottleneck. Since VI rely solely on voice or haptic feedback which are both low-bandwidth channels [6], poorly designed interfaces risk overwhelming users, increasing cognitive load, and ultimately reducing navigation efficiency and trust.

Current VI navigation interfaces fall into two main categories. The first provides direct physical guidance [7]–[10]. These systems leverage robotic mechanisms for path following, offering rigid physical guidance. However, their use in direct physical guidance is constrained by incomplete modeling of human–robot joint dynamics, making these systems still insufficient to guide visually impaired users. The second category relies on multimodal instruction guidance, typically through auditory [11]–[13] or vibrotactile cues [14], [15] to direct the user's movement. Among these modalities, vibrotactile feedback stands out as discreet and minimally restrictive, enabling greater flexibility and autonomy. As a result, it has been increasingly recognized as a user-friendly and preferred option for navigation support [16], [17].

However, most existing systems employing auditory or vibrotactile feedback use reactive strategies that issue guidance commands only after the user deviates from the target path [14], [15]. Such strategies neglect the how the users respond to the guidance command. This oversight leads to inefficient repeated command issuance until realignment is achieved, which not only increases cognitive load but also reduces walking speed. To address these limitations, we propose a predictive path-following approach for VI that integrates a walking world model with model predictive control (MPC). The walking world model explicitly captures the walking dynamics of VI. We use a navigation system that employs a haptic belt as the user interface and integrates our walking world model, enabling the system to issue real-time guidance commands—minimizing deviation, enhancing safety, and reducing corrective interventions for smoother, more efficient navigation.

To accurately capture the walking dynamics of visually impaired, the world model requires a large amount of action-annotated data for training, yet in practice, collecting such data is highly inefficient. Therefore, we designed a two-stage training approach to enhance the model's generalization capability using unlabeled free-walking data. Specifically, the model is firstly pre-trained on large-scale action-free walking data collected from both different participants, allowing it to learn general gait patterns. Next, it is fine-tuned on a smaller, action-annotated data to enable the model to accurately predict user reactions to different guidance commands, enhancing prediction fidelity. The walking world model is then integrated into an MPC framework. Considering the low-bandwidth nature of human processing of vibrotactile commands and cognitive load caused by excessively high command rates, we also

Manuscript received: July, 4, 2025; Revised September, 24, 2025; Accepted November, 7, 2025.

This paper was recommended for publication by Editor Sonia Chernova upon evaluation of the Associate Editor and Reviewers' comments. This work is supported by the National Natural Science Foundation of China (Nos.62495082, 62495084, and 62461160331). (*Corresponding author: Meina Kan.*)

Haokun Ju, Lixuan Zhang, Xiangyu Cao, Meina Kan, Shiguang Shan, and Xilin Chen are with the State Key Laboratory of AI Safety, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (email: juhaokun24z@ict.ac.cn; lixuan.zhang@vipl.ict.ac.cn; 2512483264@qq.com; kanmeina@ict.ac.cn; sgshan@ict.ac.cn; xlchen@ict.ac.cn).

Digital Object Identifier (DOI): see top of this page.

This work involved human subjects in its research. Approval of all ethical and experimental procedures and protocols was granted by the Research Ethics Committee of the Institute of Computing Technology, Chinese Academy of Science under Approval No. JLS2023.

The experimental data and demonstration videos are available at the project website: https://haokunju.github.io/walking-wm.github.io

introduce an MPC framework with cognitive load cost to reduce cognitive load during user walking.

Our main contributions are as follows:

(1) We propose a predictive path-following approach for the visually impaired, whose core lies in a walking world model combined with an MPC framework for optimized guidance command generation. To the best of our knowledge, this is the first successful demonstration of this technology in real-world setting for visually impaired navigation.
(2) To leverage unannotated free-walking data for enhancing the world model's generalization, we propose a two-stage training strategy: first pre-training on large-scale action-free data to capture general gait patterns, then fine-tuning on smaller action-annotated walking data to model user walking dynamics under specific guidance commands.
(3) To reduce cognitive load during user walking, we design an MPC framework considering command frequency by introducing the cost of control frequency.
(4) We evaluated our system with a total of 16 participants—8 VI and 8 eye-masked—across static and dynamic real-world scenarios. Results show notable performance gains and indicate improved safety and reduced cognitive load versus baseline methods.

II. RELATED WORKS

In recent years, navigation systems for the VI have diversified significantly. The interface between the user and the navigation system plays a crucial role in determining usability and effectiveness [2]–[5], [7]–[9]. Current implementations primarily follow two paradigms: physical guidance and multimodal instruction interfaces. Physical guidance systems utilize mechanical actuators to directly steer or constrain user movement during navigation. In contrast, multimodal instruction systems provide guidance through sensory feedback channels such as auditory or haptic cues, enabling more flexible wayfinding.

A. Direct Physical Guidance

Physical guidance systems provide mechanical assistance to VI users by physically steering or pulling them along a path, mimicking the role of guide dogs. These systems often formulate the problem of path following for VI as robot path following, focusing primarily on how the robot should follow a planned path while assuming the human follows passively. Early implementations adopted open-loop control strategies [18]–[20]. For a specified steering angle, the systems calculate a fixed motor steering, which may result in inaccurate adjustments when there are variations in user movement. More recent methods adopt closed-loop controllers that react to user's feedback [7], [9], [21]. For example, Balatti et al. [7] proposed a system using adaptive impedance control to adjust the robot's pulling force in response to the user's movements, ensuring safer navigation experience. Robotic Guide Dog [21] employs a leash-guided system, where a quadrupedal robot adjusts its movement in response to changes in the tension of the leash connected to the user. Although the aforementioned closed-loop control systems account for user feedback and are easy to develop and implement, the lack of sufficient research on human–robot joint dynamics introduces potential risks when applying them to assistive navigation. For example, during sharp turns, the robot's inertia can push the walking user outward, risking collisions with obstacles the robot avoids [22]. Moreover, the robot must be positioned at a certain distance from the user to pull them, which may be restricted in narrow spaces.

B. Multimodal Instruction Guidance

As a less intrusive alternative, multimodal instruction interfaces [11]–[15], [23] deliver guide cues via sensory modality such as auditory and vibrotactile feedback. These guide cues typically map path deviations to directional feedback conveyed through earphones or wearable tactile devices. Currently, most existing systems [14], [15], [23] adopt reactive strategies, primarily due to two challenges: the inability to directly control user behavior and the absence of an accurate model of human walking dynamics, which hinders predictive planning. For example, Lee et al. [15] designed a system that issues vibrotactile cues when a user's lateral deviation exceeds a predefined threshold. Similarly, Li et al. [23] designed a wearable navigation system that only give auditory cues when the distance to an obstacle or a turning point in global navigation is less than the threshold. However, reactive strategies often lead to excessive cognitive load. When using systems that rely on fixed command-triggering thresholds, users might receive redundant commands even when naturally realigning their path, resulting in repeated commands that can confuse users or lead to overcompensation. To tackle the drawbacks of reactive strategies, only limited preliminary exploration has been conducted. For example, PING [13] models user-specific walking dynamics to generate more anticipatory guidance. However, it relies per-user training data, leading to potential cold-start issues. In contrast, our approach offers a generalizable predictive framework without user-specific adaptation, which has been deployed and tested in real-world scenarios.

III. PROBLEM FORMULATION

Path following for visually impaired is the task of generating discrete guidance commands that enable the user to safely and efficiently follow a planned path. The goal is to keep the user aligned with the planned path while avoiding obstacles and minimizing cognitive load. In this paper, we model the interaction between the navigation system and the user as a Markov Decision Process (MDP), defined by the tuple $(\mathcal{S}, \mathcal{A}, p, r, \gamma)$. Here, $\mathcal{S}$ denotes the user state space. A state of user is defined as $(\mathbf{p}, \mathbf{v}, \theta, \omega)$, where $\mathbf{p} \in \mathbb{R}^3$ denotes the 3-D position, $\mathbf{v} \in \mathbb{R}^3$ denotes the linear velocity, θ represents the orientation and ω represents the angular velocity of the user. $\mathcal{A}$ denotes the system action space (e.g. guidance commands including turn left, turn right and go straight). The dynamics $p(s_{t+1}|s_t, a_t)$ describe the underlying process by which the user evolves from state s_t to the next state s_{t+1} in response to the given action a_t. The reward function $r(s_t, a_t)$ is designed to encourage consistency with planned path while penalizing deviations, excessive instructions and potential collision. γ is the discount factor, which quantifies the

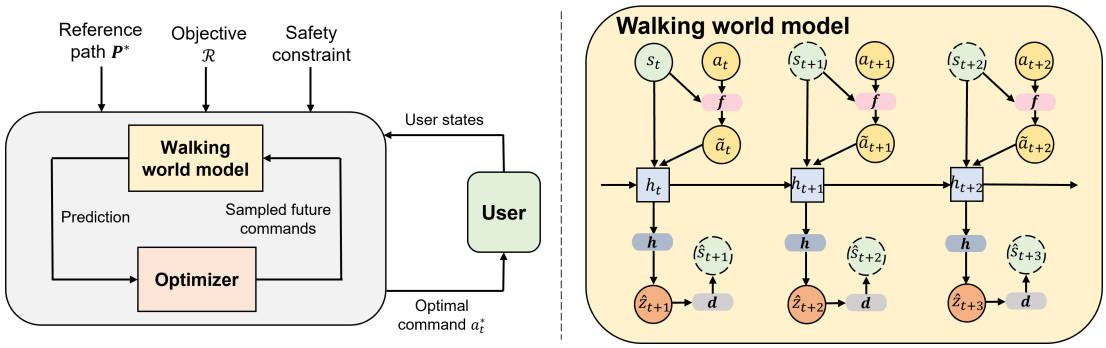


Fig. 1: Predictive path-following with a learned walking world model. **Left:** An MPC loop samples command sequences, scores them with the world model under objective $\mathcal{R}$ with satisfying safety constraints, closes the loop with the user's current state, and issues the first action of the best sequence. **Right:** The workflow of the world model in use: The world model maintains a recurrent latent state h_t to encode the historical information; the transition h advances the latent, and the decoder d reconstructs the state for the next-step prediction.

diminishing importance of future rewards. The optimization goal is to learn a policy $\pi_t = \pi(a_t|s_t)$ that maximizes the expected cumulative reward over a finite planning horizon H, formulated as $\sum_{t'}^{t+H-1} \gamma^{t'-t} r(s_{t'}, a_{t'})$. If the the dynamics p of the user can be acquired directly, it is easy to learn the optimal policy π by optimization methods like dynamic programming. However, the transition dynamics p are unknown a priori. Therefore, accurately estimating the dynamics of users becomes the primary objective of this work, as it enables the system to simulate user behavior and thereby optimize policies.

IV. METHOD

The overview of our proposed predictive path-following approach is illustrated in Fig. 1. In this section, we outline the methodology behind the proposed predictive path-following approach. To accurately model the dynamics of VI and ensure generalization, we first pre-train the walking world model using a large action-free dataset (detailed in Section IV-A), and then finetune it using small action-annotated dataset (detailed in Section IV-B). Finally, leveraging the walking world model, an MPC planner is utilized to search the optimal guidance commands (detailed in Section IV-C).

A. World Model Pre-training from Action-Free Walking Data

Since the world model estimates user dynamics under specified actions, its training relies on annotated state–action pairs. While offering precise supervision, such data are costly and time-consuming to collect at scale. In contrast, action-free data—captured as users move naturally without guidance—are easier to obtain. To initialize the world model with broad locomotion priors, we first pre-train the world model using large-scale action-free data.

The usefulness of action-free data comes from the fact that even without external guidance, the movements of users are still purposeful. This is due to *the sense of agency*—a cognitive process where individuals feel in control of own actions, guided by internal goals and real-time sensory feedback [24]. As a result, free walking behavior naturally reflects meaningful motion patterns that can be exploited to learn generalizable walking dynamics.

In this context, the world model refers to a predictive model that approximates the walking dynamics of the user. Formally, it learns a mapping from the current state and action to the next state. However, there is no explicit actions in free walking data, which hinders direct training. So During pre-training, a single-layer bidirectional GRU g is used as the action extractor to infer latent actions between adjacent states:

$$\{a_1^*, a_2^*, a_3^*, ..., a_{T-1}^*\} = g(s_1, s_2, s_3, ..., s_T). \tag{1}$$

After acquiring latent action, the next step is the dynamics prediction process by the world model. In this paper, we adopt an Recurrent State-Space Model (RSSM) [25] with deterministic states as the world model's backbone.

The world model $\hat{p}$ is composed of four components—a recurrent model, a latent transition model, a state decoder and a representation model—defined as follows:

$$\begin{aligned} &\text{Recurrent model:} && h_{t+1} = r(h_t, z_t, a_t^*), \\ &\text{Latent transition:} && \hat{z}_{t+1} = h(h_{t+1}), \\ &\text{State decoder:} && \hat{s}_{t+1} = d(\hat{z}_{t+1}), \\ &\text{Representation:} && z_{t+1} = q(h_{t+1}, s_{t+1}). \end{aligned} \tag{2}$$

Here, the hidden state is updated recurrently by recurrent backbone $r(\cdot)$. The latent transition model $h(\cdot)$ predicts the next hidden state $\hat{z}_t$ purely from the previous information. The state decoder $d(\cdot)$ maps the hidden state back to an observable state, and the representation model $q(\cdot)$ infers a latent state z_t that is consistent with the observed state s_t and its dynamic context, which acts as the ground truth of $\hat{z}_t$.

All components of the world model and the action extractor are optimized jointly: the output of latent transition model and the state decoder are both trained to approximate their respective ground-truth values. Specifically, we utilize a joint L2 loss as the optimization objective:

$$L_{pre-train} = \|\hat{z}_t - z_t\|_2^2 + \alpha\|s_t - d(z_t)\|_2^2. \tag{3}$$

In the above loss function, the first term encourages consistency between hidden states inferred with and without seeing the next state. The second term aims to reduce the discrepancy between observed state and the state reconstructed from the hidden state in order to learn a map from hidden state space

to observable state space. α is the weight of the reconstruction term, controlling the trade-off between the two objectives.

B. World Model Fine-tuning from Action-Annotated Walking Data

Despite pre-training the world model on action-free data, the world model still cannot predict trajectories conditioned on guidance commands. Thus, we fine-tune the model using action-annotated data to enable action-conditioned trajectory prediction. The action-annotated dataset includes walking trajectories from four users responding to commands.

To leverage the latent dynamics learned during pre-training, explicit actions in the annotated data should be translated into the same latent action space the model has already learned. In pre-training, the model infers a latent action $\widetilde{a}_t$. However, in annotated data, we have explicit actions a_t paired with each state s_t. To align the two, we introduce a 2-layer MLP action adapter f to map each (s_t, a_t) pair into the latent action space:

$$\widetilde{a}_t = f(s_t, a_t). \tag{4}$$

After obtaining the action representation from action adapter, we proceed to fine-tune the world model using action-annotated data based on the pre-trained model. The overall fine-tuning stage is summarized as below:

$$\begin{aligned} &\text{Recurrent model:} && h_{t+1} = r(h_t, z_t, \widetilde{a}_t), \\ &\text{Latent transition:} && \hat{z}_{t+1} = h(h_{t+1}), \\ &\text{State decoder:} && \hat{s}_{t+1} = d(\hat{z}_{t+1}), \\ &\text{Representation:} && z_{t+1} = q(h_{t+1}, s_{t+1}). \end{aligned} \tag{5}$$

In fine-tuning stage, L2 normalization are applied to prevent overfitting. Similar to the pre-training stage, the loss function in fine-tuning stage is expressed as follows:

$$L_{fine-tune} = \|\hat{z}_t - z_t\|_2^2 + \alpha\|s_t - d(z_t)\|_2^2 + \lambda\|\mathbf{w}\|_2^2. \tag{6}$$

In the above loss function, $\|\mathbf{w}\|_2$ is the 2-norm of the model. λ represents the weight of the L2 normalization term.

C. Guidance Commands Generation by MPC

After fine-tuning, we acquire a predictive world model $\hat{p}$, acting as an estimation of user's dynamics p defined in Section III. Once the world model is ready, we employ Model Predictive Control (MPC) to search the optimal guidance command. MPC is a control method that enables goal-directed planning by iteratively optimizing future action sequences based on a dynamics model, as shown in Fig. 1. It has two essential components: a dynamics model and an optimization procedure involving an optimization object and constrains. However, standard MPC formulations primarily optimize tracking accuracy, control effort, and endpoint accuracy. For mechanical systems, such as cars typically support high control frequencies, the above factors are sufficient to achieve satisfactory control results. However, when assisting humans, frequent control signals may impose considerable cognitive load. To mitigate this, it becomes necessary to incorporate a cognitive load cost, which explicitly penalizes excessive control frequency. Therefore, in our approach, we formulate an optimization problem considering path-following task with a cognitive load cost to select a sequence of discrete actions $\{a_t, ..., a_{t+H-1}\}$ that maximizes a multi-objective reward over a planning horizon H as follows:

$$\begin{aligned} \max_{\{a_t,\dots,a_{t+H-1}\}} \mathcal{R} = &\sum_{k=1}^{H-1} \left(-\|\mathbf{p}_{t+k} - \mathrm{c}(\mathbf{p}_{t+k}, \mathbf{P}^*)\|_2^2 - \omega_1\|a_{t+k}\|_2^2\right) \\ &- \omega_2\|\mathbf{p}_{t+H} - \mathbf{p}_n^*\|_2^2 + \omega_3 \sum_{k=1}^{H} \frac{1}{\mathrm{e}(\mathbf{p}_{t+k}, \Omega)} \quad (7a) \\ \text{s.t.} \quad &\hat{s}_{t+1} = \hat{p}(s_t, a_t), \quad (7b) \\ &\hat{s}_{t+k} = \hat{p}(\hat{s}_{t+k-1}, a_{t+k-1}), \forall k \in [2, H], \quad (7c) \\ &\mathbf{p}_{t+k} = \mathcal{T}_p \hat{s}_{t+k}, \mathcal{T}_p = [\mathbf{I}_3 \ \ \mathbf{0}], \forall k \in [1, H], \quad (7d) \\ &\mathbf{P}^* = \{\mathbf{p}_1^*, \mathbf{p}_2^*, ..., \mathbf{p}_n^*\}, \quad (7e) \\ &\mathrm{c}(\mathbf{p}, \mathbf{P}^*) = \arg\min_{\mathbf{p}^* \in \mathbf{P}^*} \|\mathbf{p} - \mathbf{p}^*\|_2^2, \quad (7f) \\ &\mathrm{e}(\mathbf{p}, \Omega) = \min_{\mathbf{q} \in \Omega} \|\mathbf{p} - \mathbf{q}\|_2^2, \quad (7g) \\ &\mathrm{L}(\mathbf{p}_{t+k-1}, \mathbf{p}_{t+k}, \Omega) = \text{True}, \forall k \in [1, H], \quad (7h) \\ &a_{t+k} \in \{-1, 0, 1\}, \forall k \in [1, H]. \quad (7i) \end{aligned}$$

This formulation in (7) integrates path following, control frequency, goal reaching, and safety. The object $\mathcal{R}$ in (7a) is the optimization goal. The first term of $\mathcal{R}$ minimizes the distance between predicted positions $\mathbf{p}_{t+k}$—obtained via the projection $\mathcal{T}_p$ in (7d)—and the closest point in the target path $\mathbf{P}^*$, where the closest point is determined by minimizing the Euclidean distance described by (7f). The second term of $\mathcal{R}$ is the cost of control frequency, this term is designed for reducing cognitive load. The third term of $\mathcal{R}$ encourages the user reaching the final goal $\mathbf{p}_n^*$ by the end of the planning horizon, and the fourth term improves safety by favoring trajectories that maintain greater distance from obstacles in obstacle set Ω, using the inverse-distance measure in (7g). Future states of user are recursively predicted using the learned world model, as described in (7b) and (7c). Moreover, safety constraint of the predicted trajectories are ensured by enforcing collision-free transitions between positions, as described by (7h). The actions (i.e. guidance commands) defined in (7i) are constrained to a discrete set {-1, 0, 1}, representing *turn left*, *go straight*, and *turn right*, respectively.

Owing to the discrete nature of the action space, we employ random sampling methods to efficiently solve the optimization problem. At each timestep, the optimization yields an optimal action sequence. Only the first action of the optimal action sequence a_t^* is executed as the current guidance command. The process then repeats in a receding-horizon fashion, allowing online adaptations to the changing environment.

V. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we conducted experiments to study whether the predictive path-following approach and the training paradigm of the world model we proposed can address the following two question:

– Does our proposed predictive path-following approach improve the walking speed of users and reduce their cognitive

load without compromising safety performance compared to previous reactive navigation approach?
– Does leveraging action-free walking data through a two-stage training scheme improve the prediction accuracy and generalization to unseen users?

Datasets for training. We train the walking world model on two datasets. For *pre-training*, we collected over 10 h of free walking from 5 eye-masked and 8 VI participants across four routes, with participants walking with a white cane and our device (for recording data). For *Fine-tuning*, 3 eye-masked and 1 VI participant were asked to walk in different open squares respectively under vibrotactile guidance, performing normal walking and rapid turns on command. Each of them contributed 1 h of annotated data. Poses, velocities, angular velocities and headings were estimated with VINS-Mono [26] and recorded at 1 Hz, yielding more than 15k samples (approximately 1.1k instances each of "turn left" and "turn right" commands). Although the dataset is moderately imbalanced across command types, simple per-frame resampling or reweighting brought no clear benefit in our preliminary tests, likely because user motion patterns are strongly time-coupled. We therefore train on the natural sequence distribution. The model predicts 5 s ahead, conditioned on 10 s of past observations/actions plus the planned actions for the next 5 s.

Fig. 2: Hardware configuration of the navigation system.

Platform. As shown in Fig. 2, the hardware configuration follows the architecture presented by a prior work [27], which comprises the three core components of a navigation system. An Intel Realsense RGB-D camera is employed for perception, and calculation is performed on a mini PC, which processes the sensor input and plans a target path. For the user interface, a haptic belt is utilized to deliver guidance commands to the user via vibrations. As for the software design, we adopted Mask2Former [28] for semantic segmentation. The obstacle set are obtained by fusing segmentation with depth. RGB images are segmented with Mask2Former, lifted via depth into semantic point clouds, and integrated into a 3D voxel map, which is compressed to a 2D obstacle map. Then, A* is applied for path planning. The path-following method used in the system are introduced in Section V-A in detail. On the mini PC's CPU, the world model itself can run up to 30 Hz, the optimization reaches a maximum frequency of 1.8 Hz, and the command release frequency is set to 1 Hz, which was primarily determined by users' gait frequency and information-processing capacity; it is not globally optimal and can be tuned to meet specific user needs if needed.

Experiment Setting. To evaluate the effectiveness of the proposed path-following approach, we carried out a series of experiments. Our path-following approach was first integrated into a navigation system. Evaluation was then performed under two distinct settings. In the first setting, we tested our approach on a static map, where the target path was fixed and known in advance. This allowed us to assess the performance of our approach under idealized conditions. In contrast, the second setting involved a dynamic real-world environment in which both the surroundings and the target path changed dynamically, providing a more realistic and challenging setting. Moreover, to verify the effectiveness of two-stage strategy for training the world model, we investigate whether the two-stage training strategy yields improved performance on users not seen during training by designing an ablation experiment.

Participants. Eight VI participants (6Male/2Female; 20–56 years; 4 congenital, 4 adventitious), all long-term white-cane users ($>$ 10 years), and eight eye-masked participants (6Male/2Female; 20–30 years) took part in our experiment. The eye-masked group can quickly learn to use the white cane and the navigation system and was included to assess path-following performance in short-term lost sight scenarios.

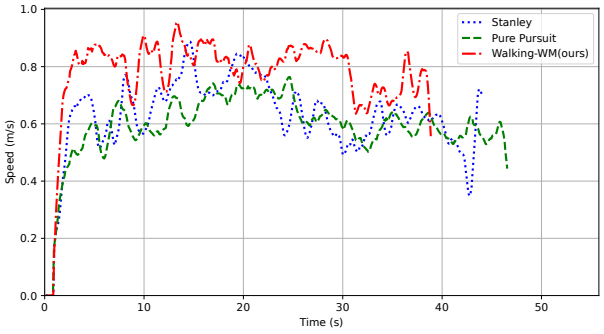
A. Evaluation of Path-Following under Static Map

We first test our approach in static scenario, where the target path on a static map remains constant. The static map and target path is shown in Fig. 3b. The target path is 25.5 meters long and includes two 90-degree turns in different directions.

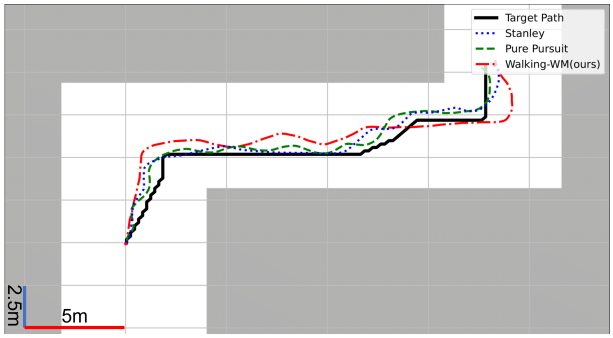
For baseline comparison, two geometry-based path-following methods were selected—Pure Pursuit [29] and Stanley [30]—because our haptic belt supports only binary vibration cues. Pure Pursuit uses a single look-ahead parameter to detect when a turn is needed, and issue guidance command at once, while Stanley incorporates both heading and cross-track errors for more aggressive lateral correction. Both algorithms have been proven in various robotic platforms, making them the most practical choices for guiding VI with current hardware configuration. In the experiment, both methods were adapted to trigger a discrete vibration when the user's steering deviation exceeds a predefined threshold.

During experiment, to ensure a fair comparison, at the beginning of each test session, participants receive a 10-minute tutorial that introduces the system and instructs them on how to respond to the commands. Participants were asked to walk at their usual walking speed, steering left and right according to the vibrotactile commands. The test session reach a termination when collision happens or the position of the user is within 1 meter away from the termination point. To avoid bias from route memorization due to repetition, we use a Latin square design and randomly mirror the scene layouts during experiments.

We recorded each participant's trajectory and completion time. As shown in Table I, four VI (V1-V4) and four eye-masked (M1-M4) participants took part in the experiment. To prevent user-obstacle collision, reactive path-following



(a) Visualization of V3's velocity trends



(b) Visualization of V3's walking trajectories

Fig. 3: The speed profile and walking trajectories of V3 using our Walking-WM, Pure Pursuit and Stanley, respectively.

TABLE I: Comparison of path-following approaches under static map. **Bold** indicates the best performance for each metric. "EM" stands for the eye-masked, while "TL" stands for the trajectory length.

	Time(s) ↓		Velocity(m/s) ↑		TL(m)	
Method	**EM**	**VI**	**EM**	**VI**	**EM**	**VI**
Stanley	59.7	58.0	0.53	0.51	29.2	28.5
Pure Pursuit	63.4	56.6	0.51	0.50	30.1	27.8
Walking-WM (ours)	**49.3**	**45.1**	**0.65**	**0.66**	31.2	29.5

approaches use a small deviation threshold, resulting in shorter paths but frequent adjustment commands that slow down walking speed. In contrast, with the predictive approach based on our Walking-WM, both VI and eye-masked users completed the path faster, with walking speed improved by over 20% (0.51→0.66, 0.53→0.65) due to fewer unnecessary corrections.

We analyzed V3's experimental results. Figure 3a shows the speed profiles, and Figure 3b the walking trajectories for the three path-following methods. As shown in Figure 3a, V3 walked faster with our Walking-WM approach. To quantify fluctuations, we used the coefficient of variation (CV), a unit-less measure where lower values indicate steadier speed. Walking-WM achieved 0.12, compared to 0.14 for Pure Pursuit and 0.17 for Stanley, demonstrating the most stable performance, and this phenomenon is consistently observed across all test users. Table I shows longer trajectories with Walking-WM, due to smoother, larger-radius turns. Although paths were longer, these trajectories helped maintain velocity and reduced braking and re-acceleration. Similar results were observed among other experimental users.

Moreover, to assess the impact of dynamics accuracy in predictive path following, we replaced our learned walking world model with a hand-crafted hybrid that assumes fixed user behavior for turning and going straight while keeping the same optimization objective. As a result, its inaccurate predictions often misclassify genuinely safe, low-cognitive-load command sequences as collision-prone, causing the system to favor higher-frequency commands it predicts to be safer, especially as the planning horizon grows, indicating that efficient VI path following requires both an accurate user dynamics model and a carefully chosen planning horizon.

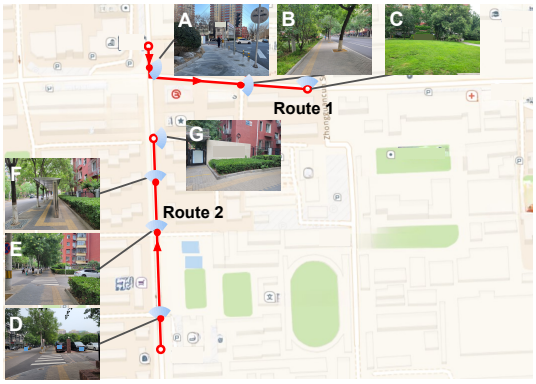


Fig. 4: Layouts of Route 1 and Route 2, with key street views marked along each path.

B. Evaluation of Path-Following in Dynamic Real-World

To further evaluate the effectiveness of our system beyond controlled settings, we conducted additional experiments in dynamic real-world scenarios, which introduces more complexity and uncertainty such as moving obstacles. In this testing scenario, two daily-life routes of VI were selected. The first route has a total length of 240 meters, involving a 90-degree turn and entirely on sidewalks with few moving obstacles. The second route is about 300 m, with three tactile paving interruptions and more moving obstacles like vehicles and pedestrians. Participants were asked to walk from a start point to a target under varying conditions until they are close enough to the target point. Eight people joined the tests (four VI and four eye-masked, labeled V5–V8 and M5–M8).

None of the participants had walked on either of the two routes prior to the experiment.

In real-world testing, all of the VI and eye-masked users were asked to walk along the two routes under 3 conditions:
(1) White Cane and Navigation app (Cane+App);
(2) Navigation system employing Pure Pursuit (System-PP).
(3) Navigation system employing the Walking-WM algorithm (System-WM).

A trail was considered complete when the participant was within 10 meters of the target point. Latin square design was applied to avoid bias from route memorization.

As shown in Table II, compared to using white canes to walk along an unfamiliar route and relying on a navigation

TABLE II: Performance of users on two routes. Each cell reports Route 1 | Route 2. **Bold** indicates the best performance for each metric.

	Velocity(m/s) ↑		**Collision**$(/trial)$ ↓	
Method	**EM**	**VI**	**EM**	**VI**
Cane+App	0.62|0.62	**0.74**|0.71	1.00|1.50	0.75|0.75
System-PP	0.53|0.55	0.58|0.54	0.50|0.75	**0.25**|0.75
System-WM(ours)	**0.67**|**0.70**	**0.74**|**0.73**	**0.25**|**0.25**	**0.25**|**0.50**

app, the usage of the navigation system reduces collisions. This is because the system provides real-time guidance based on the surroundings. For comparison between System-WM and System-PP, System-WM leads to fewer collisions. This improvement is attributed to the system's ability to anticipate the user's future trajectory and issue early avoidance commands. However, occasional misjudgments—such as semantic segmentation errors—can still result in collisions when the system is used alone. These errors might be mitigated when a white cane is also used.

Regarding walking speed, VI users moved faster with a white cane than eye-masked users. System-PP tended to confuse users with frequent, successive commands, often causing them to pause and reorient. In contrast, System-WM allowed users to maintain walking speeds comparable to those achieved with a white cane and even faster, which may stem from increased user confidence in using the system.

In addition to the objective performance metrics, subjective evaluation was also carried out to better understand users' perceptions. Participants were asked to fill out a questionnaire assessing their experience with different navigation strategies. The evaluation focused on three key dimensions: safety, cognitive load, and overall helpfulness. Scores the users can give range from 1 to 7, with 7 representing very safe, very low cognitive load, and very helpful, respectively. As seen in Table III, users generally reported a significant reduction in cognitive load after adopting our proposed method. The cognitive ease score increased from 3.6 to 5.6. Additionally, due to "anchor effect" can be caused by Likert scale evaluation [31], we ran an additional SWORD pairwise workload comparison [32] as a supplement. Instead of absolute ratings, participants made relative judgments of cognitive demand between the two methods. We used a 5-point symmetric scale (-2: much less demanding; +2: much more demanding). The results in Table IV confirm that users experienced System-WM as significantly less demanding, reinforcing our earlier finding that our proposed method can reduce cognitive load of users. In a small follow-up study on an additional static route, we also collected NASA-TLX [33] workload ratings. Although conducted in a slightly different setting, these results showed the same trend as our Likert-scale and SWORD analyses, with our method perceived as less demanding than the baselines. For brevity, we omit the full protocol and scores here.

C. Evaluation of Generalization of the World Models

As introduced in Section IV, we propose a two-stage training strategy for the walking world model. To evaluate both (i) what the learned model adds beyond basic kinematic assumptions and (ii) whether the two-stage training improves generalization, we consider three approaches: (1) *Hybrid model*, a hand-crafted dynamics model adapted from the Bicycle Model [34] with discrete "turn left / go straight / turn right" commands, assuming constant speed, a fixed steering angle and a fixed turning duration, similar as the Hybrid Model used in Section V-B; (2) *Walking-WM (w/o pre-train)*, where the world model is trained from randomly initialized weights using only action-annotated data; and (3) *Walking-WM (w pre-train)*, where the model is first pre-trained on large-scale action-free walking data and then fine-tuned on action-annotated data. We adopt a leave-one-out strategy over users during fine-tuning and report prediction accuracy using Average Displacement Error (ADE) and Final Displacement Error (FDE).

TABLE III: Subjective ratings on a 7-point Likert scale (higher is better). Values are mean ± std.

Metric	Pure-Pursuit	Walking-WM
Safety ↑	3.88 ± 1.25	5.63 ± 0.74
Cognitive Ease ↑	3.63 ± 1.69	5.63 ± 1.06
Helpfulness ↑	4.13 ± 1.13	5.63 ± 0.92

TABLE IV: Dominance scores of workload, with negative indicating less demanding and positive indicating more demanding.

Participant	V5	V6	V7	V8	M5	M6	M7	M8
Dominance scores	-1	-1	0	-2	-1	-2	-2	-1

TABLE V: Comparison of generalization across different approaches. Results are reported as mean ± standard deviation over four users.

	ADE ↓	FDE ↓
Hybrid Model	0.91 ± 0.10	1.69 ± 0.21
Walking-WM(w/o pre-train)	0.44 ± 0.04	0.82 ± 0.09
Walking-WM(w pre-train)	0.39 ± 0.03	0.73 ± 0.07

Table V presents a comparison of all approaches. The *Hybrid Model* baseline yields the highest prediction errors, indicating its inability to capture complex user dynamics. In contrast, learning-based world model achieve significantly lower ADE and FDE, indicating that the learned latent dynamics capture unique dynamic characteristics of human walking that are not encoded in the basic model. Meanwhile, compared to training from scratch, training based on pre-trained model reduces the Average Displacement Error (ADE) by 9.2% and the Final Displacement Error (FDE) by 10.9%, demonstrating that incorporating large-scale unannotated data during pre-training leads to improved generalization on unseen users.

VI. CONCLUSIONS AND FUTURE WORKS

We propose a predictive navigation approach for VI users that integrates a walking world model with MPC to proactively guide users via vibrotactile feedback. By modeling walking dynamics and predicting user trajectories, our method reduces unnecessary instructions and improves walking efficiency. The walking world model is trained in two stages—pre-trained on large-scale action-free data to capture general gait patterns

and then fine-tuned on action-annotated data for instruction-conditioned trajectories—and combined with an MPC planner that penalizes path deviation, goal error, safety risk, and cognitive load to generate optimized command sequences. Extensive real-world tests show improved walking performance and reduced cognitive load compared to baseline methods.

Although our path-following method demonstrated effectiveness in testing, There is still room for improvement. Future work mainly concentrates on two key areas. First, we aim to improve the world model's generalization through more diverse unannotated data. Second, we plan to validate the system in more complex dynamic environments.

REFERENCES

[1] W. Jeamwatthanachai, M. Wald, and G. Wills, "Indoor navigation by blind people: Behaviors and challenges in unfamiliar spaces and buildings," *British Journal of Visual Impairment*, vol. 37, no. 2, pp. 140–153, 2019.

[2] A. J. Ramadhan, "Wearable smart system for visually impaired people," *Sensors*, vol. 18, no. 3, p. 843, 2018.

[3] A. L. Fraga, X. Yu, W.-J. Yi *et al.*, "Indoor navigation system for visually impaired people using computer vision," in *International Conference on Electro Information Technology*, 2022, pp. 257–260.

[4] M. Kuribayashi, T. Ishihara, D. Sato *et al.*, "Pathfinder: Designing a map-less navigation system for blind people in unfamiliar buildings," in *Proceedings of the Conference on Human Factors in Computing Systems*, 2023, pp. 1–16.

[5] H. Zhang, N. J. Falletta, J. Xie *et al.*, "Enhancing the travel experience for people with visual impairments through multimodal interaction: Navigpt, a real-time ai-driven mobile navigation system," in *Companion Proceedings of the ACM International Conference on Supporting Group Work (GROUP)*, 2025, pp. 29–35.

[6] T. Hermann, A. Hunt, J. G. Neuhoff *et al.*, *The sonification handbook*, 2011, vol. 1.

[7] P. Balatti, I. Ozdamar, D. Sirintuna *et al.*, "Robot-assisted navigation for visually impaired through adaptive impedance and path planning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 2310–2316.

[8] S. Liu, A. Hasan, K. Hong *et al.*, "Dragon: A dialogue-based robot for assistive navigation with visual language grounding," *IEEE Robotics and Automation Letters (RA-L)*, vol. 9, no. 4, pp. 3712–3719, 2024.

[9] P. Slade, A. Tambe, and M. J. Kochenderfer, "Multimodal sensing and intuitive steering assistance improve navigation and mobility for people with impaired vision," *Science Robotics*, vol. 6, no. 59, p. eabg6594, 2021.

[10] J. Guerreiro, D. Sato, S. Asakawa *et al.*, "Cabot: Designing and evaluating an autonomous navigation robot for blind people," in *Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility*, 2019, pp. 68–82.

[11] P.-J. Duh, Y.-C. Sung, L.-Y. F. Chiang *et al.*, "V-eye: A vision-based navigation system for the visually impaired," *IEEE Transactions on Multimedia (T-MM)*, vol. 23, pp. 1567–1580, 2020.

[12] Z. Yuan, T. Zhang, Y. Deng *et al.*, "Walkvlm: Aid visually impaired people walking by vision language model," *arXiv preprint arXiv:2412.20903*, 2024.

[13] E. OhnBar, K. Kitani, and C. Asakawa, "Personalized dynamics models for adaptive assistive navigation systems," in *Conference on Robot Learning (CoRL)*, 2018, pp. 16–39.

[14] H.-C. Wang, R. K. Katzschmann, S. Teng *et al.*, "Enabling independent navigation for visually impaired people through a wearable vision-based feedback system," in *International Conference on Robotics and Automation (ICRA)*, 2017, pp. 6533–6540.

[15] Y. H. Lee and G. Medioni, "Rgb-d camera based wearable navigation system for the visually impaired," *Computer vision and Image understanding (CVIU)*, vol. 149, pp. 3–20, 2016.

[16] Y. Li, W. R. Jeon, and C. S. Nam, "Navigation by vibration: Effects of vibrotactile feedback on a navigation task," *International Journal of Industrial Ergonomics*, vol. 46, pp. 76–84, 2015.

[17] G. Flores, S. Kurniawan, R. Manduchi *et al.*, "Vibrotactile guidance for wayfinding of blind walkers," *IEEE Transactions on Haptics*, vol. 8, no. 3, pp. 306–317, 2015.

[18] C. Ye, S. Hong, X. Qian *et al.*, "Co-robotic cane: A new robotic navigation aid for the visually impaired," *IEEE Systems, Man, and Cybernetics Magazine*, vol. 2, no. 2, pp. 33–42, 2016.

[19] J. Borenstein and I. Ulrich, "The guidecane-a computerized travel aid for the active guidance of blind pedestrians," in *International Conference on Robotics and Automation (ICRA)*, vol. 2, 1997, pp. 1283–1288.

[20] I. Ulrich and J. Borenstein, "The guidecane-applying mobile robot technologies to assist the visually impaired," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 31, no. 2, pp. 131–136, 2001.

[21] A. Xiao, W. Tong, L. Yang *et al.*, "Robotic guide dog: Leading a human with leash-guided hybrid physical interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 470–11 476.

[22] M. Tognon, R. Alami, and B. Siciliano, "Physical human-robot interaction with a tethered aerial vehicle: Application to a force-based human guiding problem," *IEEE Transactions on Robotics*, vol. 37, pp. 723–734, 2021.

[23] B. Li, J. P. Munoz, X. Rong *et al.*, "Vision-based mobile indoor assistive navigation aid for blind people," *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 702–714, 2018.

[24] P. Haggard, "Sense of agency in the human brain," *Nature Reviews Neuroscience*, vol. 18, no. 4, pp. 196–207, 2017.

[25] D. Hafner, T. Lillicrap, I. Fischer *et al.*, "Learning latent dynamics for planning from pixels," in *International Conference on Machine Learning (ICML)*, 2019, pp. 2555–2565.

[26] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics (T-RO)*, vol. 34, no. 4, pp. 1004–1020, 2018.

[27] M. Kan, L. Zhang, H. Liang *et al.*, "elabrador: A wearable navigation system for visually impaired individuals," *IEEE Transactions on Automation Science and Engineering (T-ASE)*, vol. 22, pp. 12 228–12 244, 2025.

[28] B. Cheng, I. Misra, A. G. Schwing *et al.*, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1290–1299.

[29] R. C. Coulter, "Implementation of the pure pursuit path tracking algorithm," 1992.

[30] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo *et al.*, "Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing," in *American Control Conference (ACC)*, 2007, pp. 2296–2301.

[31] P. A. Bishop and R. L. Herron, "Use and misuse of the likert item responses and other ordinal measures," *International journal of exercise science*, vol. 8, p. 297, 2015.

[32] S. M. Casner and B. F. Gore, "Measuring and evaluating workload: A primer," *NASA Technical Memorandum*, vol. 216395, p. 2010, 2010.

[33] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*, 1988, vol. 52, pp. 139–183.

[34] J. Kong, M. Pfeiffer, G. Schildbach *et al.*, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE intelligent vehicles symposium (IV)*, 2015, pp. 1094–1099.