

PAVAL: Position-Aware Virtual Agent Locomotion for Assisted Virtual Reality Navigation

Zi-Ming Ye¹, Jun-Long Chen¹, Miao Wang^{1,2*} and Yong-Liang Yang³

¹ State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China
² Peng Cheng Laboratory ³ University of Bath

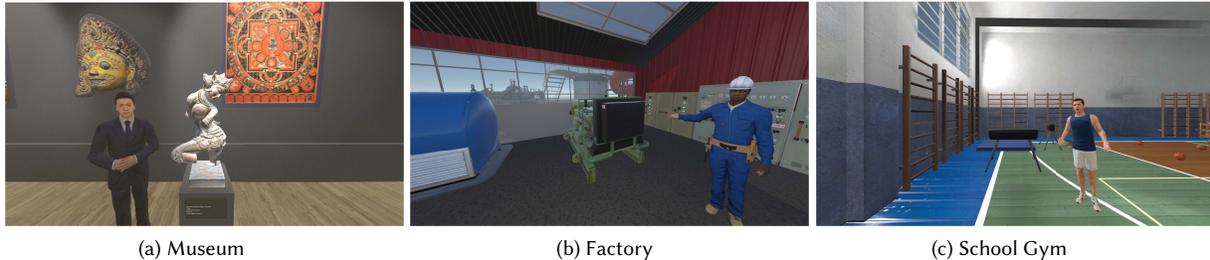


Figure 1: First-person views of user navigation assisted by virtual agents in three representative scenes. The user is (a) watching a statue at a 1-meter distance; (b) learning machine functionality at a 2-meter distance; (c) studying how to play pommel horse at a 3-meter distance. The virtual agent in each scene automatically performs locomotion during user navigation.

ABSTRACT

Virtual agents are typical assistance tools for navigation and interaction in Virtual Reality (VR) tour, training, education, etc. It has been demonstrated that the gaits, gestures, gazes, and positions of virtual agents are major factors that affect the user’s perception and experience for seated and standing VR. In this paper, we present a novel position-aware virtual agent locomotion method, called PAVAL, that can perform virtual agent positioning (position+orientation) in real time for room-scale VR navigation assistance. We first analyze design guidelines for virtual agent locomotion and model the problem using the positions of the user and the surrounding virtual objects. Then we conduct a one-off preliminary study to collect subjective data and present a model for virtual agent positioning prediction with fixed user position. Based on the model, we propose an algorithm to optimize the object of interest, virtual agent position, and virtual agent orientation in sequence for virtual agent locomotion. As a result, during user navigation in a virtual scene, the virtual agent automatically moves in real time and introduces virtual object information to the user. We evaluate PAVAL and two alternative methods via a user study with humanoid virtual agents in various scenes, including virtual museum, factory, and school gym. The results reveal that our method is superior to the baseline condition.

Keywords: Virtual Agent, Navigation, Optimization

1 INTRODUCTION

Many previous works have demonstrated that virtual agents play an important role in users’ overall VR experience. Virtual agents allow the user to perceive a higher degree of social presence [36] and co-presence [31] by promoting spatial interaction through the virtual agent’s virtual body [2]. This is particularly true for guided tours in typical virtual environments such as art galleries and museum exhibitions [5]. For example, Schmidt et al. [31] found that choosing a historical figure as the virtual agent’s representation in a virtual museum effectively promoted users’ sense of co-presence

and social presence, which explains why embodied virtual agents are commonly used in guided tours. Virtual agents can also be extremely helpful in training and rehabilitating scenarios. When there is a shortage of specialized personnel, virtual agents become a good complement to human professionals [24].

Recently, studies have focused on modeling the perceptual emotion [27] and friendliness [26], considering virtual agent behavior such as gaze, gesture and gait. Lang et al. [20] studied the positioning of virtual agents in mixed reality to enable natural interaction with the user. In their studies, the user’s position is fixed in the context of seated or standing VR. To the best of our knowledge, there exists little research on virtual agent locomotion in an adaptive manner (i.e., depending on the user’s real-time positioning) for room-scale VR navigation.

In this paper, we address the problem of automatic virtual agent locomotion generation in the context of room-scale VR navigation, where the user navigates in an virtual scene assisted by a humanoid virtual agent. The agent’s task is to accompany the user and provide verbal introductions of objects of interest to the user. Our goal is to generate real-time virtual agent locomotion according to the positions of the user and virtual objects in the virtual scene. After investigating design guidelines, we conduct a preliminary study to determine a data-driven model that predicts the virtual agent’s location and orientation given the spatial relation between the user and an object of interest. Once determined, the models are independent of virtual scenes and can be reused. Based on the model, we present Position-Aware Virtual Agent Locomotion (PAVAL) which optimizes and updates the agent’s position and orientation for free user navigation in real time. During user navigation, the virtual agent either accompanies the user, or introduces the information of a user-interested virtual object with the optimized location and orientation.

In summary, the major contributions of our work include:

- Proposal of design guidelines for generating virtual agent locomotion for assisted VR navigation.
- Determination of data-driven virtual agent position and orientation prediction models for fixed user position.
- Devising a novel real-time algorithm to dynamically optimize virtual agent locomotion for free user navigation.

*Corresponding author, e-mail: miaow@buaa.edu.cn

- Conducting a user study to evaluate and validate the proposed method with diverse virtual scenes.

The remainder of this paper is structured as follows: Section 2 reviews related work. We investigate design guidelines for virtual agent locomotion in Section 3. Section 4 presents and discusses our data-driven model that optimizes virtual agent position and orientation for fixed user position. The proposed PAVAL method for free user navigation is presented in Section 5. We report the experiments and corresponding results in Section 6 and Section 7, and discuss them in Section 8. The work is concluded in Section 9.

2 RELATED WORK

2.1 Digital Guides and Human Guides

According to previous studies, digital guides have the advantages of being available “on demand” for tourists [10], being easy to manage [34], being the development trend for museums of the 21st century [37], and being capable of delivering context-aware personalized tours [6, 10]. It is hence important to acknowledge the differences between digital guides and human guides to help us design better virtual agents for navigation assistance.

For human guides, they know how to structure the navigation according to space and audience differences. During the navigation, they learn the interests and hobbies of the audience and use their location to decide what to talk about and how to act. A skilled guide is also capable of using gazes to direct audiences and prompt their actions. These are basic skills for a human guide but may be considered difficult in assisted VR navigation. For virtual agents, they need to efficiently guide their audience around spaces and between objects, which is similar to placing a virtual camera in the scene [7, 22]. In this problem, the avatar (e.g. a virtual camera) has the user in the field of view, and satisfies constraints regarding the object of interest. The problem we address is similar because the virtual agent and object of interest are visible to the user and some rules are respected as well. In our study, due attention is paid to the interaction between the virtual agent and the audience, and our paper focuses on the locomotion of the virtual agent.

2.2 Virtual Agent Interaction

While designing interactive agents in virtual reality, it is essential to ensure that it interacts with the user properly. Cohen’s tour guide study [8] summarized two key roles of a modern tourist guide: a *path nder* (to lead the audience through an unfamiliar environment), and a *mentor* (to provide information to the audience). Apart from these two main roles, guides have also developed two new roles of being the *animator* (to motivate the audience to engage in social activities) and *tour-leader* (to be responsible for *interactive* activities).

Similar principles apply to our design of virtual agents in guided virtual navigation. That is to say, adequate interaction should be involved between the user and the guide. Best et al. [3] emphasized that *interactions* play a crucial role in guided tours. Tour guides should encourage their audience to participate and engage in the tour other than simply delivering a didactic speech. Another interesting fact pointed out by the authors is that the term “*audience*” can be misleading: people joining the tour should not be merely passive listeners but should instead take a more active role in interacting with others. It is important to point out that audiences in tours are learners and should be distinguished from those in a show or concert, who are only passive recipients of certain information. Through active engagement, they as well, can shape the navigation.

To promote the interaction between the user and virtual agent, Randhavane et al. [26] developed a friendliness model to generate non-verbal cues such as gaits, gestures, and gazes of virtual agents using a Behavioral Finite State Machine (BFSM). Apart from these non-verbal cues, another way to improve the friendliness of virtual agents, hence promoting interactions with the user is to optimize

the positioning of virtual agents, which include their *position* and *orientation* with respect to the user and her/his current interests. Here, we investigated some works specific to user-centric spatial orientation. Kendon et al. [18] analyzed various kinds of socializing phenomena, and concluded that there exist several interaction spatial formations, such as the L-shaped dyadic F-formation, the vis-à-vis F-formation, the horseshoe formation, clusters, and “common-focus” gatherings. It is important to note that their findings provide insight into the design of virtual agents in our study, where the position and orientation of virtual agents are based on the aforementioned interaction spatial formations. This will be addressed in further detail in Section 3.

2.3 Virtual Agent Positioning

The goal of virtual agent positioning is to determine the position and orientation of the virtual agent given the virtual environment and the current position of the user, without explicit manual selection operations [38]. As pointed out by many researchers [3, 14, 18], positions play an important role when it comes to face-to-face interactions. For example, the vis-à-vis F-formation can be commonly seen in salutations, while the L-shaped dyadic F-formation is more suitable for friendly conversations [18]. Weissker et al. [39, 40] developed virtual group navigation methods based on the teleportation technique, which requires a user to perform guidance rather than using virtual agents. In navigation assistance, automatic positioning of virtual agents can alleviate the workload of human guides.

Positioning techniques have been studied in many fields, ranging from virtual/augmented/mixed reality to robotics [33]. Liang et al. [21] synthesized virtual pet behaviors in mixed reality. In their work, the positioning of the virtual pet was achieved by understanding the scene context through Mask RCNN. Lang et al. [20] studied virtual agent positioning in mixed reality specifically, where they applied a Markov chain Monte Carlo technique to optimize the cost function based on the scene semantics, which consisted of a spatial term and a visibility term. In robotics, Elber et al. [9] studied the positioning and display of objects in a virtual environment. Akiyama et al. [1] studied multi-agent positioning in the dynamic environment. As the authors pointed out, a common approach to this problem is to use rule-based or numerical-function-based systems or machine learning methods.

An important aspect in positioning the virtual agent in our study is to obtain the scene semantics from where the user stands. An important concept related to this is *affordance* as introduced by Gibson [11]. In the realm of scene semantics, objects should not simply be judged by their appearances. The environment surrounding the user not only has its own *physical* properties, but also *affords* in some sense, just as a chair provides the user with something to sit on. The concept of affordance has inspired researchers in scene semantics [30] to correlate the geometry of objects in a scene with their functionality. Grabner et al. [13] found that affordance detection required much fewer training samples compared with appearance-based object detectors. Given the knowledge of objects and regional functionalities, agents in a virtual 3D environment are expected to be able to automatically generate their behaviors.

In this paper, we propose a novel method to optimize virtual agent locomotion. Our work follows the human-centric guidelines and determines the virtual agent’s location and orientation considering the user’s position within the scene. Different from [20] where scene-specific models are learned for each testing condition, our data-driven virtual agent positioning model is one-off. Besides, our method allows the user to freely navigate in the virtual scene, while the virtual agent locomotion updates in real time.

3 DESIGN GUIDELINES FOR VIRTUAL AGENT LOCOMOTION

We clarify that our goal does not include investigating the optimal gaits, gestures, gazes, or verbal contents for the virtual agent. In-

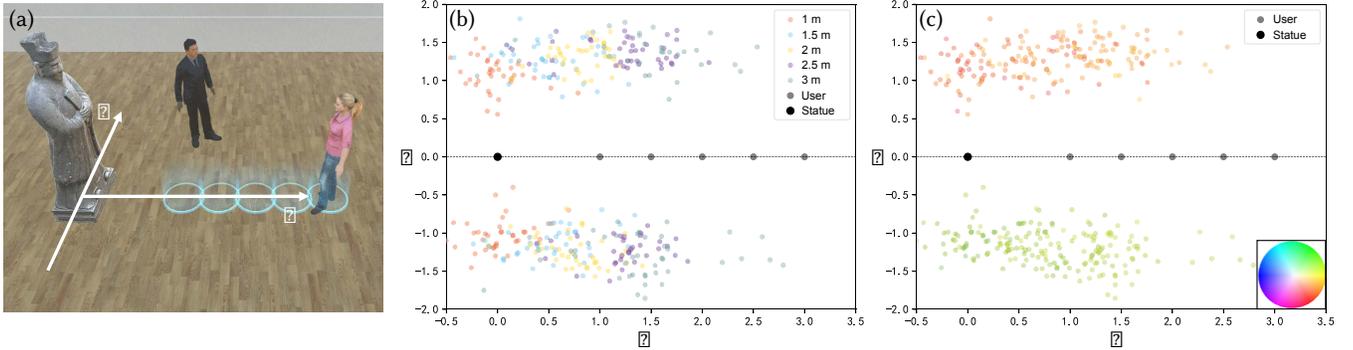


Figure 2: Preliminary study on optimal virtual agent position and orientation with fixed user position. (a) The user (shown as a female avatar only for illustration) stands at discrete positions (cyan circles) in front of a statue, and specifies the optimal position and orientation for the virtual agent (illustrated as a male). (b) The distribution of virtual agent positions and the ownership to trials at corresponding *user-target* distances. (c) The corresponding orientation for each virtual agent position is encoded using a color wheel.

stead, we aim to propose a position-aware virtual agent locomotion technique for assisted VR navigation. This is different from previous work on virtual agent positioning, such as that of Lang et al. [20], in the sense that the virtual agent not only needs to find its best position and orientation given a fixed location of the user in standing VR, but also needs to adaptively move from one place to another at the right time, depending on the user’s route. In prior work [17, 29], the position, route, and action of the virtual agent do not support room-scale VR navigation, which greatly affects the user experience. Motivated by the above observations, we investigate and summarize the following design guidelines for virtual agent locomotion:

DG1: User-centric In our study, we have to keep in mind that we only provide *assisted navigation* in a virtual environment through embodied virtual agents. In other words, the user should be able to explore the Virtual Environment (VE) as they wish, while the virtual agent only assists the navigation as a guide that introduces the virtual scene to the user. When the user is observing a specific object, the virtual guide will find an optimal position and orientation, depending on the user’s current position and orientation. To prevent cybersickness, the virtual agent should move in a continuous way (i.e., without teleportation). The design of the virtual agent also takes the user’s overall experience as its main consideration, hence the term “user-centric.”

DG2: Position first, then orientation According to Kendon et al. [18], there are many spatial formations for interactions, such as the L-shaped dyadic F-formation, the vis-à-vis F-formation, the horseshoe formation, clusters, and ‘common-focus’ gatherings. The formations agree that the virtual agent should be aware of the positions of the user and the surrounding context. In all these cases, the type of formation depends on the surrounding context. Hence, we conclude that for the case of virtual guides, the virtual agent should be aware of the positions of the user as well as information of the virtual scene. Besides, the virtual agent determines its position with a higher priority than setting up its orientation. Therefore, in our virtual agent design, determining the position of the virtual agent has a higher priority than setting its orientation.

DG3: Natural locomotion The locomotion timing and trajectory are expected to be proper and natural. For instance, when the user changes the object of interest and moves to another position, the virtual agent should move at proper timing and avoid hitting obstacles during the locomotion. Moreover, the virtual agent should avoid any unnecessary movement to prevent distraction.

Following the design guidelines, we develop a position-aware virtual agent locomotion method for real-time assisted VR navigation. Our method is user-centric and allows the user to freely navigate in the scene. The virtual agent can verbally introduce the object of interest to the user, considering the user’s position and its relation

to surrounding objects (**DG1**). We resolve the position-aware virtual agent locomotion problem by accomplishing three sub-tasks in sequence: object of interest determination, virtual agent position optimization, and virtual agent orientation optimization (**DG2**). When the virtual agent moves, the naturalness of its locomotion is considered, with the penalty of unnecessary movements and the guarantee of obstacle avoidance (**DG3**).

4 OPTIMAL POSITION AND ORIENTATION PREDICTION FOR FIXED USER POSITION

In this section, we introduce a data-driven model that predicts the virtual agent’s optimal position and orientation for fixed user position. We clarify that the model does not consider any locomotion of the user or the virtual agent, and will serve as a basis for PAVAL.

4.1 Experimental Design

Although there are some suggested formations for social interaction positioning [18], it remains unclear how to accurately deploy such formations under various environment and application conditions. We thus raise three research questions to investigate:

- **RQ1:** What is the optimal position for the virtual agent with a specific *user-target* distance?
- **RQ2:** What is the optimal orientation given the virtual agent’s position?
- **RQ3:** Are there any differences in position/orientation when the virtual agent is on the user’s left-hand and right-hand sides?

In order to respond to the above research questions, we design a preliminary subjective user study based on a simple virtual museum scene. Each participant (with transparent self-avatar) stands at fixed positions, and determines the best virtual agent position and orientation while browsing a statue whose information is introduced by a male virtual agent (see Figure 2a). To represent positions, we set a Cartesian coordinate system, where the origin is the statue and z -axis points to the user. The x -axis is perpendicular to the z -axis. We also use a polar coordinate system to represent orientations where the polar axis is parallel to the z -axis. The participant is asked to stand at discrete distances away from the statue along the z -axis, and manually place the virtual agent at a desirable position (**RQ1**), followed by a desirable orientation adjustment (**RQ2**) using the controller without any constraint. We formulate the virtual agent’s position $p = (x, z)$ as quadratic functions $X(d)$ and $Z(d)$ with respect to the distance d between the user and the statue along the z -axis:

$$X(d) = \lambda_1 \cdot d^2 + \lambda_2 \cdot d + \lambda_3 \quad (1)$$

$$Z(d) = \lambda_4 \cdot d^2 + \lambda_5 \cdot d + \lambda_6, \quad (2)$$

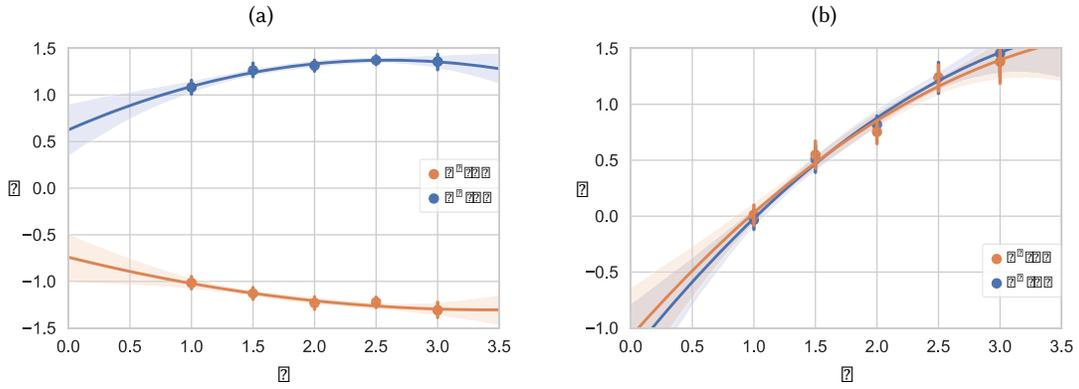


Figure 3: Regressed functions that predict optimal virtual agent position with fixed user positions. Left: left-hand side and right-hand side functions $X^L(d)$ and $X^R(d)$ for x -axis prediction; Right: left-hand side and right-hand side functions $Z^L(d)$ and $Z^R(d)$ for z -axis prediction. The shaded area shows the 95% confidence intervals.

where $\lambda_{1,2,\dots,6}$ are coefficients to be determined via modeling.

When modeling virtual agent's orientation θ , we formulate a polynomial function $\Theta(x, z)$ with respect to the virtual agent's position $p = (x, z)$ as:

$$\Theta(x, z) = \omega_1 \cdot x^2 + \omega_2 \cdot z^2 + \omega_3 \cdot x \cdot z + \omega_4 \cdot x + \omega_5 \cdot z + \omega_6, \quad (3)$$

where $\omega_{1,\dots,6}$ are constant coefficients for the polynomial.

Following **RQ3**, we split the collected data according to the relative position of the virtual agent to the user, and model the left-hand side functions $X^L(d)$, $Z^L(d)$, $\Theta^L(x, z)$ and right-hand side functions $X^R(d)$, $Z^R(d)$, $\Theta^R(x, z)$ separately. We denote $W^L = [\lambda_1^L, \lambda_2^L, \lambda_3^L, \lambda_4^L, \lambda_5^L, \lambda_6^L, \omega_1^L, \omega_2^L, \omega_3^L, \omega_4^L, \omega_5^L, \omega_6^L]$ as the coefficient vector for the left-hand side functions, and W^R similarly as the right-hand side functions. The coefficients W^L and W^R are computed by solving quadratic optimizations using least-squares with bisquare weights, which minimizes the effect of outliers in our data.

4.2 Procedure

We invited 18 participants (12 male, 6 female) from the local university. The participants had normal corrected vision, and were physically and mentally healthy to take part in this study. Each participant wore an *HTC Vive Cosmos* headset with hand-held controllers to perform trials. For each trial, the participant stood in the VE with one of the five discrete distances $d \in \{1m, 1.5m, 2m, 2.5m, 3m\}$ away from the statue, and was asked to set up the virtual agent's position and orientation (see Figure 2a). The participant could move and rotate the virtual agent using controllers until he/she felt comfortable with its positioning. In total, each participant did four trials for each distance in random order, which formed 20 trials (5 distances \times 4 repeats). At the end of the experiment, we let the participant set up the positionings for the virtual agent fixed relative to the statue (invariant to the user's position), or fixed relative to the user (invariant to the statue's position). These additional operations served to provide data for the baseline methods for comparison.

4.3 Results

We collected 360 virtual agent position and orientation data points considering the user and the statue's positions, where 175 were on the left-hand side and 185 on the right-hand side, as shown in Figure 2 (b) and (c). After regression using least-squares with bisquare weights, we obtained the left-hand side coefficients $W^L = [0.0472, -0.3242, -0.7462, -0.1452, 1.2666, -1.0927, 0.1254, -0.0511, 0.0010, -0.4685, -0.2975, 0.5856]$, and the right-hand side coefficients $W^R = [-0.1104, 0.5739, 0.6276, -0.1556, 1.3632, -1.2280, -0.0814, 0.1142, -0.0508, 0.4264, -0.3342, -0.6457]$. We illustrate the regressed functions in Figure 3 and Figure 4. The position prediction functions

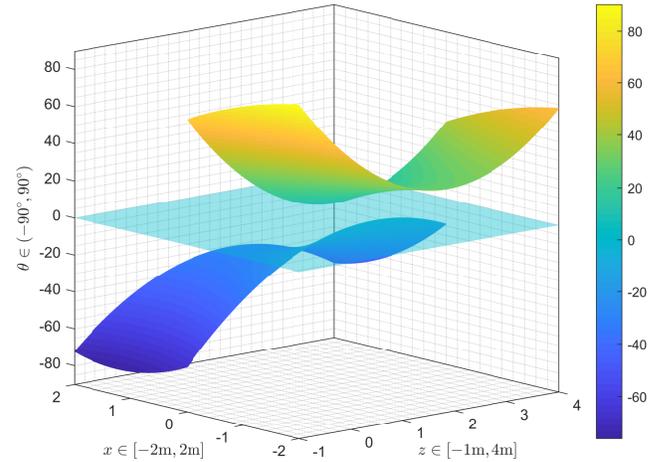


Figure 4: Regressed functions $\Theta^L(x, z)$ and $\Theta^R(x, z)$ that predict virtual agent orientation for fixed user positions.

$X^L(d)$, $Z^L(d)$ or $X^R(d)$, $Z^R(d)$ can predict the optimal positions \hat{p}_t^L and \hat{p}_t^R at each locomotion step t as an intermediate result for PAVAL, while the orientation prediction functions $\Theta^L(x, z)$ or $\Theta^R(x, z)$ determine the final virtual agent's optimal orientation $\hat{\theta}_t$ for PAVAL. The additionally collected 18 virtual agent position and orientation data invariant to the statue (fixed relative to the user, noted as FRU), and 18 virtual agent position and orientation data invariant to the user (fixed relative to the target object, noted as FRT) were used to set up baseline methods. For FRU and FRT, the optimal position and orientation (relative to the user or the target) are determined separately for virtual agents to the left and right of the user, by taking the statistical mean.

4.4 Discussion

Optimal Virtual Agent Position (RQ1) Intuitively, we would expect the optimal position of the virtual agent to lie on two curves which have reflective symmetry with respect to the z -axis. If the user is close to the target, the user would likely prefer the virtual agent to stand close to the target as well, which gives us a small x and z value for the optimal virtual agent position; if the user is far from the target, the optimal position for the virtual agent would be far from the target as well, giving us a larger z value and a slightly larger x value for the optimal virtual agent position.

By referring to Figure 3, we find that the actual experiment results are consistent with our intuitive understanding: as the distance from the user to the target d increases, the absolute value of x slightly increases, but stays around 1m to 1.5m. As for z , it is interesting to

see that z increases at almost the same rate with d for small d , but does not increase as rapidly for larger values of d . This is because as the user moves away from the target, he/she would like the virtual agent to keep close to him/her, while not expecting the virtual agent to stand too far away from the target. We also notice that when d approaches zero, z can take negative values. This is because when the user is close to the target, some of them would prefer the virtual agent to be in sight by standing behind the target.

Optimal Virtual Agent Orientation (RQ2) When the user is close to the target, we expect the virtual agent to form an L-shaped dyadic F-formation with the user; when the user is far from the target, it is more natural for the virtual guide to form a vis-à-vis F-formation with the user. Figure 4 shows the predicted θ values according to virtual agent orientation data obtained from our preliminary user study, which agrees with our intuitive understanding: when the user is close to the target, θ has a larger magnitude and is close to 90° , which corresponds to the L-shaped formation; When the user is far from the target, θ has a smaller magnitude, and the virtual agent is faced more towards the user.

Left-hand Side / Right-hand Side Symmetry (RQ3) Because the experiment conditions for the user’s left-hand side and right-hand side are the same, we would expect the position and orientation of the virtual agent at both sides of the user to be symmetric. We do acknowledge, however, that there could be more data points to the right of the user. The final results showed 185 data points to the right of the user and 175 to the left of the user. As is shown in Figure 3, $X^L(d)$ and $X^R(d)$ exhibit symmetry with respect to $x = 0$, and $Z^L(d)$ is almost identical to $Z^R(d)$, indicating that the virtual agent positions are symmetric for the left and right cases. In Figure 4, the absolute values of θ are also quite symmetric for the left and right cases, indicating that there is symmetry in the virtual agent orientation as well.

5 POSITION-AWARE VIRTUAL AGENT LOCOMOTION

In Section 4, we derived the model for optimal virtual agent position and orientation prediction under the condition that the user stands at a fixed position near an object of interest. We now introduce the PAVAL algorithm that optimizes and updates the virtual agent position and orientation in real time during user navigation.

Formally, we describe the problem as follows: in a virtual environment with N objects $O = o_1, o_2, \dots, o_N$, given the user’s positioning $u_t = (p_t^u, \theta_t^u)$ at time t , where $p_t^u = (x_t^u, z_t^u)$ is the position and θ_t^u is the orientation (under polar coordinates), our goal is to optimize the virtual agent’s positioning $a_t = (p_t^a, \theta_t^a)$ in real time, where $p_t^a = (x_t^a, z_t^a)$ is its position and θ_t^a is its orientation. Our method follows the design guidelines and decomposes the problem into three tasks. At each time t , the first task is to determine the object of interest, which is vital for the virtual agent to act correctly based on the positions of the user and the object. The second task is to optimize the virtual agent’s position, with the consideration of the naturalness of the virtual agent’s locomotion. The last task is to optimize the virtual agent’s orientation based on its optimal position. During the user’s navigation, the virtual agent updates its positioning in real time and verbally introduces the object’s information to the user if available.

5.1 Object of Interest Determination

In our problem, the user is allowed to freely navigate in a virtual scene. Only when the user stops to observe the objects of interest (target object) that can be an exhibition, a machine in a factory, or anything that requires an introduction from the virtual agent, the virtual agent will find a proper position for locomotion and then make an introduction to the user. To this end, it is essential to determine the object of interest at first. We only consider a virtual object as a candidate object of interest if the object is within a 3-meter range

from the user. This is inspired from two aspects: on one hand, the range should not be too small, as the virtual agent needs time for locomotion; on the other hand, the range should not be too large, otherwise distant objects can lead to ambiguity when we determine the user’s object of interest.

By giving the user’s current positioning $u_t = (p_t^u, \theta_t^u)$, the object of interest is determined as the one close to and oriented towards the user as much as possible, from all candidates. We compute the feasibility score to evaluate the selection of object o using the following formula:

$$F(u_t, o) = \alpha_p \cdot F_p(p_t^u, p^o) + \alpha_\theta \cdot F_\theta(\theta_t^u, \theta^o), \quad (4)$$

where $F_p(p_t^u, p^o) = \exp(-0.5 \cdot \|p_t^u - p^o\|^2)$ represents the spatial distance score between the user’s position p_t^u and the object’s position p^o , $F_\theta(\theta_t^u, \theta^o) = (\cos(\theta_t^u - (\theta^o - \pi)) + 1)/2$ represents the orientation score encouraging face-to-face orientation between the user and the object, $\alpha_p = 0.25$, $\alpha_\theta = 0.75$ are constant parameters.

Among all the k objects $O = o_1, o_2, \dots, o_k$ within the 3-meter range of the user, we choose the target with the largest score $F(u_t, o_i)$ to be the object of interest via:

$$o^* = \arg \min_{o_i \in O} F(u_t, o_i). \quad (5)$$

Note that the object of interest determination only depends on the virtual spatial context and the user’s location and orientation. If there is no object within the 3-meter range, the object of interest will be invalid and the virtual agent will simply follow the user’s movement by default until a new object is determined. In addition, it is important to prevent the object of interest from switching frequently due to the fast orientation change of the user when looking around. Thus we constrain the selected object of interest to remain at least one second for a valid selection.

5.2 Optimization of Virtual Agent’s Position

After determination of the object of interest, we optimize the virtual agent’s position p_t^a . Recall that the optimal positions \hat{p}_t^L on the left-hand side and \hat{p}_t^R on the right-hand side for fixed user position can be predicted using our data-driven model (see Section 4). We define an objective that penalizes any deviation from the optimal positions as:

$$E_p^{L,R}(p_t^a) = \exp\left(-\frac{\|p_t^a - \hat{p}_t^{L,R}\|^2}{2\sigma^2}\right), \quad (6)$$

where $\sigma = 1$ is a constant parameter. Intuitively, $E_p^L(p_t^a)$ is maximized when the virtual agent’s position p_t^a is identical to \hat{p}_t^L , and $E_p^R(p_t^a)$ is maximized when the virtual agent’s position p_t^a is identical to \hat{p}_t^R .

Since the virtual agent needs to perform natural locomotion based on varied user positionings (see **DG3**), we add a (soft) constraint to avoid the virtual agent from moving at will and distracting the user’s attention via:

$$E_l(p_t^a) = \exp\left(-\frac{\|p_t^a - \hat{p}_{t-1}^a\|^2}{2\sigma^2}\right), \quad (7)$$

where \hat{p}_{t-1}^a is the position of the virtual agent at time $t - 1$. Intuitively, this term encourages the virtual agent to stay at the previous position.

Combining the two objective terms, we compute the optimal locomotion position \hat{p}_t^a for virtual agent locomotion by maximizing:

$$E(p_t^a) = \max \left\{ E_p^L(p_t^a) + E_l(p_t^a), E_p^R(p_t^a) + E_l(p_t^a) \right\}. \quad (8)$$

To optimize Equation 8 in real time, we apply the simulated annealing technique [19] with a Metropolis Hasting [23] state-searching step to explore the complex optimization position. We constrain the solution space within a circle whose radius is twice the distance between p^o and p_t^a and the candidate position is initialized by setting its value to \hat{p}_t^L and \hat{p}_t^R to speed up the optimization. In each iteration, the current position p_t^a of virtual agent is altered by a proposed move to the next position p , the acceptance probability of next position as virtual agents position is calculated by the following Metropolis criterion:

$$P_r(p, p_t^a) = \min(1, \exp\left(\frac{E(p) - E(p_t^a)}{200 - 0.5n}\right)) \quad (9)$$

where n is the iteration number. In each iteration, p is a point $10 - 0.5n$ centimeters away from p with 50% chance to randomly adjust the searching direction. We finish searching after 200 iterations.

5.3 Optimization of Virtual Agent’s Orientation

According to **DG2**, for each time t , the virtual agent determines its position at first, then set up its orientation with the fixed position. Since the optimal position \hat{p}_t^a is determined via Equation 8, we again use the data-driven model to predict the virtual agent’s optimal orientation $\hat{\theta}_t^a$ given the \hat{p}_t^a .

5.4 From Positioning to Locomotion

Collision avoidance of the virtual agent should also be considered when moving to a target position or following the user. We need a path planning technique to move the virtual agent from the previous positioning to a new positioning. There are many off-the-shelf locomotion path planning techniques, such as traditional heuristic algorithms [15, 32], probabilistic roadmap [28], neural network-based technique [25], etc. We perform virtual agent’s locomotion by connecting the optimized positions and orientations using the A* algorithm integrated in the Unity Navigation System.

6 EXPERIMENT

We conducted a user study with 27 participants to evaluate the performance of methods for virtual agent locomotion.

6.1 Experimental Setup

We equipped a $9m \times 6m$ room with a workstation, and an *HTC Vive Cosmos* headset with hand-held controllers. Our approach was implemented using C# and Unity v2018.3.5f1 and run on a PC equipped with 32GB of RAM, a GTX 2080Ti Graphics card, and an i7 9700K processor. The headset was connected to the PC via a cable, which transmitted the position and orientation of the user as well as the optimization results to the virtual agent.

6.2 Participants

27 participants were recruited to complete the experiment (15 males and 12 females, mean age=22.8, SD=1.45). All the subjects were researchers or students from a local university. They did not report any visual impairment and claimed to be in good physical condition to take part in the experiment. 20 subjects reported that they lacked previous experience in VR and were unfamiliar with the devices.

6.3 Virtual Environments and Avatars

In our experiment, we compared different techniques in three virtual environments listed as follows:

Museum: This virtual scene contains an exhibition hall for ancient India, which contained 16 ancient art models. We prepared four commentaries for 12 models. This virtual scene can be found on Unity Asset Store - *Museum VR Complete Edition Vol. 4*.

Factory: This virtual scene of a factory contains 6 models of industrial equipment, and several complementary parts (beams, pipes,



Figure 5: Virtual scenes and corresponding virtual agents used in our experiments. Candidate objects of interest in each scene are marked in yellow.

etc.). These models were made from photos of real industrial equipment, 5 of which have commentaries prepared. This virtual scene can be found on Unity Asset Store - *Factory Hardware Collection*.

School Gym: This virtual scene of the interior of a school gym in the daytime contains parallel bars, a pommel horse, as well as facilities for various sports, such as basketball and table tennis. Similarly, we prepared commentaries for 7 of these sports facilities. This virtual scene can be found on Unity Asset Store - *School Gym*.

We selected three high-quality digital avatars from Microsoft Rocketbox Avatar Library [12] that best fit the corresponding VE as the human-like virtual agents. Figure 5 shows the VEs and virtual agent avatars. The virtual agent’s gesture animations were designed by an artist, and the emotion of each avatar was neutral. The verbal commentaries were produced using Google text-to-speech services in local language. The user’s self-avatar was transparent (unseen).

6.4 Methods

To evaluate the performance of PAVAL, we compared our method against two baselines:

Fixed Relative to Target (FRT): When the user approaches a target (e.g., an exhibit in a museum), the virtual agent’s position and orientation are fixed relative to the target object (object of interest).

Fixed Relative to User (FRU): The position of the virtual agent with respect to the user is fixed. The virtual agent kept the same position and relative to the user as he/she moves around in the VE. The optimal positions and orientations for FRT and FRU are determined in the preliminary study, as is explained in Section 4.

Position-Aware Virtual Agent Locomotion (PAVAL): The position and orientation of the virtual agent are computed and updated in real time as described in Section 5. Figure 6 shows visual comparisons of the three methods. For more visual comparison results, please refer to the supplementary video.

Here, FRT and FRU are chosen as the two baseline methods because they are extreme cases: one being static with respect to the virtual environment, and the other being dynamic which constantly follows the user, which helps us to interpret users’ ratings based on this scale provided by the two extremes.

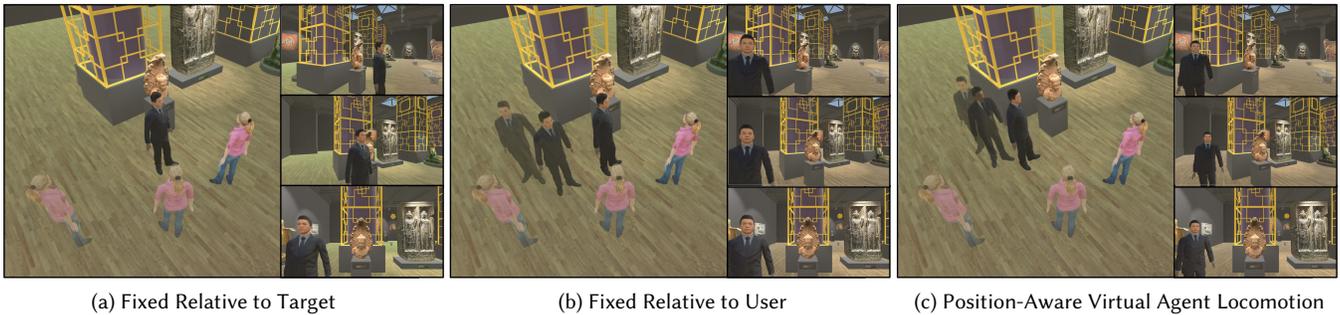


Figure 6: Visual comparisons of virtual agent locomotion methods in the museum scene. For each method, three first-person views and a third-person view are shown at three representative timings during the navigation, where the user and corresponding virtual agent’s avatar transparency are 40%, 20% and 0% in the third-person view. (a) Fixed Relative to Target (FRT): the virtual agent positions and orientations are fixed relative to the target. (b) Fixed Relative to User (FRU): the virtual agent positions and orientations are fixed relative to the user. (c) Position-Aware Virtual Agent Locomotion (PAVAL): the virtual agent positions and orientations are optimized and updated in real time.

6.5 Measures

During the user study, we used 11-point Likert scales to measure subjective perception in terms of the position and orientation of the virtual agent according to different user navigation states:

Static Position: The participant was asked to evaluate the virtual agent’s position while the participant was standing at a position and observing the object of interest (0-worst, 10-best).

Static Orientation: The participant was asked to evaluate the virtual agent’s orientation while the participant was standing at a position and observing the object of interest (0-worst, 10-best).

Dynamic Position: The participant was asked to evaluate the virtual agent’s position while the user was moving, i.e., the user was observing the object from a few different angles or distances by walking around the target (0-worst, 10-best).

We clarify that we do not evaluate the virtual agent’s orientation while the user is *walking*, because the virtual agent’s orientation during locomotion is consistent to the locomotion direction, which is determined by the path planing tool (see Section 5.4).

6.6 Procedure

Upon the participants’ arrival at the lab, they read and signed an informed consent form regarding instructions of the experiment, which was followed by an extra explanation in case the participants had any questions. Then they completed a questionnaire for demographic information, including name, gender, age, as well as VR familiarity. Before the experiment, each participant had around 5 minutes to try the VR headset. The experiment began only if the participant was already familiar with the equipment and how to navigate in a simple training VE. In each testing VE, the participant compared three different virtual agent locomotion techniques by exploring the VE with a free walk, until they were able to evaluate each method. Forced reset operations were used to keep the participant away from physical obstacles. Each trial typically lasted from 5 to 10 minutes. To avoid the potential influence of the test order on the experiment results, we counterbalanced the trials using Latin square. After each trial, the user rated the current virtual agent locomotion technique in the current VE using the 11-point Likert scales. After finishing all trials, we invited participants to do a short interview with suggestions for improvements. In the end, each participant was thanked and paid for their participation.

6.7 Hypothesis

We make the following hypotheses:

- **H1:** PAVAL outperforms baseline methods in terms of the Static Position measure.
- **H2:** PAVAL outperforms baseline methods in terms of the Dynamic Position measure.

Table 1: χ^2 and p -values for the Kruskal-Wallis test performed for FRT, FRU and PAVAL methods under SP, DP and SO measures.

| Measure | Museum | | Factory | | School Gym | |
|---------|----------|-------|----------|-------|------------|-------|
| | χ^2 | p | χ^2 | p | χ^2 | p |
| SP | 32.38 | <.001 | 27.81 | <.001 | 28.29 | <.001 |
| DP | 39.67 | <.001 | 30.31 | <.001 | 27.40 | <.001 |
| SO | 32.20 | <.001 | 29.17 | <.001 | 25.70 | <.001 |

- **H3:** PAVAL outperforms baseline methods in terms of the Static Orientation measure.

7 RESULTS

We collected user ratings on three virtual agent locomotion techniques FRT, FRU and PAVAL (ours) in terms of Static Position (SP), Dynamic Position (DP), and Static Orientation (SO) under three virtual scenes, including Museum, Factory, and School Gym. We group the subjective data under the same measure and the same virtual scene ($3 \times 3 = 9$ groups), each containing the corresponding ratings from 27 participants on three methods under a VE. Figure 7 shows the box plot of the subjective ratings.

With a Shapiro-Wilk normality test, we found that the data was not normally distributed. We then conducted the non-parametric Kruskal-Wallis test for all groups of data. Dunn’s pairwise comparison tests with the Bonferroni adjustment were carried out for the three methods under the same virtual scene and measure.

The Kruskal-Wallis test provided strong evidence that the participants found a significant difference in SP, DP, and SO among the three methods ($p < 0.001$). This result applies to all three virtual environments, details of which are shown in Table 1. We further carried out Dunn’s pairwise tests for the three pairs of methods, and found strong evidence of a statistical difference between the ratings of PAVAL and FRT, as well as PAVAL and FRU. We observed that the adjusted significance values (using the Bonferroni correction) for SP, DP, and SO ratings between PAVAL and FRT, as well as PAVAL and FRU all satisfied $p \leq 0.001$.

For the measure of SP where the user stands at a fixed position and observes a target object, we found significant differences between PAVAL and FRT for all VEs (Museum: $p < .001$, Effect-size $r = 0.601$; Factory: $p < .001$, $r = 0.563$; School Gym: $p = .001$, $r = 0.523$), with PAVAL reporting higher means (Museum: 35.4%, Factory: 34.9%, School Gym: 30.4%) compared to FRT. Results also revealed significant differences between PAVAL and FRU for all VEs (Museum: $p < .001$, $r = 0.682$; Factory: $p < .001$, $r = 0.623$; School Gym: $p < .001$, $r = 0.650$), with PAVAL reporting higher means (Museum: 53.1%, Factory: 49.5%, School Gym: 54.6%) than FRU. These results support the hypothesis **H1**.

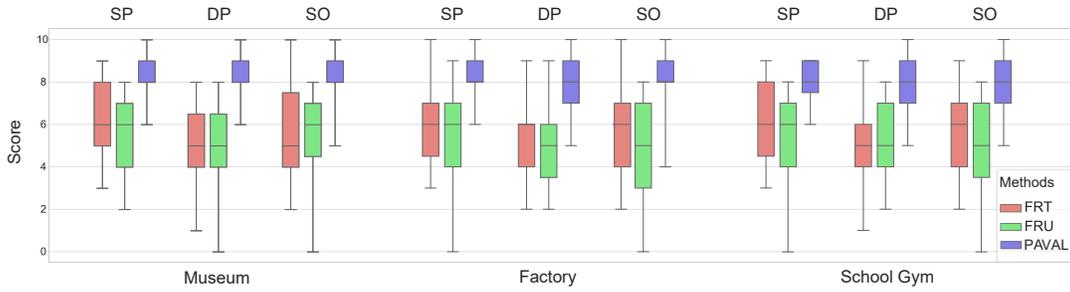


Figure 7: Subjective data of static position, dynamic position, and static orientation for methods under different VE conditions.

The results of DP revealed significant differences between PAVAL and FRT for all VEs (Museum: $p < .001$, $r = 0.724$; Factory: $p < .001$, $r = 0.601$; School Gym: $p < .001$, $r = 0.625$), with PAVAL reporting higher means (Museum: 67.5%, Factory: 45.8%, School Gym: 54.6%) than FRT. PAVAL is also significantly different from FRU in terms of DP (Museum: $p < .001$, $r = 0.715$; Factory: $p < .001$, $r = 0.655$; School Gym: $p < .001$, $r = 0.610$). The means of DP for PAVAL are higher (Museum: 66.1%, Factory: 57.8% and School Gym: 48.9%) than those for FRU. Thus, **H2** is supported.

For SO, there also exist significant differences between PAVAL and FRT for all VEs (Museum: $p < .001$, $r = 0.613$; Factory: $p < .001$, $r = 0.558$; School Gym: $p < .001$, $r = 0.601$), where PAVAL had higher mean ratings (Museum: 46.4%, Factory: 43.0%, School Gym: 44.4%) compared with FRT. PAVAL is also significantly different from FRU in terms of SO in all VEs (Museum: $p < .001$, $r = 0.636$; Factory: $p < .001$, $r = 0.634$; School Gym: $p < .001$, $r = 0.568$). The mean ratings of PAVAL in terms of SO are higher than those of FRU (Museum: 51.5%, Factory: 60.7% and School Gym: 55.0%). Hence, **H3** is supported.

8 DISCUSSION

If a user observes the target from multiple angles, we often face a trade-off between occlusion and distraction: if the virtual agent stays still when the user moves to a new location, the virtual agent may occlude key objects; if it moves too frequently even when the user only moves within a small range, the virtual agent may become a source of disturbance to the user. From the ratings, PAVAL clearly outperformed the two baseline methods, while there was not a significant difference between the scores of FRT and FRU. This is because FRT could cause the virtual agent to occlude key objects and its fixed position could be out of the user’s sight, causing it to lack practicality in real-world applications (see Figure 6(a)). For FRU, the constant presence of the virtual agent in the user’s FoV can distract the user [35]. As for our method, PAVAL computes the optimal position for the virtual agent in real time based on the sampled user location and target location, while considering the trade-off between occlusion and distraction. Between discrete optimal positions for the virtual agent, the A* search algorithm is used to generate a continuous movement path for the virtual agent (see Figure 6(c)).

Recall that orientation also plays a key role in the different formations summarized by Kendon et al. [18]. The two baseline methods have fixed orientations relative to the object or to the user, which fail to take the real-time user and target positions into account. Both methods do not adapt their orientation according to the user’s position and orientation, which probably resulted in an equally unnatural experience for the user. PAVAL, on the other hand, optimizes the virtual agent’s orientation based on the current position and orientation of the user as well as information from the virtual scene, which promotes the interaction between the user and virtual agent, making it more natural and realistic.

Open Comments: From the collected comments, five participants expressed that it would be better to enrich the virtual agent’s animations such as gesture types, styles, etc. Two participants felt

restless for being watched closely by the virtual agent when experiencing the FRU method. There was also a complaint to the FRT method that it made him felt less interactive in VR. Four participants would like to experience the PAVAL method for more scenes. One participant suggested that it would be good to allow the user to switch between different testing methods.

9 CONCLUSIONS, LIMITATIONS AND FUTURE WORK

In this paper, we demonstrated the effectiveness and potential of modeling virtual agent locomotion for assisted VR navigation. The proposed position-aware virtual agent locomotion algorithm considers the spatial relationship between the user and the target object, and optimizes the position and orientation for the virtual agent in real time. As shown in our experiments, we applied our approach in virtual tour (Museum), virtual training (Factory), and virtual education (School Gym) scenarios. User study results indicate significant improvements over baseline methods in terms of virtual agent position and orientation quality.

Our work has limitations. First, in our experiments, the behavior of virtual agents is simplified and not fully interactive. During the navigation assistance, the virtual agent’s gesture animations are repeated. The virtual agent is not as intelligent as expected, for instance, to answer questions raised by the user. As these features are not the focus of this paper, we regard the development of comprehensive virtual agent locomotion with gazes, gestures, gait [4], emotion, and clothing [16] as future work.

Second, we acknowledge that in our approach, virtual agent locomotion for single user navigation is considered. If more than one users experience the scene at the same time in a group, the locomotion should be different, considering the spatial distribution of the users [39, 40]. Moreover, in large-scale virtual scenes, multiple virtual agents with natural actions and behaviors are to be expected. Multi-agent systems would be used to solve this problem.

Third, in our data-driven model for position and orientation prediction, we fitted the subjective data using polynomials and achieved plausible results. Neural networks can be alternative solutions to solve the regression problem. In addition, the two baseline methods also have great potential for improvement with heuristic optimization. We believe that it is worthy of further research in the future.

Fourth, in our preliminary subjective user study, we simplified the scene, and did not consider the influence of different object sizes and different observing directions. Developing complex data-driven models that consider more aspects in the VE is expected. The use of global and local scene contexts for virtual agent locomotion can be a promising research direction.

ACKNOWLEDGMENTS

The authors wish to thank the anonymous reviewers for their helpful advices. This work was supported by the National Natural Science Foundation of China (Project Number: 61902012 and 61932003) and Baidu academic collaboration program. Yong-Liang Yang was supported by RCUK grant CAMERA (EP/M023281/1, EP/T022523/1), and a gift from Adobe.

REFERENCES

- [1] H. Akiyama and I. Noda. *Multi-Agent Positioning Mechanism in the Dynamic Environment*, p. 377–384. Springer-Verlag, Berlin, Heidelberg, 2008.
- [2] M. Alibali. Gesture in spatial cognition: Expressing, communicating, and thinking about spatial information. *Spatial Cognition and Computation - SPAT COGN COMPUT*, 5:307–331, 12 2005.
- [3] K. Best. Making museum tours better: understanding what a guided tour really is and what a tour guide really does. *Museum Management and Curatorship*, 27(1):35–52, 2012.
- [4] U. Bhattacharya, N. Rewkowski, P. Guhan, N. L. Williams, T. Mittal, A. Bera, and D. Manocha. Generating emotive gaits for virtual agents using affect-based autoregression. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 24–35. IEEE, 2020.
- [5] M. Carrozzino, M. Colombo, F. Tecchia, C. Evangelista, and M. Bergamasco. Comparing different storytelling approaches for virtual guides in digital immersive museums. In *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*, pp. 292–302. Springer, 2018.
- [6] K. Cheverst, N. Davies, K. Mitchell, A. Friday, and C. Efstathiou. Developing a context-aware electronic tourist guide: Some issues and experiences. p. 17–24. Association for Computing Machinery, 2000.
- [7] M. Christie, P. Olivier, and J.-M. Normand. Camera control in computer graphics. In *Computer Graphics Forum*, vol. 27, pp. 2197–2218. Wiley Online Library, 2008.
- [8] E. Cohen. The tourist guide: The origins, structure and dynamics of a role. *Annals of Tourism Research*, 12(1):5–29, 1985.
- [9] G. Elber, M. Gertelman, O. Shaked, and O. Shmueli. Object positioning and display in virtual environments, May 9 2006. US Patent 7,043,695.
- [10] G. Ghiani, F. Paternò, C. Santoro, and L. D. Spano. Ubicicero: A location-aware, multi-device museum guide. *Interacting with Computers*, 21(4):288–303, 2009.
- [11] J. J. Gibson. The concept of affordances. *Perceiving, acting, and knowing*, 1, 1977.
- [12] M. Gonzalez-Franco, E. Ofek, Y. Pan, A. Antley, A. Steed, B. Spanlang, A. Maselli, D. Banakou, N. Pelechano Gómez, S. Orts-Escolano, et al. The rocketbox library and the utility of freely available rigged avatars. *Frontiers in virtual reality*, 1(article 561558):1–23, 2020.
- [13] H. Grabner, J. Gall, and L. Van Gool. What makes a chair a chair? In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '11*, p. 1529–1536. IEEE Computer Society, USA, 2011.
- [14] E. T. Hall, R. L. Birdwhistell, B. Bock, P. Bohannon, A. R. Diebold Jr, M. Durbin, M. S. Edmonson, J. Fischer, D. Hymes, S. T. Kimball, et al. Proxemics [and comments and replies]. *Current anthropology*, 9(2/3):83–108, 1968.
- [15] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.
- [16] S. Hou, Y. Wang, B. Ning, and W. Liang. Climaxing vr character with scene-aware aesthetic dress synthesis. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 57–64. IEEE, 2021.
- [17] J. Ibanez, R. Aylett, and R. Ruiz-Rodarte. Storytelling in virtual environments from a virtual guide perspective. *Virtual Real.*, 7(1):30–42, Dec. 2003.
- [18] A. Kendon. Spacing and orientation in co-present interaction. In *Proceedings of the Second International Conference on Development of Multimodal Interfaces: Active Listening and Synchrony, COST'09*, p. 1–15. Springer-Verlag, Berlin, Heidelberg, 2009.
- [19] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- [20] Y. Lang, W. Liang, and L.-F. Yu. Virtual agent positioning driven by scene semantics in mixed reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 767–775, 2019.
- [21] W. Liang, X. Yu, R. Alghofaili, Y. Lang, and L.-F. Yu. Scene-aware behavior synthesis for virtual pets in mixed reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, CHI '21*. Association for Computing Machinery, New York, NY, USA, 2021.
- [22] C. Lino and M. Christie. Intuitive and efficient camera control with the toric space. *Acm Transactions on Graphics*, 34(4CD):82.1–82.12, 2015.
- [23] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.
- [24] N. Norouzi, K. Kim, J. Hochreiter, M. Lee, S. Daher, G. Bruder, and G. Welch. A systematic survey of 15 years of user studies published in the intelligent virtual agents conference. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents, IVA '18*, p. 17–22. Association for Computing Machinery, New York, NY, USA, 2018.
- [25] Y. Peng, S.-W. Li, and Z.-Z. Hu. A self-learning dynamic path planning method for evacuation in large public buildings based on neural networks. *Neurocomputing*, 365:71–85, 2019.
- [26] T. Randhavane, A. Bera, K. Kapsaskis, K. Gray, and D. Manocha. Fva: Modeling perceived friendliness of virtual agents using movement characteristics. *IEEE Transactions on Visualization and Computer Graphics*, 25(11):3135–3145, 2019.
- [27] T. Randhavane, A. Bera, K. Kapsaskis, R. Sheth, K. Gray, and D. Manocha. Eva: Generating emotional behavior of virtual agents using expressive features of gait and gaze. In *ACM symposium on applied perception 2019*, pp. 1–10, 2019.
- [28] M. T. Rantanen. A connectivity-based method for enhancing sampling in probabilistic roadmap planners. *Journal of Intelligent & Robotic Systems*, 64(2):161–178, 2011.
- [29] R. Rzaev, G. Karaman, N. Henze, and V. Schwind. Fostering virtual guide in exhibitions. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '19*. Association for Computing Machinery, New York, NY, USA, 2019.
- [30] M. Savva, A. X. Chang, P. Hanrahan, M. Fisher, and M. Nießner. Scene-grok: Inferring action maps in 3d environments. *ACM Transactions on Graphics (TOG)*, 33(6), 2014.
- [31] S. Schmidt, G. Bruder, and F. Steinicke. Effects of embodiment on generic and content-specific intelligent virtual agents as exhibition guides. In *Proceedings of the International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments (ICAT-EGVE)*, pp. 13–20, 2018.
- [32] A. Stentz. Optimal and efficient path planning for partially known environments. In *Intelligent unmanned ground vehicles*, pp. 203–220. Springer, 1997.
- [33] N. Techasartikul, P. Ratsamee, J. Orlosky, T. Mashita, Y. Uranishi, K. Kiyokawa, and H. Takemura. Evaluation of embodied agent positioning and moving interfaces for an ar virtual guide. In *ICAT-EGVE*, pp. 25–32, 2019.
- [34] W. R. van Hage, N. Stash, Y. Wang, and L. Aroyo. Finding your way through the rijksmuseum with an adaptive mobile museum guide. pp. 46–59. Springer Berlin Heidelberg, 2010.
- [35] M. Volonte, A. Robb, A. T. Duchowski, and S. V. Babu. Empirical evaluation of virtual human conversational and affective animations on visual attention in inter-personal simulations. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 25–32. IEEE, 2018.
- [36] A. M. Von Der Pütten, N. C. Krämer, and J. Gratch. 1 who´s there? can a virtual agent really elicit social presence? 2009.
- [37] R. Wakkary, K. Muise, K. Tanenbaum, M. Hatala, and L. Kornfeld. Situating approaches to interactive museum guides. *Museum Management and Curatorship*, 23(4):367–383, 2008.
- [38] M. Wang, Z.-M. Ye, J.-C. Shi, and Y.-L. Yang. Scene-context-aware indoor object selection and movement in vr. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 235–244. IEEE, 2021.
- [39] T. Weissker, P. Bimberg, and B. Froehlich. Getting there together: Group navigation in distributed virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1860–1870, 2020.
- [40] T. Weissker and B. Froehlich. Group navigation for guided tours in distributed virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2524–2534, 2021.