

Visual Cue Guided Video Planning for Generalizable Robot Navigation

Hojin Lee¹, Sizhe Lester Li², Maximilian Hilger¹, Susie Lu², Achim J. Lilienthal^{1,3},
Vincent Sitzmann^{2,†}, and Daniel A Duecker^{1,†}

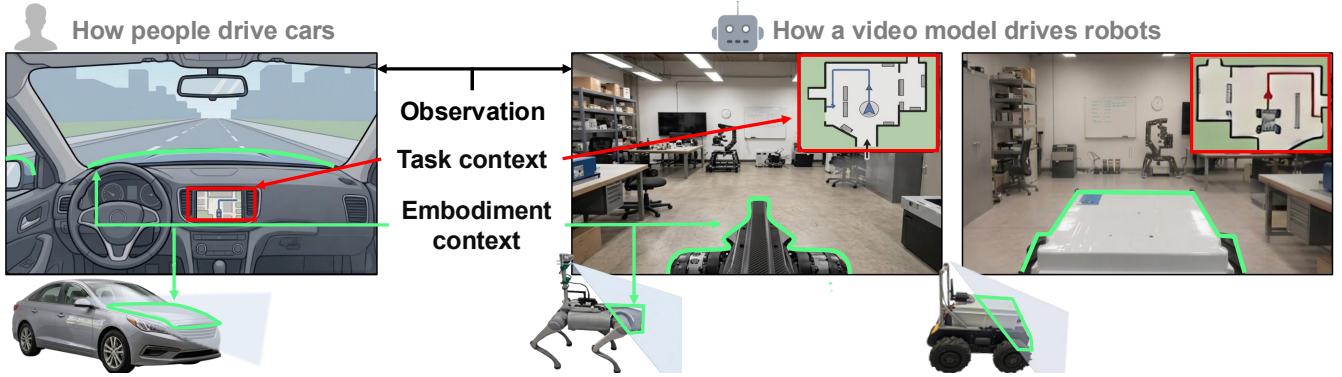


Fig. 1: **Visual cue guided video planning for generalizable robot navigation (CueNav)**. Similar to human driving, a video planner can benefit from task and embodiment context expressed through the visual observation. We illustrate these forms of context using a map of the surroundings and a partial view of the robot body.

Abstract—Generative video models can serve as a promising backbone for robot navigation by predicting future observations as video plans. Recent approaches often condition video planning on short-horizon guidance and recover geometric waypoints through scene reconstruction, leaving longer-horizon planning and precise video-to-action translation less explored. We present *CueNav*, a video model-based navigation framework combining visual cue guided video planning with an embodiment-specific Inverse-Dynamics Model (IDM). As visual cues, we use a Bird’s-Eye View (BEV) map to convey global task context and retain part of the robot body in the egocentric observation to expose embodiment context. These cues guide the video planner, while the IDM translates dense flow fields extracted from the video plan into robot actions. With the visual cue encoding global task context, CueNav achieves nearly $2\times$ higher success in maze navigation than planning without the cue. The body-aware view with the IDM enables precise navigation with 70 % success in a narrow passage where comparison methods largely fail to complete the task. We further demonstrate zero-shot semantic-conditioned navigation and deployment of the same video planner across different robot platforms. Our results show that visual cue-guided video planning with embodiment-specific action grounding paves the way toward a generalizable navigation framework for longer-horizon planning and embodiment-aware control. Additional results and code are available on our project website*.

[†]Equal advising.

¹Munich Institute of Robotics and Machine Intelligence (MIRMI), Technical University of Munich, 80797 Munich, Germany.

²Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139, USA.

³Centre for Applied Autonomous Sensor Systems (AASS), Örebro University, 72081 Örebro, Sweden.

This work was supported by the Bavarian State Ministry of Science and the Arts (StMWK) through the MIT-TUM Collaboration (grant no. 151223031).

Vincent Sitzmann and Sizhe Lester Li were supported by the National Science Foundation under Grant No. EEC 2330040 and the CAREER program under 2543631.

*Project website: <https://cuenav.github.io>

I. INTRODUCTION

General-purpose robot navigation requires robots to operate in unseen environments and respond to diverse task specifications. Recent Vision-Language Navigation (VLN) methods have made substantial progress toward this goal by building on large pretrained Vision-Language Models (VLMs), which are finetuned into Vision-Language-Action Models (VLAs) [1]. These models provide strong semantic reasoning and can support a range of navigation tasks within a unified framework [2]–[4]. However, VLMs encounter two problems when being deployed in VLN: They require large amounts of training data to adapt to the robot domain, and fine-tuning on robot-action data limits their generality across tasks, environments, and robot platforms [5].

Video models offer a promising alternative through broad priors from internet-scale video data [6]. Given recent observations and task instructions, they can predict how the scene should evolve as the task is accomplished. This allows navigation behavior to exploit semantic and motion priors learned from large-scale video data without directly committing the planner to the action space of a specific robot. Recent works have shown that generated future observations can provide useful navigation guidance and generalize to novel scenes and goals [6], [7].

Still, several challenges remain for video models in navigation:

- Existing approaches require granular instructions, e.g., turning left or right [6]. This requirement for local directional guidance does not fully utilize the capabilities of video models.
- Existing video planners do not account for the robot’s physical characteristics [8]. Knowing the robot’s size and kinematics is required for precise and accurate

planning in constrained spaces.

- Existing methods map video plans to actions via waypoints, which require a dedicated downstream controller to convert them into control commands that the robot can feasibly execute [6]–[8]. As a result, the generated visual motion is not directly grounded in the robot’s kinematics and dynamics.

To address these challenges, we introduce CueNav, a video-based navigation framework that guides video planning through task and embodiment visual cues and grounds the resulting visual motion into continuous robot control. Our approach is built around two design principles:

- **Visual cues for task and embodiment aware planning.** We enable longer-horizon tasks and embodiment-awareness by providing visual cues to the video model. For navigation from high-level task specifications, we embed a BEV map into the visual observation to provide global task context (See Fig. 1). For embodiment-aware navigation, we retain part of the robot body within the egocentric view, exposing cues about its geometry and spatial relationship to nearby obstacles and free space.
- **Translate visual motion through an IDM.** We infer robot actions from dense flow fields in the generated video using an embodiment-specific IDM. The IDM captures the kinematics and dynamics of each robot, enabling precise execution while preserving a shared video planner across embodiments.

We evaluate CueNav through a series of simulation and real-world experiments and show that (i) the video planner enables zero-shot semantic-conditioned navigation; (ii) global task context provided through the observation enables planning beyond local directional guidance and generalization to larger, unseen environments; (iii) embodiment-visible observations and the embodiment-specific flow-based IDM improve precise navigation in geometrically constrained spaces; and (iv) the same video planner can be deployed across wheeled and legged robot platforms using embodiment-specific IDMs. Together, these results suggest that visual cue-guided video planning, paired with a flow-based IDM, offers a promising direction for general robot navigation using video models.

II. RELATED WORK

A. Foundation Models for Vision-Language Navigation

Recent methods build on large pretrained VLAs to support multiple navigation tasks within a unified framework [3], [9]. These models have demonstrated strong semantic reasoning across tasks such as instruction following [10], object-goal navigation [11], and person following [12].

Despite this progress, learning robot-specific control still requires large amounts of paired observation-action data. Recent navigation foundation models are trained at correspondingly large scale, from 8M navigation samples in NavFoM [13] to 30M supervised samples in ABot-N1 [3]. Since collecting comparable amounts of real-world robot data is costly, much of this data is generated in simulation

or reconstructed environments, which can introduce sim-to-real gaps and couple the learned policy to particular action spaces and embodiments.

Video models offer a promising alternative by learning semantic and motion priors from internet-scale videos without requiring robot action labels. Rather than predicting robot actions directly, they allow navigation behavior to be planned in visual space, while robot-specific control can be learned separately.

B. Video Models for Robot Navigation

Recent works use video models as visual planners for robot navigation, differing mainly in how future behavior is specified and translated into robot motion. ImagiNav [6] and ImagineUAV [8] condition future visual trajectories on navigation instructions, but their evaluations largely rely on local motion guidance, leaving task-level navigation without explicit directional cues less explored. NavDreamer [7], ImagiNav [6], and ImagineUAV [8] recover camera motion or geometric waypoints from generated video for downstream control. This introduces additional geometric estimation and tracking stages, while the visual planner itself does not explicitly account for the executing embodiment, although robot size, shape, and motion capabilities affect which behaviors are physically feasible. DreamToNav [14] generates videos in a third-person perspective, containing the robot environment. However, to recover the motion of the robot, both camera pose and robot pose in the camera frame need to be estimated, which complicates the motion reconstruction. SparseVideoNav [15] avoids explicit geometric reconstruction by predicting actions from generated future visual features. However, this still requires paired visual and action supervision, retaining limitations similar to VLAs.

In contrast, CueNav uses global task context to support navigation from high-level objectives without explicit local directional guidance, while an embodiment-specific flow-based IDM grounds generated visual motion directly into continuous robot commands without explicit geometric reconstruction or joint video-action training.

III. MAIN METHOD

A. Problem Formulation

A video model $\mathcal{P}$ produces a short-horizon visual plan for navigation. Given a recent observation history $\mathbf{I}_{t-(N-1):t} = \{\mathbf{I}_{t-(N-1)}, \dots, \mathbf{I}_t\}$ and a navigation prompt g , the video model samples a sequence of future observations as

$$\hat{\mathbf{I}}_{t+1:t+M} \sim \mathcal{P}(\cdot \mid \mathbf{I}_{t-(N-1):t}, g), \quad (1)$$

where N denotes the observation history length and M is the prediction horizon. The predicted future observations encode the intended navigation motion. To execute this motion, the visual plan must ultimately be translated into a sequence of continuous control commands. Hence, this video-to-navigation control formulation raises two central questions concerning how to generate navigation-relevant visual plans and how to ground them in executable robot control.

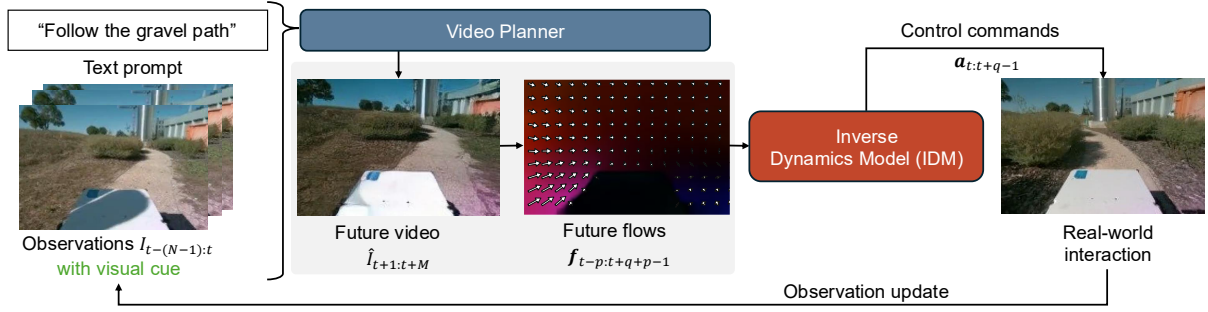


Fig. 2: Overview of CueNav. Given recent visual observations with embodiment or global task context and a text prompt, the video planner predicts short-horizon future observations. The resulting flow fields from the predicted video are mapped to continuous robot actions by the IDM, followed by closed-loop replanning from new observations.

B. Visual Cue Guided Planning with a Video Model

The video planner predicts short-horizon future observations conditioned on recent observations and a navigation prompt. A key design choice is how the visual input is constructed. To support generalizable navigation, we expose task and embodiment context directly through visual cues.

Visual Cue Design. We consider two visual cue designs. First, for navigation from high-level goal specifications, we embed a BEV map into the observation (Fig. 8(a)). The map contains the robot and goal locations while leaving the path between them unspecified, providing global situational awareness beyond the egocentric view. Second, for embodiment-aware navigation, we position the camera such that parts of the robot body remain visible in the egocentric observation (Fig. 3). This exposes cues about the robot’s geometry and its spatial relationship to nearby obstacles and free space, without requiring a separate embodiment representation or an explicit geometric description.



Fig. 3: Embodiment-aware observation with part of the robot body visible as a visual cue.

Action-free Video Model Post-training. We build the video planner on the open-weight Wan2.2-5B video diffusion transformer [16] and adapt the pretrained TI2V backbone for video-to-video prediction. We choose this model for its open availability and practical balance between model capacity and onboard computational cost, while CueNav can in principle be used with other video-model backbones. We use navigation videos only and do not require robot action labels. During post-training, each navigation video is divided into N context frames and M future frames, with the visual cues present in all frames. The context frames and navigation prompt condition the model. The training loss uses the same generative prediction objective as during video model pretraining and is applied only to future frames. We follow a diffusion-forcing formulation for autoregressive video prediction [17].

At deployment, the planner predicts a short-horizon visual future from the latest observation history. The predicted video frames are passed to the flow-based IDM described in the following section, which maps the predicted visual motion to continuous robot commands. After executing these commands, newly observed frames are used to update the observation history for the next planning step, enabling closed-loop receding-horizon planning. An overview of the proposed framework is shown in Fig. 2.

C. Flow-based Inverse-Dynamics Model

The predicted video describes how the scene should evolve but does not directly specify executable robot actions, $\mathbf{a}_t \in \mathbb{R}^n$. We infer these actions from the predicted frames using an IDM, π_{IDM} . The IDM uses dense flow fields between consecutive frames rather than the predicted RGB frames directly, inspired by the use of flow for action inference in [18]. Specifically, at time step t , it predicts a sequence of q robot actions as

$$\mathbf{a}_{t:t+q-1} = \pi_{\text{IDM}}(\mathbf{f}_{t-p}, \dots, \mathbf{f}_{t+q+p-1}), \quad (2)$$

where $\mathbf{a}_{t:t+q-1} = \{\mathbf{a}_t, \dots, \mathbf{a}_{t+q-1}\}$ and $\mathbf{f}_i \in \mathbb{R}^{H \times W \times 2}$ denotes the dense flow field computed from the frame pair at time steps i and $i+1$ of the combined observed-and-predicted sequence $\{\mathbf{I}_{t-(N-1):t}, \hat{\mathbf{I}}_{t+1:t+M}\}$. The $p \geq 1$ additional flow fields at both ends of the window provide temporal context, allowing the IDM to account for system dynamics such as delayed control responses. In total, the window contains $q + 2p$ flow fields, which requires $p + q \leq M$ and $p \leq N - 1$. We compute the flow fields using AllTracker as an off-the-shelf flow estimator [19].

IDM with Spatiotemporal Transformer. The IDM consists of a shared convolutional encoder followed by a spatiotemporal transformer. Each flow field is encoded by the convolutional encoder and adaptively average-pooled to a fixed spatial grid. The resulting spatial tokens from all flow fields are processed jointly by the transformer with 3D rotary positional embeddings over space and time. A feed-forward network maps the resulting features to the q predicted actions. The IDM is trained to regress the corresponding robot actions using a mean-squared error objective.

Embodiment-specific Training. We train a separate IDM for each robot platform using paired flow fields and robot

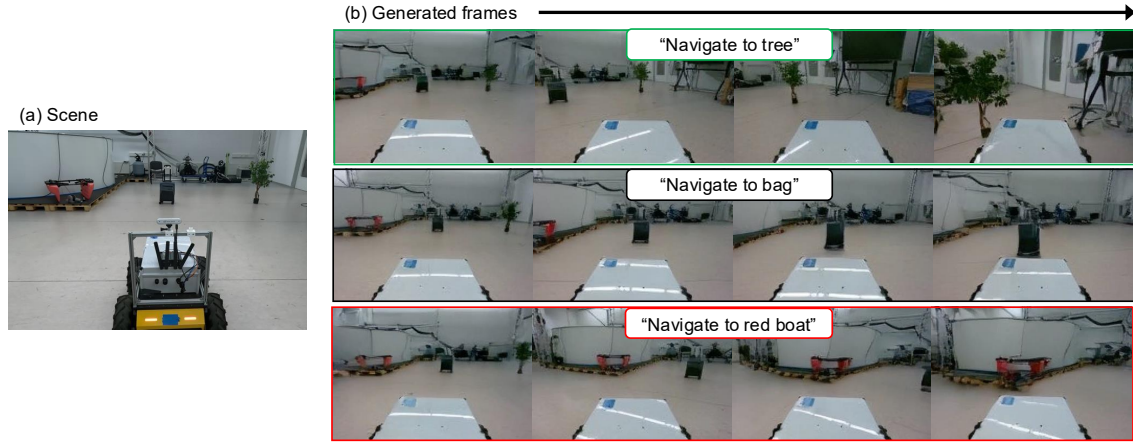


Fig. 4: Zero-shot semantic-conditioned visual planning. (a) Initial scene. (b) Generated future observations conditioned on different semantic goal prompts.

actions, where the flow fields are computed from RGB observations recorded during navigation. For wheeled robots, the action space is $\mathbf{a} = (v_x, \omega_z) \in \mathbb{R}^2$, where v_x and ω_z denote the forward linear velocity and yaw angular velocity, respectively. For legged robots, we use $\mathbf{a} = (v_x, v_y, \omega_z) \in \mathbb{R}^3$, additionally including the lateral linear velocity v_y .

Closed-loop Execution. At deployment, the flow fields from the predicted frames are passed to the IDM to obtain a sequence of robot actions. We execute $\mathbf{a}_{t:t+q-1}$ actions before acquiring new observations and querying the video planner again. This receding-horizon procedure, summarized in Algorithm 1, updates the visual plan using real observations and translates the predicted visual motion into robot-specific control inputs.

Algorithm 1 CueNav closed-loop navigation

Initialize: video planner $\mathcal{P}$, IDM π_{IDM} , text prompt g , video context length N , video prediction horizon M , action horizon q , and temporal padding length p .

- 1: **while** not terminated **do**
 - 2: Acquire recent observations $\mathbf{I}_{t-(N-1):t}$
 - 3: Sample future video frames using (1)
 - 4: Compute flow fields $\mathbf{f}_{t-p:t+q+p-1}$
 - 5: Infer robot actions using (2) and execute the actions
 - 6: Append the newly acquired observations
 - 7: $t \leftarrow t + q$
 - 8: **end while**
-

IV. EXPERIMENTS

To validate the main design choices of CueNav, we organize our experiments around three questions.

Semantic-conditioned closed-loop navigation. Can CueNav leverage the generalization capability of the video model to enable semantic-conditioned closed-loop navigation in real environments? (Sec. IV-B)

Precise embodiment-aware navigation. Do embodiment cues in the visual observation improve navigation in geometrically constrained spaces? (Sec. IV-C)

Planning beyond local directional guidance. Does global task context enable navigation beyond local directional guidance? (Sec. IV-D)

A. Experimental Setup

Robotic Platforms and Data Collection. We evaluate CueNav on two ground robot platforms: a Clearpath Husky A300 wheeled robot and a Unitree Go2 quadruped, both controlled through their low-level velocity controllers. Both robots are equipped with a RealSense D455 camera, of which we use only RGB images. For each platform, we collect approximately two hours of navigation data in indoor and outdoor environments around industrial buildings, consisting of RGB observations paired with the executed robot actions. The video planner is post-trained using only RGB observations from both platforms, without requiring action labels. A separate IDM is trained for each platform using paired flow fields and robot actions from the corresponding robot navigation data. We use the Husky A300 for semantic-conditioned and precise embodiment-aware navigation experiments, where its non-holonomic motion makes constrained navigation particularly challenging. The Unitree Go2 is used to demonstrate cross-embodiment deployment of the shared video planner.

Implementation Details. We build the video planner on the pretrained Wan2.2-5B video model [16], operating at a spatial resolution of 192×128 pixels (width $\times$ height) and a frame rate of 10 Hz. The pretrained Variational Autoencoder remains frozen during post-training, while Low-Rank Adaptation [20] is applied to the diffusion transformer. Post-training is performed on a single AMD MI300X GPU using mixed-precision training where supported.

During inference, the model conditions on $N = 21$ context frames and generates $M = 16$ future frames as the visual plan. CueNav runs fully onboard on an NVIDIA Jetson Thor, where generating one visual plan and inferring the corresponding robot actions with the IDM takes approximately 4 s. For both robot platforms, we use $p = 1$ temporal padding flow fields and execute the $q = 15$ predicted actions after each video-planning step.

Baselines and Ablations. We compare *CueNav* against two publicly available vision-language navigation methods, *StreamVLN* [21] and *InternVLA-NI* [1]. We select these as strong recent VLN baselines with publicly released implementations and checkpoints. To our knowledge, there are no publicly available models and checkpoints for closely related video-model-based navigation methods. Both baselines are evaluated using their released checkpoints and egocentric observations consistent with their respective training setups. For *InternVLA-NI*, when the policy requests a camera look-down action that cannot be executed by our fixed-camera platform, we instead rotate the robot to acquire a new observation. For *StreamVLN*, we additionally attempted fine-tuning with our in-domain navigation data. Since this degraded performance relative to the released checkpoint, we report results using the released model without additional fine-tuning. To isolate the effect of embodiment cues, we also train a *CueNav* variant using a pure egocentric observation in which the robot body is not visible, denoted as *CueNav (w/o body)*. To evaluate the proposed flow-based IDM, we additionally consider *CueNav (w/o IDM)*, which uses the same visual observations and video planner as *CueNav* but replaces the IDM with geometry-based grounding similar to that in [6]. We use MAST3R [22] to reconstruct the scene and recover the camera trajectory from the generated video, which is then tracked by a simple heuristic controller. All other training and inference settings are kept unchanged.

B. Semantic-conditioned Goal Navigation

We first evaluate whether *CueNav* can leverage the generalization capability of the video planner for semantic-conditioned closed-loop navigation in real environments. The robot is given a semantic target description and must navigate to the corresponding object.

Task Setup and Evaluation Metrics. We place three unseen semantic targets in the environment and prompt the robot to navigate toward each target from the same initial position. Each target is evaluated over 10 runs, resulting in 30 trials per method.

We report success rate (SR), success weighted by path length (SPL), final navigation error (NE), and trajectory length (TL). A trial is considered successful when the robot terminates within 1.5 m of the target center. SR measures the fraction of successful trials, while SPL is defined as $SPL = \frac{1}{E} \sum_{i=1}^E S_i \frac{\ell_i}{\max(p_i, \ell_i)}$, where E is the total number of trials, $S_i \in \{0, 1\}$ indicates success, ℓ_i is the shortest-path distance from the start to the target, and p_i is the executed path length. NE measures the final distance between the robot and the target center, and TL denotes the total distance traveled during the trial.

Results. Table I summarizes the zero-shot semantic-conditioned navigation performance. *CueNav* achieves the highest SR and SPL. *CueNav (w/o body)* remains competitive and attains the lowest final navigation error, indicating that the video planner retains strong semantic goal-conditioning capability even without an embodiment-visible observation. In contrast, even with the same video planner, *CueNav (w/o*

TABLE I: Zero-shot semantic-conditioned goal navigation performance over 30 trials per method. NE and TL are reported as mean $\pm$ standard deviation.

Method	SR $\uparrow$	SPL $\uparrow$	NE [m]	TL [m]
StreamVLN [21]	0.467	0.418	1.88 ± 1.58	3.69 ± 0.97
InternVLA-NI [1]	0.733	0.665	1.22 ± 1.03	3.62 ± 1.29
<i>CueNav (w/o body)</i>	0.900	0.833	0.70 ± 0.53	3.35 ± 0.56
<i>CueNav (w/o IDM)</i>	0.667	0.615	1.25 ± 0.80	3.31 ± 0.78
<i>CueNav</i>	0.933	0.913	0.84 ± 0.71	3.02 ± 0.72

IDM), which uses geometry-based camera-pose reconstruction, substantially reduces SR and SPL, highlighting the importance of accurate visual-motion grounding. Overall, these results suggest that semantic-conditioned video-based navigation remains effective without an embodiment-visible observation, but depends more strongly on faithful grounding of the generated visual motion into robot actions. Fig. 4 shows the future observations generated by *CueNav* during closed-loop navigation toward different semantic targets.

C. Precise Embodiment-aware Navigation

We next evaluate whether including an embodiment cue by keeping part of the robot body visible improves precise navigation in geometrically constrained spaces.

Task Setup and Evaluation Metric. We construct narrow navigation tracks using boxes and require the Husky robot to traverse the constrained passage without becoming immobilized by collisions. These narrow-track environments are not included during post-training, making this a zero-shot precise-navigation task. We consider two track widths, 1.5 m and 1.0 m, as shown in Fig. 5. The Husky is 0.7 m wide, making the 1.0 m track substantially more constrained. Each condition is evaluated over 10 runs. A trial terminates when the robot reaches the end of the track or becomes immobilized upon contact with surrounding obstacles. We evaluate the maximum normalized progress reached along the track before termination, where 0 denotes the start, and 1 denotes successful completion.

Results. Fig. 7 shows that *CueNav* maintains high progress as the track becomes more constrained. It completes all trials in the 1.5 m setting and 70% of trials in the 1.0 m setting, while the comparison methods frequently terminate before reaching the end. The ablations indicate that both the embodiment-visible observation and the proposed IDM contribute to precise navigation. Removing the embodiment cue degrades performance in the narrower passage, while replacing the IDM with geometry-based grounding also reduces progress despite using the same video planner and visual observation. Fig. 5 provides a qualitative example of *CueNav*’s generated visual plan and the corresponding real observations during navigation. Fig. 6 further illustrates the effect of action grounding. *CueNav* closely realizes the generated visual plan, whereas *CueNav w/o IDM* shows visible deviations between the generated and realized observations, indicating less accurate translation of the visual plan into robot motion.

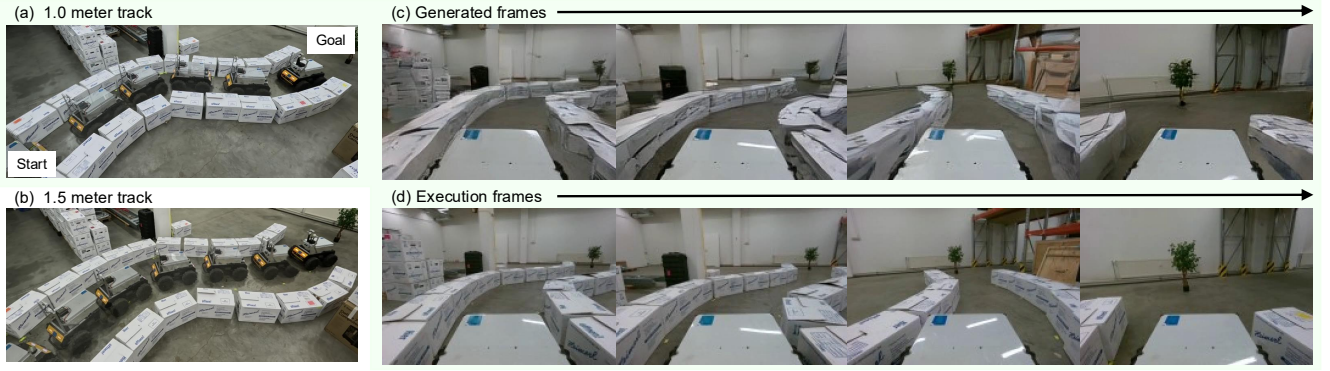


Fig. 5: Zero-shot precise embodiment-aware navigation through narrow tracks. (a,b) CueNav trajectories overlaid on the 1.0 m- and 1.5 m-wide track scenes, respectively. (c) Generated future observations for a representative rollout in the 1.0 m track. (d) Corresponding real observations at matched timesteps.

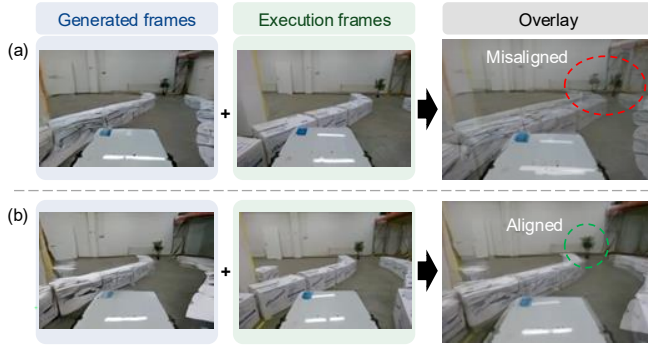


Fig. 6: Visual-plan realization with and without the proposed IDM. (a) CueNav w/o IDM shows visible misalignment between generated and realized observations. (b) CueNav maintains close alignment.

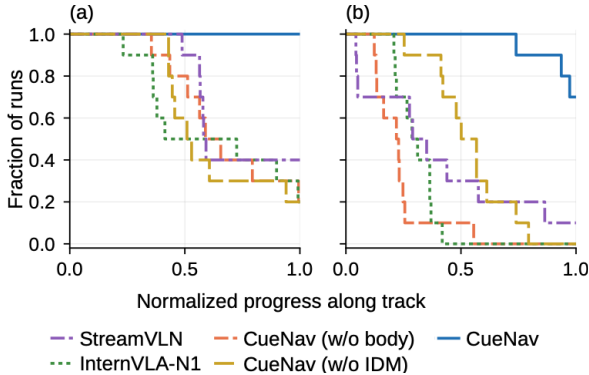


Fig. 7: Normalized progress along the narrow-track task over 10 runs per method. Each curve shows the fraction of runs reaching at least a given progress. (a) 1.5 m track width. (b) 1.0 m track width.

D. Planning beyond Local Guidance with Global Task Context

We use maze navigation as a controlled setting in which reaching a distant goal requires a sequence of decisions over multiple local observations. This allows us to evaluate whether providing global task context helps the video planner make longer-horizon navigation decisions.

Simulation Environment and Task Setup. We use sim-

ulated maze environments in DeepMind Lab [23] with grid sizes ranging from 3×3 to 6×6 . The agent is controlled by continuous forward linear velocity and yaw rate commands at 10 Hz. The agent’s initial position and a red-box goal are randomly placed in the maze, and the agent navigates to the goal without receiving turn-by-turn directional commands.

The visual input to *CueNav* combines the egocentric view with a BEV map indicating the agent and goal locations while leaving the solution path unspecified (See Fig. 8(a)). The two views are tiled into a single 832×480 pixel observation (width $\times$ height), providing the video planner with global task context in addition to the egocentric view. For action grounding, the IDM is trained on flow fields computed from the egocentric view portion of the observation, keeping the BEV cue specific to the video planner.

Training and Evaluation Protocol. Training data is collected from 300 successful goal-reaching trajectories in randomly generated 3×3 mazes. In each maze, the agent follows a solution trajectory generated by a local waypoint controller, and the resulting observations are used to post-train the video planner. The planner is therefore exposed only to 3×3 mazes during post-training and is evaluated zero-shot on larger maze sizes up to 6×6 . We evaluate 20 randomly generated episodes for each maze size and report the fraction of trials that reach the goal as a function of simulation time.

Ablation on Global Task Context. To isolate the contribution of the visual cue providing global task context, we compare *CueNav* with *CueNav (w/o map)*, a variant that receives only the egocentric view without access to the BEV map, as shown in Fig. 8(b).

Results. Fig. 9 shows that *CueNav* with global task context consistently outperforms *CueNav (w/o map)* across all maze sizes. The performance gap is already visible in the 3×3 training setting and becomes increasingly pronounced for larger 5×5 and 6×6 mazes, suggesting that global task context becomes more beneficial as navigation requires decisions over longer horizons. Although performance decreases with increasing maze size, *CueNav* retains 55% success in unseen 6×6 mazes, compared with 30% for *CueNav (w/o map)*.

Fig. 8 further illustrates representative trajectories across

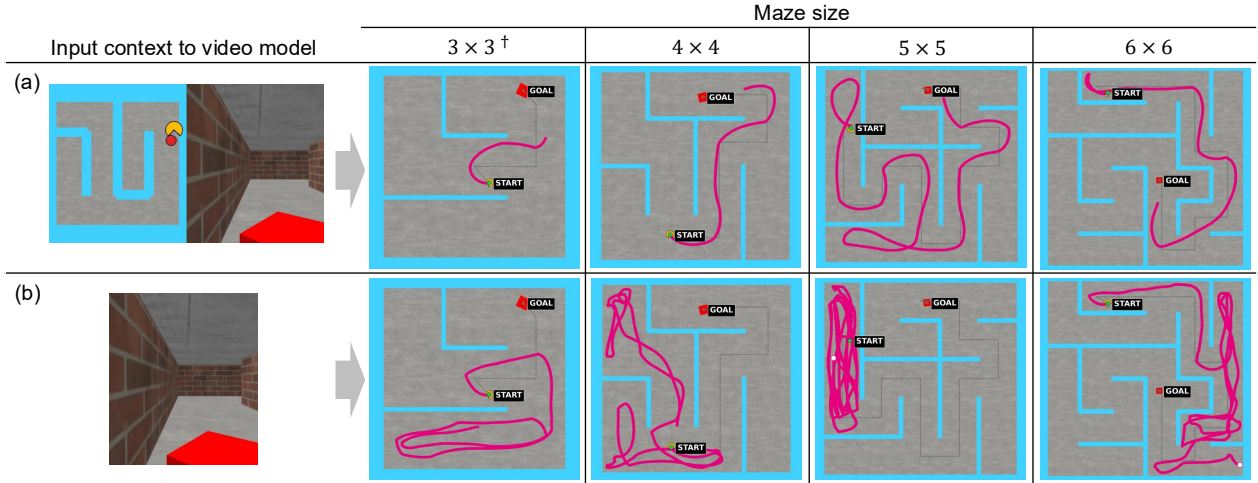


Fig. 8: Maze navigation trajectories (magenta) with (a) global task context provided as a visual cue and (b) egocentric view only. Both variants are trained on 3×3 mazes ($\dagger$ denotes the training distribution) and evaluated zero-shot on larger mazes up to 6×6 .

increasing maze sizes. With the global map context, *CueNav* produces more goal-directed trajectories, while *CueNav* (w/o map) exhibits more local wandering and less consistent progress toward the distant goal. Together, these results indicate that providing global task context through visual observation enables planning beyond local directional guidance and supports zero-shot generalization to larger unseen environments.

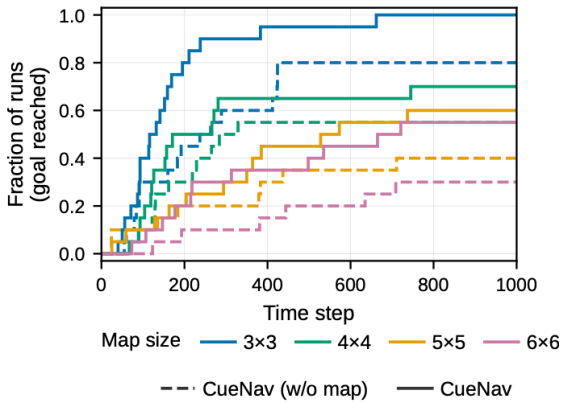


Fig. 9: Goal-reaching performance across maze sizes. Curves show the fraction of 20 runs that have reached the goal by each timestep.

E. Case Study: Cross-embodiment Deployment with a Shared Video Planner

Beyond the controlled evaluations above, we examine cross-embodiment deployment by reusing the same video planner on the Unitree Go2 while training a separate IDM for the platform. Compared with the Husky A300, the Go2 has a different morphology and action space, while embodiment-specific control is handled by the platform-specific IDM.

Fig. 10 shows a zero-shot deployment on the Go2 in an environment not seen during post-training, where the robot traverses randomly placed obstacles without collision. The

embodiment-visible observation provides the planner with visual information about the executing robot and its spatial relation to nearby obstacles, allowing the shared planner to generate behavior appropriate to the different embodiment. This case study illustrates that *CueNav* can reuse a common video planner across embodiments while adapting execution through an embodiment-specific IDM.

V. CONCLUSION AND FUTURE WORK

We presented *CueNav*, a video-based navigation framework that conditions a generative video planner on task- and embodiment-relevant visual cues and translates predicted visual motion into continuous robot actions using a flow-based IDM. Our results show that this formulation enables navigation beyond local directional guidance and precise embodiment-aware control while supporting generalizable navigation across different robot platforms. Despite these results, several limitations remain. First, video-planning inference remains computationally expensive, limiting the replanning rate during deployment. Few-step or latent-space distillation could help alleviate this bottleneck. Second, the current video model predicts only a short horizon and conditions on a finite observation window, limiting longer-horizon reasoning and memory. Longer-term visual context or hierarchical memory could extend the planner beyond its current temporal horizon. Finally, the IDM is tied to a fixed camera configuration, and its flow-to-action mapping may not transfer directly across different camera poses. Camera-conditioned or adaptive IDMs could improve generalization across sensor configurations.

ACKNOWLEDGMENT

The authors gratefully acknowledge funding and computational resources provided by AMD through the MIT AI hardware program and the AMD University Program’s AI & HPC Cluster. The authors acknowledge the use of ChatGPT (GPT-5.5) and Claude (Opus 5) for language editing and figure/table formatting; the images in Fig. 1 were generated



Fig. 10: Cross-embodiment deployment of CueNav on the Unitree Go2. (a) Execution frames during zero-shot navigation in a cluttered environment. (b) Corresponding generated future observations, demonstrating reuse of the shared video planner on a different robot embodiment.

with ChatGPT. All AI-assisted content was verified by the authors.

REFERENCES

- [1] M. Wei, C. Wan, P. Peng, X. Yu, Y. Yang, D. Feng, W. Cai, C. Zhu, T. Wang, J. Pang *et al.*, “Ground slow, move fast: A dual-system foundation model for generalizable vision-language navigation,” in *International Conference on Learning Representations*, vol. 2026, 2026, pp. 12 380–12 396.
- [2] W. Liu, H. Zhao, C. Li, J. Biswas, B. Okal, P. Goyal, Y. Chang, and S. Pouya, “X-mobility: End-to-end generalizable navigation via world modeling,” in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 7569–7576.
- [3] R. Gong, Y. Guo, J. Hu, J. Kong, X. Leng, T. Li, W. Li, F. Liu, Z. Liu, J. Lu *et al.*, “Abot-n1: Toward a general visual language navigation foundation model,” *arXiv preprint arXiv:2607.10383*, 2026.
- [4] A. Sridhar, D. Shah, C. Glossop, and S. Levine, “Nomad: Goal masked diffusion policies for navigation and exploration,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 63–70.
- [5] S. Grover, A. Gopalkrishnan, B. Ai, H. I. Christensen, H. Su, and X. Li, “Enhancing generalization in vision-language-action models by preserving pretrained representations,” *arXiv preprint arXiv:2509.11417*, 2025.
- [6] J. Chen, Y. Cai, Y. Wang, R. Bai, Y. Cao, J. Li, Y. W. Yun, and G. Sartoretti, “Imaginav: Scalable embodied navigation via generative visual prediction and inverse dynamics,” *arXiv preprint arXiv:2603.13833*, 2026.
- [7] X. Huang, W. Gai, T. Wu, C. Wang, Q. Zheng, Z. Liu, X. Zhou, Y. Wu, and F. Gao, “Navdreamer: Video models as zero-shot 3d navigators,” *IEEE Robotics and Automation Letters*, 2026.
- [8] X. Liu, J. Huang, S. Xia, B. Liu, J. Cui, and J. Yang, “Imagineuav: Aerial vision-language navigation via world-action modeling and kinodynamic planning,” *arXiv preprint arXiv:2606.01205*, 2026.
- [9] A. Majumdar, A. Sooriyachchi, B. Tibi, C. Bamford, E. Chane-Sane, G. Lample, K. R. Chandu, L. H. Fuh, M. Poiree, O. Duchenne *et al.*, “Robostravel,” *arXiv preprint arXiv:2607.20785*, 2026.
- [10] 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 *2018 IEEE/CVF conference on computer vision and pattern recognition*. IEEE, 2018, pp. 3674–3683.
- [11] N. Yokoyama, R. Ramakrishna, A. Das, D. Batra, and S. Ha, “Hm3d-ovon: A dataset and benchmark for open-vocabulary object goal navigation,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 5543–5550.
- [12] S. Wang, J. Zhang, M. Li, J. Liu, A. Li, K. Wu, F. Zhong, J. Yu, Z. Zhang, and H. Wang, “Trackvla: Embodied visual tracking in the wild,” in *Proceedings of The 9th Conference on Robot Learning*. PMLR, 2025.
- [13] J. Zhang, A. Li, Y. Qi, M. Li, J. Liu, S. Wang, H. Liu, G. Zhou, Y. Wu, X. Li *et al.*, “Embodied navigation foundation model,” in *International Conference on Learning Representations*, vol. 2026, 2026, pp. 127 293–127 322.
- [14] V. Serpiva, J. Sam, C. Simon, H. Amjad, I. Zhura, A. Lykov, and D. Tsetserukou, “Dreamtonav: Generalizable navigation for robots via generative video planning,” *arXiv preprint arXiv:2603.06190*, 2026.
- [15] H. Zhang, S. Liang, L. Chen, Y. Li, Y. Xu, Y. Zhong, F. Zhang, and H. Li, “Sparse video generation propels real-world beyond-the-view vision-language navigation,” *arXiv preprint arXiv:2602.05827*, 2026.
- [16] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang *et al.*, “Wan: Open and advanced large-scale video generative models,” *arXiv preprint arXiv:2503.20314*, 2025.
- [17] B. Chen, D. Martí Monsó, Y. Du, M. Simchowitz, R. Tedrake, and V. Sitzmann, “Diffusion forcing: Next-token prediction meets full-sequence diffusion,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 24 081–24 125, 2024.
- [18] S. L. Li, E. Kim, X. Bai, T. Zhao, T. Pang, M. Simchowitz, and V. Sitzmann, “Turning video models into generalist robot policies,” *arXiv preprint arXiv:2605.27817*, 2026.
- [19] A. W. Harley, Y. You, X. Sun, Y. Zheng, N. Raghuraman, Y. Gu, S. Liang, W.-H. Chu, A. Dave, S. You *et al.*, “Alltracker: Efficient dense point tracking at high resolution,” in *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2025, pp. 5253–5262.
- [20] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” *arXiv preprint arXiv:2106.09685*, 2021.
- [21] M. Wei, C. Wan, X. Yu, T. Wang, Y. Yang, X. Mao, C. Zhu, W. Cai, H. Wang, Y. Chen *et al.*, “Streamvln: Streaming vision-and-language navigation via slowfast context modeling,” *arXiv preprint arXiv:2507.05240*, 2025.
- [22] V. Leroy, Y. Cabon, and J. Revaud, “Grounding image matching in 3d with mast3r,” in *European conference on computer vision*. Springer, 2024, pp. 71–91.
- [23] C. Beattie, J. Z. Leibo, D. Teplyaev, T. Ward, M. Wainwright, H. Küttler, A. Lefrancq, S. Green, V. Valdés, A. Sadik *et al.*, “Deepmind lab,” *arXiv preprint arXiv:1612.03801*, 2016.