Learning Human Perception Dynamics for Informative Robot Communication

Shenghui Chen[1], Ruihan Zhao[1], Sandeep Chinchali[1] and Ufuk Topcu[1]

***Abstract*— Human-robot cooperative navigation is challenging in environments with incomplete information. We introduce *CoNav-Maze*, a simulated environment where a robot navigates using local perception while a human operator provides guidance based on an inaccurate map. The robot can share its camera views to improve the operator's understanding of the environment. To enable efficient human-robot cooperation, we propose *Information Gain Monte Carlo Tree Search* (IG-MCTS), an online planning algorithm that balances autonomous movements and informative communication. Central to IG-MCTS is a fully convolutional neural human perception dynamics model that estimates how humans distill information from robot communications. We collect a dataset through a crowdsourced mapping task in CoNav-Maze to train this model before the cooperative navigation experiments. User studies show that IG-MCTS outperforms teleoperation and instruction-following baselines, achieving comparable task performance with significantly less communication and lower human cognitive load, as evidenced by eye-tracking metrics.**

I. Introduction

In novel environments, autonomous robots often face a "cold-start" problem, where they operate without sufficient prior environment knowledge. Without human guidance, the robot may make inefficient decisions, leading to wasted time and resources during exploration. Beyond improving efficiency, human involvement may also play a valuable role for safety, ethical, and moral considerations [1], [2], [3].

Collaboration between humans and robots under incomplete information presents a realistic and challenging problem. Since they often receive input from different sources, maintaining a shared understanding becomes especially difficult in dynamic or unfamiliar environments, where sudden changes can quickly lead to misalignment. The key challenge lies in enabling the robot to effectively leverage human knowledge to complement its local observations and support robust human-robot synergy. Such robots hold the potential to assist human operators in search-and-rescue missions [4], [5] and support individuals with disabilities in daily tasks [6], [7].

This paper investigates the problem of human-robot cooperative navigation in a simulated environment, *CoNav-Maze*, where the robot receives local SLAM observations while the human provides guidance based on an initially inaccurate global map. The objective is to develop a control algorithm that goes beyond passively following instructions—one that actively collaborates by transmitting visual observations to improve the human's situational awareness, integrating human trajectory suggestions, and maintaining sufficient autonomy to reach target locations efficiently.

[1]The University of Texas at Austin, USA

We introduce *Information Gain Monte Carlo Tree Search* (IG-MCTS), an online planning algorithm that embodies the idea that *communication is action*. Aside from task-centric objectives, IG-MCTS strategically decides between movement and communication actions based on their potential to enhance the human's understanding of the environment. Inspired by evidence that when reading, humans minimize perceptual errors and extract relevant features under limited processing capacity [8], we hypothesize that a similar cognitive strategy applies to visual tasks. To align with this cognitive pattern, IG-MCTS chooses camera angles that maximize an information reward that measures the change in the human's perception of the environment. IG-MCTS also incorporates human-guided trajectories as reward augmentations [9].

At the core of IG-MCTS is a data-driven human perception dynamics model that predicts how humans update their understanding of the environment in response to the robot's actions. We introduce a fully convolutional neural network (FCNN) that unifies the effects of both robot movements and communication while incorporating spatial structure and contextual awareness. To train this model, we crowdsourced a dataset of human-robot interactions in CoNav-Maze, capturing human information-processing patterns in navigation tasks. The model learns to estimate the human operator's evolving perception by fitting to human annotations. Evaluation results show that the FCNN-based approach achieves higher prediction accuracy than a psychometric function-based model [10].

We evaluate the performance of IG-MCTS in CoNav-Maze against two baselines: teleoperation and instruction-following. A user study with 10 eye-tracked participants shows that interacting with the IG-MCTS robot significantly reduces communication demands while yielding eye-tracking metrics indicative of lower cognitive load, all while maintaining task performance on par with the baselines.

Related Work. Prior work on human perception has modeled how physical stimuli influence sensory responses using psychometric functions, particularly logistic variants that capture detection probabilities as stimulus intensity varies [11], [12]. However, such models may oversimplify nuanced perceptual dynamics or cannot capture diverse human suboptimality [13]. In teleoperation, immersive interfaces have enhanced remote robot control [14], but cognitive load and communication constraints still limit human-to-robot ratio [15]. Meanwhile, vision-and-language navigation (VLN) tasks require agents to interpret human instructions and navigate through 3D environments by executing action sequences [16], but often assume complete information, static environments [17], [18], [19], or panoramic action spaces [20]. In contrast, our

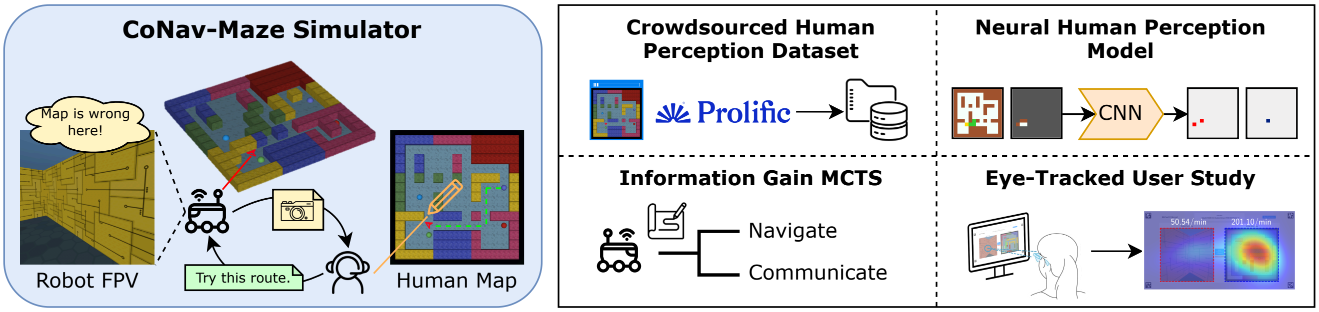


Fig. 1: We study efficient human-robot collaboration in maze navigation under incomplete information. **Left:** The robot gathers local observations; the human relies on an imprecise global map. They collaborate by exchanging images and path suggestions. **Right:** Our study involves: (1) Crowdsourcing a dataset of human perceptual updates; (2) Training a perception dynamics model to estimate human understanding; (3) Developing IG-MCTS to balance navigation and communication; (4) Validating our approach via eye-tracking and task performance in a user study.

approach focuses on human-robot coordination in incomplete and dynamic environments, allowing users to specify flexible routes that evolve as new information is gathered.

II. Problem Setting

This paper addresses a human-robot cooperative navigation task under incomplete information. The remote human operator possesses an outdated map of the environment, while the robot can acquire accurate local observations. The human provides navigation guidance, and the robot communicates environmental updates. Together, they aim to reach a set of goal locations as efficiently as possible.

We design *CoNav-Maze*, a simulated maze environment adapted from MemoryMaze [21]. In CoNav-Maze, the robot has perfect knowledge of its position and uses motion primitives to navigate between adjacent grid cells. This setup abstracts away low-level control and estimation errors, focusing on high-level human-robot coordination.

The environment is modeled as a Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, T, R_{\text{env}}, \gamma)$. $\mathcal{S}$ is a product space comprising the robot's discrete finite state and the set of remaining goal locations, capturing both its position and task progress. $\mathcal{A}$ is a finite set of actions, including movement to adjacent grids and transmitting a first-person image from one of eight evenly spaced camera angles. $T : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is a deterministic transition function. $R_{\text{env}} : \mathcal{S} \to \mathbb{R}$ is a real-valued reward function. $\gamma \in [0, 1)$ is a discount factor.

At each step t, the robot collects a local observation of nearby traversable and blocked cells within a radius r. It may also receive a human-provided trajectory ζ_t. The robot then selects an action a_t to either move or transmit an image.

The human operator starts with an inaccurate global map $x \in \mathcal{X}$, representing traversable and blocked cells. By analyzing the robot's trajectory and image transmissions, the human refines their map to provide more accurate guidance.

III. Method

Effective human-robot cooperation in CoNav-Maze hinges on efficient communication. Maximizing the human's information gain enables more precise guidance, which in turn accelerates task completion. Yet for the robot, the challenge is not only *what* to communicate but also *when*, as it must balance gathering information for the human with pursuing immediate goals when confident in its navigation.

To achieve this, we introduce *Information Gain Monte Carlo Tree Search* (IG-MCTS), which optimizes both task-relevant objectives and the transmission of the most informative communication. IG-MCTS comprises three key components: **(1)** A data-driven human perception model that tracks how implicit (movement) and explicit (image) information updates the human's understanding of the maze layout. **(2)** Reward augmentation to integrate multiple objectives effectively leveraging on the learned perception model. **(3)** An uncertainty-aware MCTS that accounts for unobserved maze regions and human perception stochasticity.

A. Human Perception Dynamics

As the robot navigates the maze and transmits images, humans update their understanding of the environment. Based on the robot's path, they may infer that previously assumed blocked locations are traversable or detect discrepancies between the transmitted image and their map.

To formally capture this process, we model the evolution of human perception as another Markov Decision Process, referred to as the *Perception MDP*. The state space $\mathcal{X}$ represents all possible maze maps. The action space $\mathcal{S}^+ \times \mathcal{O}$ consists of the robot's trajectory between two image transmissions $\tau \in \mathcal{S}^+$ and an image $o \in \mathcal{O}$. The unknown transition function $F : (x, (\tau, o)) \to x'$ defines the human perception dynamics, which we aim to learn.

1) Crowd-Sourced Transition Dataset: To collect data, we designed a mapping task in the CoNav-Maze environment. Participants were tasked to edit their maps to match the true environment. A button triggers the robot's autonomous movements, after which it captures an image from a random angle. In this mapping task, the robot, aware of both the true environment and the human's map, visits predefined target locations and prioritizes areas with mislabeled grid cells on the human's map.

Subsequently, we recruited over 50 annotators through Prolific [22] for the mapping task. Each annotator labeled three randomly generated mazes. They were allowed to

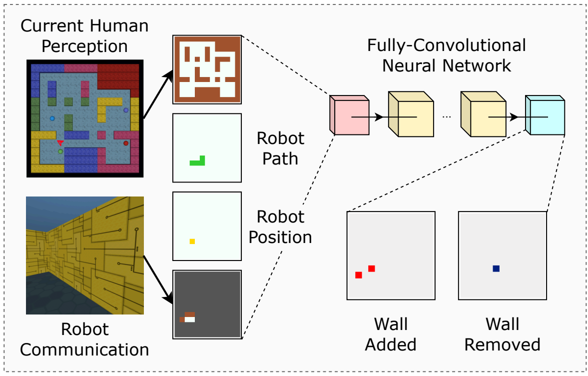


Fig. 2: Neural Human Perception Model (NHPM). **Left:** The human's current perception, the robot's trajectory since the last transmission, and the captured environment grids are individually processed into 2D masks. **Right:** A CNN predicts two masks: one for the probability of the human adding a wall to their map and another for removing a wall.

proceed to the next maze once the robot had reached all four goal locations. However, they could spend additional time refining their map before moving on. To incentivize accuracy, annotators receive a performance-based bonus based on the final accuracy of their annotated map.

2) Fully-Convolutional Dynamics Model: We introduce the Neural Human Perception Model (NHPM), a fully convolutional neural network (FCNN) designed to predict human perception transition probabilities, as formulated in Section III-A. We denote this model as F_θ, where θ represents the trainable parameters. This design is inspired by recent advances in model-based reinforcement learning [23], where agents learn environment dynamics—often from visual observations [24], [25].

As shown in Figure 2, NHPM takes as input the human's current perception, the robot's trajectory, and the image captured by the robot. These inputs are encoded into a unified 2D representation, concatenated along the channel dimension, and processed by the CNN. The model outputs a two-channel image: one channel predicts the probability of the human adding a new wall, and the other predicts the probability of removing an existing wall.

B. Perception-Aware Reward Augmentation

The robot optimizes its actions over a planning horizon H by solving the following optimization problem:

$$\max_{a_{0:H-1}} \mathop{\mathbb{E}}_{T,F} \left[\sum_{t=0}^{H-1} \gamma^t \left(\underbrace{R_{\text{task}}(\tau_{t+1}, \zeta)}_{\text{(1) Task reward}} + \underbrace{\|x_{t+1} - x_t\|_1}_{\text{(2) Info reward}} \right) \right] \tag{1a}$$

$$\text{s.t.} \quad x_{t+1} = F(x_t, (\tau_t, a_t)), \quad a_t \in \mathcal{O} \tag{1b}$$

$$\tau_{t+1} = \tau_t \oplus T(s_t, a_t), \quad a_t \in \mathcal{U} \tag{1c}$$

The objective in (1a) maximizes the expected cumulative reward over T and F, reflecting the uncertainty in both physical transitions and human perception dynamics. The reward function consists of two components: (1) The *task reward* incentivizes efficient navigation. The specific formulation for the task in this work is outlined in Section V-A. (2) The *information reward* quantifies the change in the human's perception due to robot actions, computed as the L_1-norm distance between consecutive perception states.

The constraint in (1c) ensures that for movement actions, the trajectory history τ_t expands with new states based on the robot's chosen actions, where s_t is the most recent state in τ_t, and $\oplus$ represents sequence concatenation. In constraint (1b), the robot leverages the learned human perception dynamics F to estimate the evolution of the human's understanding of the environment from perception state x_t to x_{t+1} based on the observed trajectory τ_t and transmitted image $a_t \in \mathcal{O}$.

C. Information Gain Monte Carlo Tree Search (IG-MCTS)

IG-MCTS follows the four stages of Monte Carlo tree search: *selection*, *expansion*, *rollout*, and *backpropagation*, but extends it by incorporating uncertainty in both environment dynamics and human perception. We introduce uncertainty-aware simulations in the *expansion* and *rollout* phases and adjust *backpropagation* with a value update rule that accounts for transition feasibility.

1) Uncertainty-Aware Simulation: As detailed in Algorithm 1, both the *expansion* and *rollout* phases involve forward simulation of robot actions. Each tree node v contains the state (τ, x), representing the robot's state history and current human perception. We handle the two action types differently as follows:

- A movement action u follows the environment dynamics T as defined in Section II. Notably, the maze layout is observable up to distance r from the robot's visited grids, while unexplored areas assume a 50% chance of walls. In *expansion*, the resulting search node v' of this uncertain transition is assigned a feasibility value $\delta = 0.5$. In *rollout*, the transition could fail and the robot remains in the same grid.
- The state transition for a communication step o is governed by the learned stochastic human perception model F_θ as defined in Section III-A.2. Since transition probabilities are known, we compute the expected information reward $\bar{R_{\text{info}}}$ directly:

$$\begin{aligned} \bar{R_{\text{info}}}(\tau_t, x_t, o_t) &= \mathbb{E}_{x_{t+1}} \|x_{t+1} - x_t\|_1 \\ &= \|p_{\text{add}}\|_1 + \|p_{\text{remove}}\|_1, \end{aligned}$$

where $(p_{\text{add}}, p_{\text{remove}}) \leftarrow F_\theta(\tau_t, x_t, o_t)$ are the estimated probabilities of adding or removing walls from the map. Directly computing the expected return at a node avoids the high number of visitations required to obtain an accurate value estimate.

2) Feasibility-Adjusted Backpropagation: During backpropagation, the rewards obtained from the simulation phase are propagated back through the tree, updating the total value $Q(v)$ and the visitation count $N(v)$ for all nodes along the path to the root. Due to uncertainty in unexplored environment dynamics, the rollout return depends on the feasibility of the

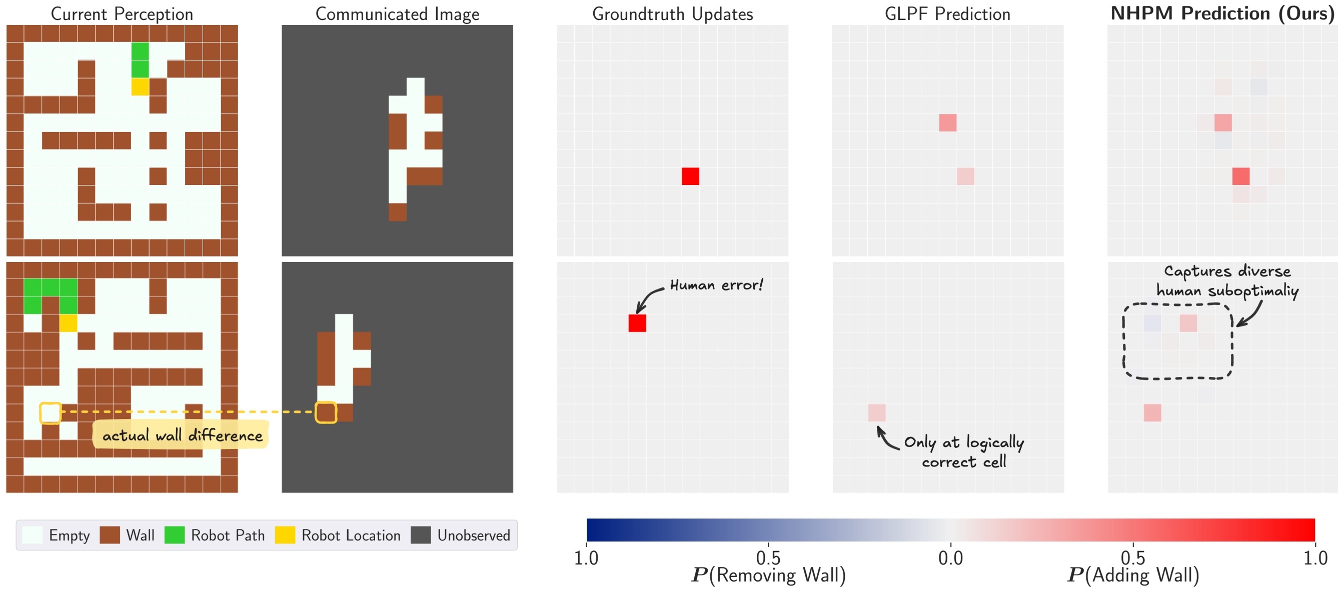


Fig. 3: Visualization of human perception models. The left two columns show the inputs, including the human's current map, the robot's path, and the visible grids communicated by the robot. The models predict how the human will update the maze based on this information. **Top:** The human correctly adds a distant wall instead of a nearby one, a behavior accurately predicted by NHPM. **Bottom:** The human mistakenly marks a nearby wall at the wrong location. Despite never encountering this exact scenario, NHPM successfully anticipates the error by generalizing from similar training examples.

Method	Train Loss (MBCE)	Test Loss (MBCE)	Test Accuracy (IoU @ Γ)
GLPF-Train	$1.15 \cdot 10^{-1}$	$9.10 \cdot 10^{-2}$	0.335 @ 0.38
GLPF-Test	N/A	$8.98 \cdot 10^{-2}$	0.335 @ 0.21
NHPM (Ours)	$1.36 \cdot 10^{-2}$	$\mathbf{1.43 \cdot 10^{-2}}$	**0.352**

TABLE I: Our neural human perception dynamics model (NHPM) achieves lower mean binary cross entropy error and higher prediction accuracy in the test set compared to a grid-based logistic psychometric function (GLPF). The advantage holds even when GLPF is fit directly on the test set and an optimal decision boundary Γ is searched for highest accuracy.

transition from the child node. Given a sample return q'_{sample} at child node v', the parent node's return is:

$$q_{\text{sample}} = r + \gamma \left[\delta' q'_{\text{sample}} + (1 - \delta') \frac{Q(v)}{N(v)} \right], \quad (2)$$

where δ' represents the probability of a successful transition. The term $(1 - \delta')$ accounts for failed transitions, relying instead on the current value estimate.

IV. Experiments

A. Human Perception Dynamics Evaluation

We evaluate the effectiveness of the proposed Neural Human Perception Model (NHPM) in predicting how humans perceive environmental information based on a robot's movement and transmitted images.

1) Baseline: Grid-Based Logistic Psychometric Function: We compare our method to the Logistic Psychometric Function (LPF), a standard model that relates human observer performance (e.g., detection or discrimination) to stimulus intensity [11], [12]. LPF fits a curve that predicts human response to a single source of stimulation. Hence, it lacks the expressive power to model complex spatial dependencies.

In our setting, humans receive dense stimulus from images, which LPF cannot model directly. To this end, we extend it to operate at the grid level, treating each cell in the maze independently. This adaptation, referred to as the Grid-Based LPF (GLPF), models the probability of perception updates as a function of stimulus intensity at individual cells:

$$P(y = 1 \mid x) = \rho + \frac{1 - \rho - \lambda}{1 + e^{-\beta(x - \alpha)}}, \quad (3)$$

where $P(y = 1 \mid x)$ is the probability of a human updating their perception of a grid cell given stimulus intensity x. The slope parameter β governs sensitivity to stimulus changes, while α defines the middle point of the logistic curve. ρ represents the guessing rate—the likelihood of a response in the absence of a stimulus—while λ accounts for lapses where the signal is missed even at maximum intensity.

In GLPF, stimulus intensity x is modeled as an exponentially decaying function of the distance d between the grid cell and the robot: $x(d) = e^{-\alpha d}$. We assume correctly labeled grids provide no stimulus.

2) Quantitative Evaluation: We split the dataset into training and test sets and consider three distinct test settings:

1) GLPF-Train: The psychometric function is fit on the training set to evaluate how well it generalizes to unseen environments based on prior human data.
2) GLPF-Test: To establish an upper performance bound, we fit the GLPF directly to the test set. This removes

Method	Communication (MB)	#Robot Step	#Human Guidance	Mapping Accuracy (%)
Teleoperation	86.69 ± 32.11	481.60 ± 178.40	N/A	18.77 ± 21.95
Instruction-following	75.94 ± 3.97	421.90 ± 22.08	13.40 ± 4.28	42.53 ± 7.92
IG-MCTS	2.25 ± 0.60	407.10 ± 107.80	11.50 ± 3.33	49.26 ± 12.73

TABLE II: Average task metrics reported as mean $\pm$ standard deviation. Communication assumes 180 KB/image.

the generalization gap, revealing the best-case scenario for an LPF-based approach.

3) NHPM: We train our neural network model using backpropagation, optimizing parameters by minimizing the binary cross-entropy loss between predicted and ground-truth human edits to the map.

The test loss and accuracy in Table I highlight NHPM's advantage over GLPF, demonstrating that incorporating spatial structure and contextual awareness improves human perception prediction. Even when fit on test data, GLPF remains limited by its lack of spatial expressiveness, whereas the CNN generalizes effectively from the training set.

3) Model Prediction Visualization: In Figure 3, we visualize the outputs of human perception models and highlight two representative scenarios where NHPM outperforms GLPF. In the top row, the robot transmits an image looking down a hallway, and the human adds a distant wall appearing in the center of the image. NHPM accurately predicts this behavior, but the psychometric function assigns a low probability due to the wall's distance. In the bottom row, the human mistakenly marks a wall close by, misjudging its distance from the first-person view image. Despite never encountering this exact test scenario, NHPM correctly anticipates the error by generalizing from similar patterns in the training set.

B. User Study

We investigate the following question: Can IG-MCTS, under reduced communication constraints, lower human cognitive load while maintaining task performance comparable to teleoperation and instruction-following? To address this, we conduct a within-subject user study.

Independent Variables. The study compares IG-MCTS to two baseline interaction methods:

Teleoperation: Participants manually control the robot's low-level movements by providing actions $u_t^{\text{low}} \in \mathcal{U}^{\text{low}}$ at each timestep t using keyboard arrow keys. The robot deterministically executes these actions based on the environment's low-level transition function T_{low}. The robot streams its front view RGB images, providing real-time visual feedback.

Instruction-following: Participants issue guidance as trajectories $\zeta = \langle s_t, \ldots, s_{t+n} \rangle$, specifying the desired sequence of states. The robot autonomously executes this trajectory until the trajectory is either completed or blocked. Same as in the teleoperation setting, the robot streams the front view RGB images to the human.

Dependent Measures. The study measures the following dependent variables. First, task performance metrics include the number of robot steps, the instances of human guidance provided, and mapping accuracy. Second, eye-tracking metrics serve as physiological indicators of cognitive load, including pupil diameter measurements, blink rate, and fixation shifts between areas of interests (AOIs).

Why do we choose these eye tracking metrics? Task-evoked pupillary responses have long been established as reliable indicators of mental effort [26], [27], with increased cognitive demand leading to greater pupil dilation. In this study, we choose the mean pupil diameter and the percent change in pupil dilation (PCPD) [28]. Blink rate, on the other hand, is inversely correlated with cognitive load, with higher rates indicating reduced mental effort [29], [30]. Fixation shifts between AOIs reflect visual and cognitive resource allocation during tasks [31]. Two AOIs are defined: the robot's egocentric view (left) and the top-down global maze map (right). A lower fixation shift rate suggests reduced cognitive effort needed to integrate information across the AOIs.

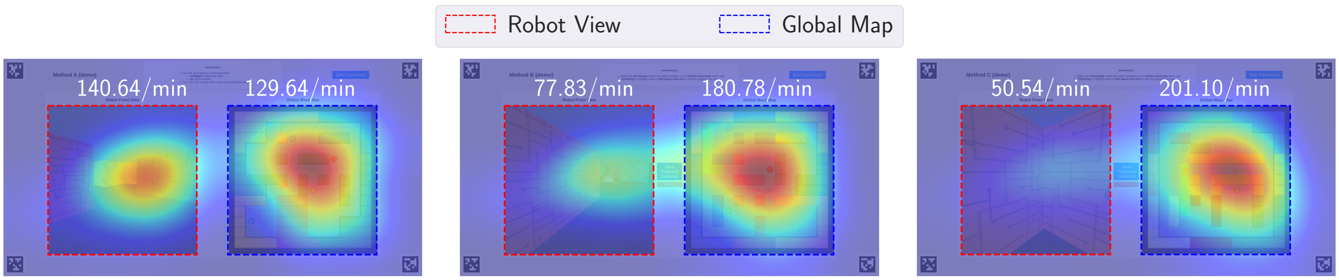
Hypotheses. We hypothesize that IG-MCTS, compared to teleoperation and instruction-following, will (**H1**) achieve better or comparable task performance and (**H2**) yield eye-tracking metrics indicative of lower cognitive load.

Participants. The study recruited 10 graduate students, with a demographic breakdown of 80% male, 20% female. The participants' average age was 25.9 years (SD = 1.91).

Procedure. Participants wore the Pupil Labs Core [32] eye tracker and completed a 5-point calibration before the study. After providing consent, they completed three sessions in random order, each corresponding to a different interaction type, to control for ordering effects [33]. To minimize confounds in pupilometry, the experiment was conducted in controlled lighting with emotionally neutral content. Each session began with a 30-second baseline pupil measurement while participants read a neutral paragraph (Section V-D.2). Participants then practiced the controls in a demo maze. This was followed by two tasks in distinct maze layouts generated with different seeds but similar structure (similarity scores: 0.843, 0.876; see Section V-D.4). The same layouts were used across sessions with 90° or 180° rotations to control difficulty and reduce memorization.

Results.

a) On H1 (Task Performance Metrics): Table II summarizes the average task metrics across all participants for two maze layouts. The results indicate that IG-MCTS requires significantly less communication compared to teleoperation and instruction-following, as it selectively transmits images at specific angles and times rather than streaming continuously. IG-MCTS also results in the fewest robot steps, indicating more efficient task execution. Additionally, our method requires less human guidance than instruction-following, demonstrating reduced reliance on human intervention. Finally, we



(a) Teleoperation (b) Instruction-following (c) IG-MCTS

Fig. 4: Aggregate heatmaps showing gaze point distributions. Dashed red box: robot view AOI; dashed blue box: global map AOI. Fixation rates in each AOI are labeled in white text above the boxes.

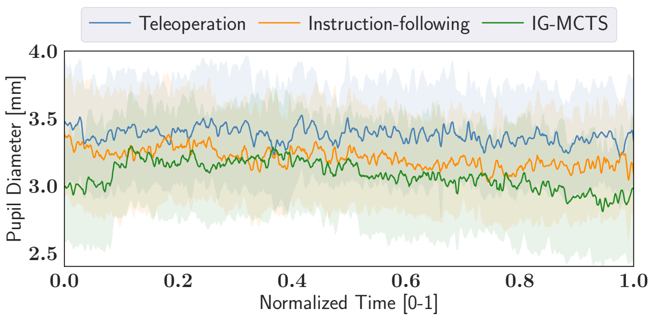


Fig. 5: Aggregated mean pupil diameter with a 95% confidence interval (Interpolation: 1000, Smoothing: 5).

Method	PCPD (%)	Blink Rate (/min)	Fixation Shift Rate (/min)
Teleoperation	32.74 ± 7.95	8.86 ± 5.88	45.69 ± 11.50
Instruction-following	20.29 ± 7.17	10.43 ± 4.46	35.18 ± 6.78
IG-MCTS	17.11 ± 7.95	12.32 ± 7.85	33.30 ± 7.39

TABLE III: Eye-tracking metrics for cognitive load: PCPD (lower is better), blink rate (higher is better), and fixation shift rate (lower is better).

achieve the highest mapping accuracy, outperforming both baselines. While these observed trends cannot be concluded as statistically significant due to the small sample size[1], the results suggest that IG-MCTS achieves at least comparable task performance to the baselines despite significantly reduced communication, providing preliminary support for **H1**.

b) On H2 (Eye tracking for Cognitive Load): Figure 5 shows the mean pupil diameter for each method, normalized to a 0-1 time scale, interpolated at 1000 points, and smoothed with a window size of 5. The plot reveals that pupil diameters are overall smallest in IG-MCTS, followed by instruction-following, and largest in teleoperation. This qualitative trend aligns with the percent change in pupil diameter (PCPD) statistics in Table III (see calculation details in Section V-D.3), which also show the lowest value for IG-MCTS, followed by instruction-following and teleoperation. Table III also shows that IG-MCTS results in a higher blink rate than instruction-following and teleoperation. Additionally, Figure 4 presents the aggregate gaze heatmaps from all participants for each method, showing the distribution of attention between the robot ego view and the global map. The heatmaps reveal a clear trend in gaze allocation across methods. Teleoperation divides attention between the ego view and the global map due to the demand for continuous monitoring and low-level control. Instruction-following alleviates the need for low-level control, thus shifting more focus to the global map. IG-MCTS concentrates gaze primarily on the global map, as it automates low-level control and provides selective ego-view snapshots. This observed trend is supported quantitatively by the fixation rates of each AOI, labeled in white text above the AOI boxes in Figure 4. This pattern is also reflected in the fixation shift rate differences listed in Table III, where IG-MCTS results in the lowest rate compared to the two baselines. These results collectively support **H2**.

[1]The eye-tracking requirement necessitated in-person recruitment, thus limiting the sample size. We leave larger-scale studies to future work.

Discussion. We observe several helpful behaviors exhibited by IG-MCTS during participant interactions. First, it efficiently reaches goals by using SLAM observations to navigate toward visible targets within its field of view, even if the human path doesn't reach the exact goal location. Second, when human guidance is suboptimal, IG-MCTS evaluates potential information gain and may pause to reorient or capture critical snapshots, ensuring the human doesn't miss key details. Finally, IG-MCTS maximizes communication efficiency by angling itself at 45° toward corners when blocked, providing views of multiple walls in a single image.

V. Conclusion

We introduced IG-MCTS, an algorithm for human-robot cooperative navigation in partially observable environments that balances autonomous exploration with informative communication. By leveraging a learned model of human perception dynamics, IG-MCTS improves interaction efficiency, reducing communication and cognitive load without sacrificing task performance compared to teleoperation and instruction-following baselines.

While promising, IG-MCTS has limitations that suggest directions for future work. Its reliance on in-context training data may limit generalization, motivating exploration of meta-learning or domain adaptation. Extending the method to continuous state and action spaces would enhance real-world applicability. Additionally, improving the interaction design—such as replacing static ego-view snapshots with short video clips—could further enrich spatial understanding.

References

[1] S. Sterz, K. Baum, S. Biewer, H. Hermanns, A. Lauber-Rönsberg, P. Meinel, and M. Langer, "On the quest for effectiveness in human oversight: Interdisciplinary perspectives," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2024, pp. 2495–2507.

[2] J. van der Waa, J. van Diggelen, L. Cavalcante Siebert, M. Neerincx, and C. Jonker, "Allocation of moral decision-making in human-agent teams: A pattern approach," in *Proceedings of the International Conference on Engineering Psychology and Cognitive Ergonomics*, 2020, pp. 203–220.

[3] J. van der Waa, S. Verdult, K. van den Bosch, J. van Diggelen, T. Haije, B. van der Stigchel, and I. Cocu, "Moral decision making in human-agent teams: Human control and the role of explanations," *Frontiers in Robotics and AI*, vol. 8, p. 640647, 2021.

[4] B. Doroodgar, M. Ficocelli, B. Mobedi, and G. Nejat, "The search for survivors: Cooperative human-robot interaction in search and rescue environments using semi-autonomous robots," in *Proceedings of the International Conference on Robotics and Automation*, 2010, pp. 2858–2863.

[5] I. R. Nourbakhsh, K. Sycara, M. Koes, M. Yong, M. Lewis, and S. Burion, "Human-robot teaming for search and rescue," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 72–79, 2005.

[6] S. Jain and B. Argall, "Probabilistic human intent recognition for shared autonomy in assistive robotics," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 9, no. 1, pp. 1–23, 2019.

[7] D. Gopinath, S. Jain, and B. D. Argall, "Human-in-the-loop optimization of shared autonomy in assistive robotics," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 247–254, 2016.

[8] L. Kangassalo, M. Spapé, N. Ravaja, and T. Ruotsalo, "Information gain modulates brain activity evoked by reading," *Scientific Reports*, vol. 10, no. 1, p. 7671, 2020.

[9] S. Chen, R. Zhao, S. Chinchali, and U. Topcu, "Human-agent coordination in games under incomplete information via multi-step intent," in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, 2025.

[10] B. Treutwein and H. Strasburger, "Fitting the psychometric function," *Perception & psychophysics*, vol. 61, no. 1, pp. 87–106, 1999.

[11] N. Prins *et al.*, *Psychophysics: A practical introduction*. Academic Press, 2016.

[12] F. A. Wichmann and N. J. Hill, "The psychometric function: I. fitting, sampling, and goodness of fit," *Perception & psychophysics*, vol. 63, no. 8, pp. 1293–1313, 2001.

[13] M. Morgan, B. Dillenburger, S. Raphael, and J. A. Solomon, "Observers can voluntarily shift their psychometric functions without losing sensitivity," *Attention, Perception, & Psychophysics*, vol. 74, pp. 185–193, 2012.

[14] M. Moniruzzaman, A. Rassau, D. Chai, and S. M. S. Islam, "Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey," *Robotics and Autonomous Systems*, vol. 150, p. 103973, 2022.

[15] R. R. Murphy, "Human-robot interaction in rescue robotics," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, no. 2, pp. 138–153, 2004.

[16] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. Van Den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3674–3683.

[17] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, M. Deitke, K. Ehsani, D. Gordon, Y. Zhu *et al.*, "Ai2-thor: An interactive 3d environment for visual ai," *arXiv preprint arXiv:1712.05474*, 2017.

[18] X. Puig, E. Undersander, A. Szot, M. D. Cote, T.-Y. Yang, R. Partsey, R. Desai, A. W. Clegg, M. Hlavac, S. Y. Min *et al.*, "Habitat 3.0: A co-habitat for humans, avatars and robots," *arXiv preprint arXiv:2310.13724*, 2023.

[19] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9068–9079.

[20] D. Fried, R. Hu, V. Cirik, A. Rohrbach, J. Andreas, L.-P. Morency, T. Berg-Kirkpatrick, K. Saenko, D. Klein, and T. Darrell, "Speaker-follower models for vision-and-language navigation," in *Proceedings of the International Conference on Neural Information Processing Systems*, 2018, pp. 3318 – 3329.

[21] J. Pasukonis, T. Lillicrap, and D. Hafner, "Evaluating long-term memory in 3d mazes," *arXiv preprint arXiv:2210.13383*, 2022.

[22] S. Palan and C. Schitter, "Prolific. ac—A subject pool for online experiments," *Journal of Behavioral and Experimental Finance*, vol. 17, pp. 22–27, 2018.

[23] N. Hansen, X. Wang, and H. Su, "Temporal difference learning for model predictive control," *arXiv preprint arXiv:2203.04955*, 2022.

[24] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *Proceedings of the International Conference on Machine Learning*, 2019, pp. 2555–2565.

[25] M. Watter, J. Springenberg, J. Boedecker, and M. Riedmiller, "Embed to control: A locally linear latent dynamics model for control from raw images," *Proceedings of the International Conference on Neural Information Processing Systems*, vol. 28, 2015.

[26] J. Beatty, "Task-evoked pupillary responses, processing load, and the structure of processing resources." *Psychological Bulletin*, vol. 91, no. 2, p. 276, 1982.

[27] E. H. Hess and J. M. Polt, "Pupil size in relation to mental activity during simple problem-solving," *Science*, vol. 143, no. 3611, pp. 1190–1192, 1964.

[28] J.-L. Kruger, E. Hefer, and G. Matthew, "Measuring the impact of subtitles on cognitive load: Eye tracking and dynamic audiovisual texts," in *Proceedings of the Conference on Eye Tracking*, 2013, pp. 62–66.

[29] J. Zagermann, U. Pfeil, and H. Reiterer, "Studying eye movements as a basis for measuring cognitive load," in *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 2018, pp. 1–6.

[30] ——, "Measuring cognitive load using eye tracking technology in visual computing," in *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization*, 2016, pp. 78–85.

[31] A. W. Joseph and R. Murugesh, "Potential eye tracking metrics and indicators to measure cognitive load in human-computer interaction research," *Journal of Scientific Research*, vol. 64, no. 1, pp. 168–175, 2020.

[32] M. Kassner, W. Patera, and A. Bulling, "Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction," in *Proceedings of the International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, 2014, pp. 1151–1160.

[33] D. W. Martin, *Doing psychology experiments*, 7th ed. Belmont, CA: Wadsworth/Thomson Learning, 2007.

Appendix

A. Task Reward Composition

The *task reward* R_{task} in our implementation incentivizes effective navigation and consists of three components:

$$R_{\text{task}}(\tau, \zeta) = R_{\text{env}}(\tau) + R_{\text{guidance}}(\tau, \zeta) + R_{\text{smooth}}(\tau). \quad (4)$$

Suppose we rewrite τ as a state sequence $(s_1, s_2, \cdots, s_n)$, we can formally define each reward as follows:

1. The navigation environment assigns a fixed reward $r_g > 0$ when the robot reaches a goal location for the first time:

$$R_{\text{env}}(\tau) = \begin{cases} r_g, & s_n \in \mathcal{G}, \\ -1, & \text{otherwise.} \end{cases} \quad (5)$$

2. A step incurs an additional cost when the robot wanders away from the human's guidance:

$$R_{\text{guidance}}(\tau) = \begin{cases} 0, & s_n \in \zeta, \\ -\log(n), & \text{otherwise.} \end{cases} \quad (6)$$

Intuitively, R_{guidance} is zero when the robot stays on the human-suggested path. The penalty is greater if the robot has taken many steps since its previous interaction with the human. In other words, it is better in the long run for the robot to stop and communicate rather than straying far.

3. The smoothness reward penalizes unnecessary revisits to previously visited states:

$$R_{\text{smooth}}(\tau) = -\sum_{i=1}^{n-1} \mathbb{1}[s_i = s_n]. \quad (7)$$

a) Additional implementation-specific design: To reduce the search horizon, lower estimation variance, and improve computational efficiency, we terminate an MCTS rollout when the agent either (a) reaches the goal or (b) performs a communication action. However, this modification introduces a bias toward shorter paths. To correct for this, we impose a communication cost of $c = 10 + n$, where n represents the number of unfulfilled states in the human guidance.

B. Algorithm Pseudocode

Procedure 2: $\textsc{Expand}(v)$

Choose an untried action $a \in \mathcal{A}$
if $a \in \mathcal{U}$ **then**
  $s = \text{last}(\tau)$
  $\tau' \leftarrow \tau \oplus T(s, a)$
  Create node v' with (τ', x)
  $\mathbb{C}(v) \leftarrow \mathbb{C}(v) \cup \{v'\}$
  Return v', False
else if $a \in \mathcal{O}$ **then**
  $x' \leftarrow F(x, (\tau, a))$
  Create node v' with (τ, x')
  Return v', True
end if

Algorithm 1 IG-MCTS

Input: human guidance ζ, previous state-visitation history τ_0, current human perception state x_0
Parameters: iterations $n = 100$, exploration constant $k = \sqrt{2}$, discount factor $\gamma = 0.99$, depth $d = 100$
Create root node v_0 with (τ_0, x_0), initialize $Q(v_0) \leftarrow 0$, $N(v_0) \leftarrow 0$, $\mathbb{C}(v_0) \leftarrow \emptyset$
for each iteration i from 1 to n **do**
  Set $v \leftarrow v_0$, stopping $\leftarrow$ False
  while v is not terminal and stopping = False **do**
    if v is fully expanded **then**
      $v \leftarrow \arg\max_{v' \in \mathbb{C}(v)} \left(\frac{Q(v')}{N(v')} + k\sqrt{\frac{\log N(v)}{N(v')}} \right)$
    else
      $v, \text{stopping} \leftarrow \textsc{Expand}(v)$
    end if
  end while
  $q \leftarrow \textsc{Rollout}(v)$
  $\textsc{BackPropagate}(v, q)$
end for
Return action of best child $c^\star = \arg\max_{c \in \mathbb{C}(v_0)} N(c)$

Procedure 3: $\textsc{Rollout}(v)$

Initialize $q \leftarrow 0$, depth $d \leftarrow 0$
while $d < T$ and (τ, x) not terminal **do**
  Sample $a \in A$
  if $a \in U$ **then**
    $\tau \leftarrow \tau \oplus T(s, a)$ where $s = \text{last}(\tau)$
  else if $a \in O$ **then**
    $x' \leftarrow F(x, (\tau, a))$
  end if
  $q \leftarrow q + \gamma^d R(\tau, \zeta, x, x')$
  $x \leftarrow x'$
  $d \leftarrow d + 1$
end while
Return q

Procedure 4: $\textsc{BackPropagate}(v, q)$

Initialize $q_{\text{sample}} = q$ and $\delta = 1$
while v is not null **do**
  Current value estimate $w = \frac{Q(v)}{N(v)}$ if $N(v) > 0$ else 0
  $q_{\text{sample}} \leftarrow r(v) + \gamma\left[\delta q_{\text{sample}} + (1 - \delta) w\right]$
  $Q(v) \leftarrow Q(v) + q$
  $N(v) \leftarrow N(v) + 1$
  $\delta \leftarrow \delta(v)$
  $v \leftarrow$ parent of v
end while

C. Mapping Dataset Details

The dataset we use for training the NHPM includes 113 trajectories with mapping accuracy $> 50\%$. We conduct a 95/5 train/test split for the evaluation of the model's generalization capability. The maze layout and robot movements in each trajectory are generated randomly without seeds, ensuring

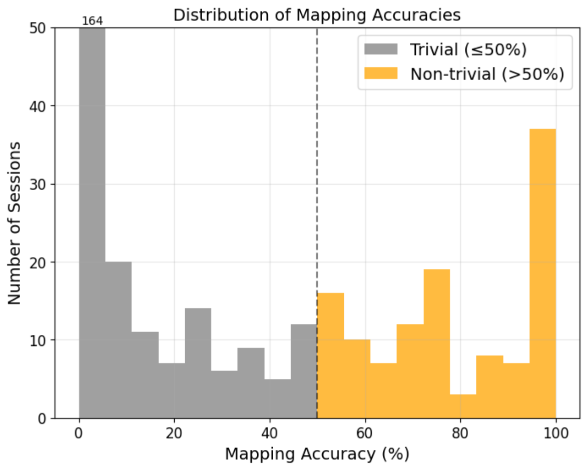


Fig. 6: The crowd-sourced mapping dataset contained low quality data (grey) which we exclude from training.

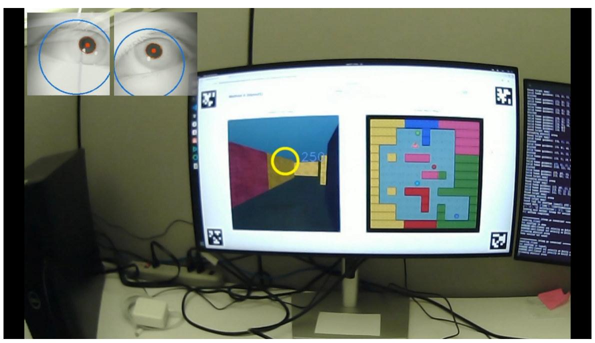


Fig. 7: A snapshot from a recorded eye-tracking session. The window in the top left shows how the software measures pupil dilation and estimates gaze location. The yellow circle indicates the estimated point of gaze.

diversity in the dataset. 254 trajectories were excluded due to low accuracy (many participants clicked through for compensation). A histogram of the overall mapping accuracy is shown in Figure 6.

Each trajectory is segmented at time steps where the robot initiates communication. This results in data chunks of the form (current human perception, robot path, robot position, sent image, human annotation). In total, we collect 1505 training segments and 137 test segments. An animated visualization of how such a data segment is constructed is available in the supplementary materials.

D. User Study Details

1) Eye Tracking Setup: Figure 7 shows a snapshot from the eye-tracking recording, where the participant is controlling the agent via teleoperation. The AR tags on the screen assist the eye-tracking device in localizing the plane of interest. The window in the top left visualizes how the software measures pupil dilation and estimates gaze location. The yellow circle marks the estimated point of gaze. A full video of the session is available in the supplementary materials.

2) Baseline Pupil Diameter Measurement: To account for individual differences in pupil sizes, we ask each participant to read a brief text paragraph at the beginning of each session. The eye-tracking data from this period is used to calculate the mean pupil diameter as the baseline for the session's PCPD. We drafted these paragraphs to ensure comparable length and maintain neutral content.

Text Before Method A (Teleoperation)

Making a sandwich begins by picking your favorite type of bread. You can spread butter, mayonnaise, or other condiments before adding a layer of vegetables, meat, or cheese. Once the ingredients are in place, press the slices together gently. Preparing a sandwich is a simple task, but it's also a quick and satisfying way to create a meal.

Text Before Method B (Instruction-Following)

Washing dishes starts by filling the sink with warm, soapy water. Plates, bowls, and utensils are scrubbed clean with a sponge to remove food residue. Once clean, they are rinsed under running water and placed on a rack to dry. While it's a routine chore, it's also a small step toward keeping the kitchen tidy and organized.

Text Before Method C (IG-MCTS)

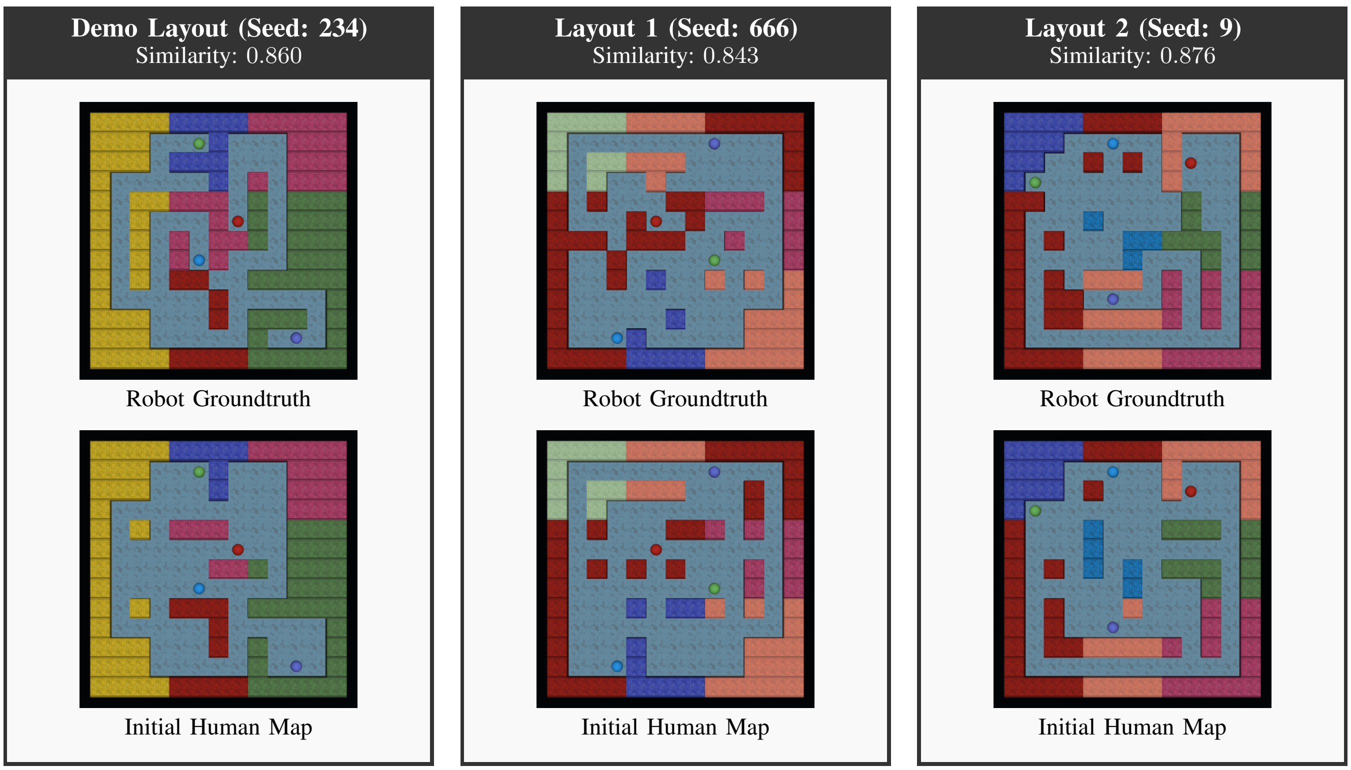
Sitting in a chair can be a relaxing moment during a busy day. You adjust your position to get comfortable, letting your body rest as you settle in. Sometimes, it's a chance to pause and think quietly. Whether you're sitting to read, work, or simply take a break, it's a small but familiar part of daily life.

3) PCPD Calculation: The percent change in pupil diameter (PCPD) is calculated as follows:

$$\text{PCPD}_t = \frac{\text{pupil diameter}_t - \text{baseline diameter}}{\text{baseline diameter}}. \tag{8}$$

Here, *baseline diameter* is the average pupil diameter recorded during the baseline period, as detailed in Section V-D.2. For each recording, we compute the mean and standard deviation of PCPD over time. These statistics are then aggregated, and we report the averages in Table III.

4) Maze Layouts: To ensure consistency and reproducibility across participants, we generated three distinct maze layouts using fixed random seeds on a Linux system. By fixing the seeds and standardizing the platform, we ensured that all participants encountered identical navigation challenges, enabling fair comparisons across conditions. Visualizations of the three maze layouts are provided on the following page.


Demo Layout (Seed: 234)
Similarity: 0.860
Robot Groundtruth
Initial Human Map
Layout 1 (Seed: 666)
Similarity: 0.843
Robot Groundtruth
Initial Human Map
Layout 2 (Seed: 9)
Similarity: 0.876
Robot Groundtruth
Initial Human Map