D-CLING: Prior-Preserving Depth-Conditioned Fine-Tuning for Navigation Foundation Models

Shintaro Nakaoka, Takayuki Kanai, and Kazuhito Tanaka

***Abstract*— Navigation Foundation Models (NFMs) trained on large, cross-embodied datasets have demonstrated powerful generalizability on various scenarios. Adopting in-domain fine-tuning upon an NFM efficiently calibrates the visuomotor policy, promising further improvement even in a novel scenario. However, the fine-tuned models still suffer from poor obstacle avoidance or fail to properly reach the provided goals. Furthermore, such model updates in a small subset of data typically erode the pretrained prior, compromising the pretraining generalization. Consequently, fine-tuning rather deteriorates the model's capability of robust and accurate navigation. In this work, we present a novel fine-tuning method that leverages the large-scale pretraining while efficiently learning novel setups, such as the environment or camera configuration. In particular, inspired by ControlNet, we fine-tune an NFM by attaching a trainable copy of the pretrained backbone using zero-initialized residual pathways, thereby learning geometric cues. This design enables the model to efficiently acquire the in-domain geometry while preserving the pretrained knowledge for various behaviors. Despite the simplicity, our comprehensive evaluation of real-world navigation suggests that our proposal effectively enables robust long-horizon navigation with minimum collisions or human intervention.**

I. Introduction

Visual navigation based on a sequence of images has emerged as a fundamental paradigm in mobile robotics research, encompassing diverse task formulations such as Image-Goal navigation and Vision-and-Language Navigation [1, 2, 3, 4]. Recently, imitation learning approaches that learn optimal navigation policies from expert-demonstrated trajectories have gained significant attention in robotics navigation [5, 6]. In particular, Navigation Foundation Models (NFMs), such as ViNT [7] and NoMaD [8], which learn goal-conditioned policies from large-scale data and transfer the pretrained knowledge across environments and robot embodiments, have achieved reliable goal-reaching and obstacle avoidance. Importantly, the scale of the pretraining dataset is critical to acquiring the diversity of navigation behaviors, including obstacle avoidance and alternative routes at junctions. Therefore, NFMs typically employ image-to-policy learning without additional sensory modalities (RGB-only learning) to enable massive, low-cost, and standardized pretraining.

However, NFMs still struggle with adaptation to a novel scene and robotic configuration, even in a *seemingly* close to those seen during pretraining. We attribute this primarily to *domain-shift* in geometric perception, stemming from camera configuration, such as the field of view, distortion, and so on. Indeed, we observed that input images that are more *undistorted* than those used for pretraining the backbone, led to hitting obstacles or failing to reach the proper goal point (Fig. 1). Generally, a well-known approach to address such failures is full fine-tuning, i.e., calibrating the entire policy backbone to new domains [7, 9].

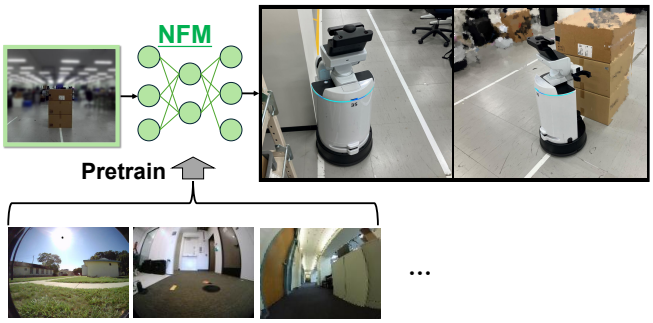

Fig. 1. Failure scenes of zero-shot NoMaD [8] in real-robot navigation, such as an unsafe clearance (left) and distance misestimation (right). In particular, the pretraining images exhibit *out-of-distribution* distortion relative to our experimental condition, impairing geometric perception.

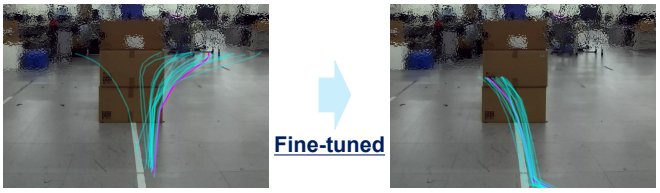

Fig. 2. **Less divergent trajectory generation after fine-tuning.** Based on a shared start–goal and an observation point, we visualize $N = 20$ trajectories sampled from each model: **Left**: the zero-shot *NoMaD* (before being fine-tuned); **Right**: the fine-tuned model. Fine-tuning yields markedly lower diversity, with narrower spatial spread and reduced heading variance, indicating a collapse of pretrained priors.

Yet, we also observed that the problem persists even after fine-tuning. Especially, the fine-tuned model is prone to generating less diverse trajectories, suggesting *catastrophically-forgetting* of the pretrained knowledge (Fig. 2).

In short, techniques for efficiently using large-scale pretraining suffer from two key difficulties: (i) a lack of accurate geometry awareness given a novel scene/embodiment, and (ii) insufficient behavioral diversity that is crucial to dealing with the various navigational scenarios.

In this work, we present a novel fine-tuning for NFMs, **D-CLING**[1] (***D**epth-conditioned, **C**ontrolNet-driven **L**earn**I**ng for General **N**avi**G**ation Models*), that leverages large-scale pretraining while efficiently adapting to novel environments

Authors are with Frontier Research Center, Toyota Motor Corporation, Toyota, Aichi, Japan. {first_lastname}@mail.toyota.co.jp

[1]We named D-CLING, hoping the model ***cling*** the pretrained knowledge of NFMs without *catastrophically forgetting* it in the ***d**epth*-guided tuning.

and geometric perception (Fig. 3). The core idea is fine-tuning a policy backbone with *dense depth* conditioning to capture accurate scene geometry, using ControlNet-style residual pathway learning [10]. The pathway injects dense depth signals into the model's intermediate layers, progressively updating the parameters with geometry-awareness. Hence, the fine-tuned model is expected to preserve the original navigation capability while smoothly increasing its in-domain geometry-awareness, resulting in robust and accurate navigation at the end of the fine-tuning.

We evaluate D-CLING by building upon NoMaD [8], a standard NFM pretrained on diverse domains. Our real-world evaluation demonstrates that D-CLING offers a substantial improvement in goal-reachability and collision avoidance skills relative to the baselines: zero-shot NoMaD, the RGB fine-tuned model following the typical protocol, and the RGB-D fine-tuned model via an *early-fusion* strategy [11].

In summary, the main contributions of this work are as follows:

- **Prior-preserving fine-tuning framework:** We introduce a depth-conditioned adaptation that retains pretrained policy priors while explicitly injecting geometry-awareness.
- **Comprehensive validation:** We present comprehensive experiments, showing that D-CLING achieves superior goal reachability and obstacle avoidance in real-robot deployments compared with typical baselines.

II. Methodology

A. Overview of Proposed Method Behavior

Figure 3 presents our proposed framework adopted to a representative NFM, NoMaD [8]. As in the original NoMaD, the pretrained weight-frozen NoMaD backbone (*RGB Branch*) maps a short RGB history and a goal image to actions. In parallel, a depth-conditioned branch (*Depth Branch*) encodes the same RGB inputs together with a depth map to produce conditioning features. Then, the model outputs short-horizon waypoints.

B. Model Architecture

We freeze all layers of the pretrained NoMaD (*RGB Branch*) and create a copy of them to form the *Depth Branch*. *Depth Branch* receives an RGB-D frame $\tilde{\mathbf{o}}_t \in \mathbb{R}^{h\times w\times 4}$ and begins with a $4 \to 3$ embedding layer that projects the four-channel input to three channels. All subsequent modules follow NoMaD. Conditioned on the context vector c'_t, a U-Net based diffusion model of *Depth Branch* produces intermediate features. At every corresponding U-Net layer, depth intermediate features are added to the *RGB Branch*.

Following ControlNet, we insert zero-initialized 1×1 convolutions immediately before the U-Net and immediately after each U-Net layer on the *Depth Branch*. Let $F_\ell(\cdot;\Theta_\ell)$ denote the intermediate block at stage $\ell \in \{1,\dots,L\}$ of the U-Net–based diffusion model of the *RGB Branch*, with input feature h_ℓ and output $y_\ell = F_\ell(h_\ell;\Theta_\ell)$. Here, Θ_ℓ denotes the model parameters of F_ℓ. In the *RGB Branch*, the parameters Θ_ℓ of F_ℓ are frozen.

For the *Depth Branch*, we introduce a counterpart $F^d_\ell(\cdot;\Theta^d_\ell)$ and a single zero-initialized 1×1 convolution $Z_\ell(\cdot;\Theta^z_\ell)$. With the depth-derived feature h^d_ℓ, let $u^d_\ell = F^d_\ell(h^d_\ell;\Theta^d_\ell)$ be the intermediate feature of the U-Net based diffusion model at stage ℓ. We form the block output as the element-wise sum of y_ℓ and the zero-initialized 1×1 convolution applied to u^d_ℓ:

$$y'_\ell = y_\ell + Z_\ell\left(u^d_\ell\,;\,\Theta^z_\ell\right). \tag{1}$$

Importantly, the 1×1 fusion gate is zero-initialized as:

$$\Theta^z_\ell = \mathbf{0} \implies \forall x:\ Z_\ell(x;\Theta^z_\ell) = \mathbf{0} \tag{2}$$

Hence, at the initialization phase, all the layer-wise outputs y_ℓ behave as their original form, s.t., $y'_\ell = y_\ell = F_\ell(h_\ell;\Theta_\ell)$. Thus, gradients update *Depth Branch* parameters gradually via the zero-initialized fusion, while the RGB trunk remains frozen.

Note that our approach is a relatively simple adoption of the ControlNet philosophy to validate the proposal's impact. Although further extensions, e.g., a repulsive safety head from monocular depth [12], externally providing a 3D map [13], can be integrated for future extensions, we intentionally exclude them from this paper.

III. Experiments

A. Fine-tuning Setups

Dataset construction. We collected synchronized RGB-odometry sequences using a Toyota Human Support Robot (HSR) [14] equipped with a ZED 2. For dense depth estimation, we used a learning-based stereo-to-depth estimator pretrained on in-house datasets [15]. The sequences are collected in a large-scale office room. The space combines standard office furniture with specialized robotics equipment and experimental setups, resulting in a heterogeneous environment that challenges navigation with both typical and atypical obstacles. This dedicated fine-tuning dataset, RealHSRNav, is collected in roughly three hours for demonstration data. Note that RealHSRNav is used as the *sole* dataset for all fine-tuning experiments reported in this paper.

Importantly, our collected dataset can pose the model a domain-shift due to differences in the camera field of view — though the model was originally pretrained on *fisheye*-like images mostly, the equipped camera provides a *pinhole*-like projection (approximately 110° horizontal). Thus, adequately *calibrating* scene perception is needed to leverage the pretrained knowledge.

Implementation details. To implement D-CLING, we fine-tuned an off-the-shelf checkpoint of the NoMaD[2] for 30 epochs on a single NVIDIA RTX 4090 GPU with a batch size of 256 and a learning rate of 2.5×10^{-5}. Following the original study [8], we train with AdamW [16] using a cosine learning-rate schedule with warm-up, optimizing the unmodified NoMaD loss.

[2] `https://github.com/robodhruv/visualnav-transformer` (retrieved 10 July 2025)

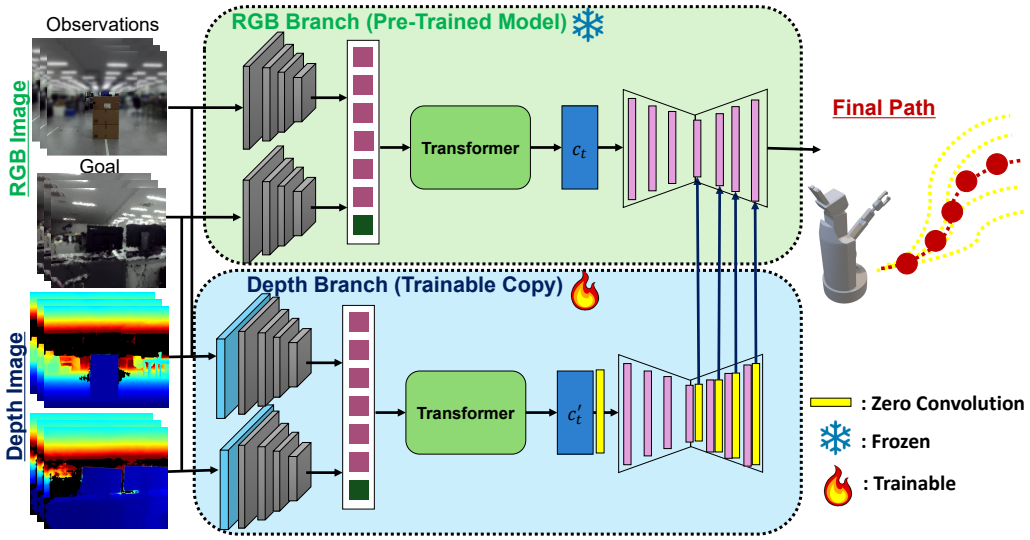

Fig. 3. Architecture overview. A frozen pretrained *RGB Branch* (identical to the NoMaD architecture) maps an RGB history and a goal image to intermediate features. In parallel, a trainable *Depth Branch* ingests RGB–D (with a $4 \to 3$ embedding) and injects zero-initialized residual features into a U-Net based diffusion model; the two streams are fused by element-wise addition at each U-Net stage during training and inference. The diffusion head outputs a short-horizon action distribution, enabling prior-preserving depth conditioning.

B. Baselines

We used the following ablative models to compare with the proposed D-CLING:

NoMaD (Zero-Shot). We evaluate the same NoMaD checkpoint for the fine-tuning variants, including our proposal. We show that this zero-shot adoption frequently achieves a *runner-up* position in various tasks, supported by the knowledge obtained in large-scale pretraining.

NoMaD-FT (Full Fine-Tuning). All NoMaD parameters are fine-tuned on our dataset under identical conditions to D-CLING. This baseline shows the difficulty of in-domain learning, i.e., that naive fine-tuning of the zero-shot model is insufficient or rather degrades its original ability in various cases.

NoMaD-EF (Early Fusion). Based on the NoMaD checkpoint, we added a depth encoder with the same architecture as the RGB encoder, which is trainable and randomly initialized. This is a common practice to multimodalize model input [11]: it adds a trainable depth-only backbone parallel to the RGB backbone and fuses tokens via channel-wise concatenation followed by a 1×1 projection without additional residual paths.

C. Real-world Experiments

Scenarios. We evaluate all the methods in the following three scenarios, which align with real-world scenarios of navigation tasks (Fig. 4):

(i) **Basic obstacle avoidance (*Basic Obstacle*)**. The robot traverses a corridor while avoiding a stationary box; no additional obstacles are introduced during the evaluation.

(ii) **Dynamic corridor (*Dynamic Corridor*)** with a map-absent chair. After traversing approximately $10\,\mathrm{m}$ in a dynamical environment, the robot encounters a chair placed at the corridor center that is not represented in the pre-collected goal images, and thus must be avoided only by visual observation. This scenario reflects everyday human-space disturbances such as moved furniture, crossing pedestrians, and people stepping away.

(iii) **Long-range navigation (*Long-range*)**. The robot follows an approximately $50\,\mathrm{m}$ semicircular trajectory through the office, crosses two junctions, and deals with various scene dynamics. The environment contains changes not present in the pre-collected goal images, which evaluates robustness to appearance shifts and long-horizon navigation.

Experimental details. We conducted the experiments in our office environment on the Toyota HSR, the same platform used for the dataset collection. Linear and angular speeds are limited to $0.45\,\mathrm{m/s}$ and $1.0\,\mathrm{rad/s}$. The policy consumes two sources of context: (1) a short visual history of $T+1$ frames (the current RGB frame and its T immediate predecessors), each paired with a per-frame depth estimate; and (2) a topological map encoded as an ordered sequence of goal images captured at uniform spatial intervals during the initial setup of the environment. The model outputs $H+1$ waypoints including the current step. We set $T = 3$ and $H = 7$.

Metrics. For scenarios (i) and (ii), we run 10 trials each and report the success rate (SR). A trial is considered successful if the robot reaches the goal within the allotted time without collisions or human intervention. For scenario (iii), we run 5 trials, and record the number of detected *safety triggers* for operator interventions. Note that the trigger is used in an off-the-shelf manner, implemented on Toyota HSR [14]. We report the mean interventions per trial (lower is better), together with the 95% confidence intervals.

Results. Table I reports real-robot performance across three scenarios. Our proposal, D-CLING, consistently outperforms the baselines. It achieves the highest success rates in (i) and (ii), and requires far fewer interventions in (iii). We attribute these gains to the geometry-awareness provided by dense depth adoption while preserving the diverse action patterns

TABLE I

REAL-WORLD NAVIGATION PERFORMANCE ACROSS THREE SCENARIOS. THE AVERAGE SUCCESS RATE (SR) IN 10 TRIALS EACH FOR (I) AND (II), AND THE AVERAGE HUMAN INTERVENTIONS (INTERVENTIONS) OF 5 TRIALS FOR (III) ARE LISTED.

Method	Training	Modality	(i) *Basic Obstacle*	(ii) *Dynamic Corridor*	(iii) *Long-range*
			SR (%) ↑	SR (%) ↑	Interventions ↓
NoMaD [8]	Frozen	RGB	<u>50</u>	0	<u>2.6</u>
NoMaD-FT	Full fine-tune	RGB	30	<u>10</u>	3.2
NoMaD-EF	Early fusion	RGB-D	40	0	4.4
D-CLING (Ours)	**Zero-init**	**RGB-D**	**70**	**60**	**1.2**

(i) *Basic Obstacle* (ii) *Dynamic Corridor* (iii) *Long-range*

Fig. 4. Representative frames of **our proposed method** from real-world experiments with a robot overlaid in an office environment: (i) *Basic Obstacle*—corridor traversal with visual avoidance of a single stationary box; (ii) *Dynamic Corridor*—after 10 m the robot must avoid an unmapped chair; and (iii) *Long-range*—a 50 m semicircular route across two junctions.

of the model, which in turn improves obstacle avoidance and long-horizon goal reachability (Fig. 4).

In contrast, zero-shot NoMaD remains problematic, particularly on (ii), even though the model was originally pretrained on similar *indoor* datasets [17, 18]. We conjecture that this is owing to domain shift stemming from the camera geometry and/or scene appearance. Furthermore, NoMaD-FT and NoMaD-EF underperform zero-shot NoMaD in (i) and (iii), though in-domain training is applied. We anticipate that input sequences from novel scenes or a novel modality, i.e., dense depth, have eroded pretrained knowledge. In fact, in NoMaD-EF, where the learning for a novel domain is forcibly applied to the RGB-prelearned policy (i.e., off-the-shelf NoMaD), intervention is needed the most frequently to execute the long-horizon task (iii). In other words, interventions are more frequent than in the policy that leverages only the RGB modality as originally trained, i.e., NoMaD-FT. Thus, it can be interpreted that the policy that is *the most severely* affected by catastrophic-forgetting —depth learning that catastrophically overrides the RGB learned backbone— provides the worst score on the task.

IV. CONCLUSION

Zero-shot adoption of NFMs still suffers from novel scene complexity, camera parameters, etc. Nevertheless, fine-tuning on limited in-domain dataset is still insufficient to adapt them. Furthermore, typical fine-tuning hinders diverse action generation of pretrained behavior, which is crucial for various real-world navigation tasks. We presented D-CLING, a prior-preserving and depth-conditioned NFM fine-tuning strategy that leverages a frozen RGB branch and enables learning of dense depth guidance upon it through the zero-initialized layer. Real-world experiments compared with ablative strategies exhibited the efficacy of our proposal to achieve robust navigation, as well as awareness of metrically accurate action prediction.

In future work, we will systematically compare conditioning modalities of ControlNet-based fine-tuning to clarify each contribution. Specifically, RGB-only vs. RGB plus depth will quantify the gains attributable to depth versus on-domain photometric adaptation, while dense vs. sparse and metric vs. up-to-scale relative depth will assess the benefits and trade-offs of the modality. A promising direction to further improve the navigation capability is to maintain off-screen awareness through multi-frame temporal modeling and auxiliary sensing to reduce collisions. This targets our most frequent failure case, where once an obstacle leaves the camera's view, the robot *forgets* the obstacle and returns to the original path, which at last collides with the previously avoided obstacle.

REFERENCES

[1] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al., "End to end learning for self-driving cars," *arXiv*, 2016.

[2] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva, et al., "On evaluation of embodied navigation agents," *arXiv*, 2018.

[3] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning." In *Proc. of the ICRA*, 2017, pp. 3357–3364.

[4] E. Wijmans, A. Kadian, A. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra, "Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames," *arXiv*, 2019.

[5] L. Tai, J. Zhang, M. Liu, and W. Burgard, "Socially compliant navigation through raw depth inputs with generative adversarial imitation learning." In *Proc. of the ICRA*, 2018, pp. 1111–1117.

[6] R. Ramrakhya, E. Undersander, D. Batra, and A. Das, "Habitat-web: Learning embodied object-search strategies from human demonstrations at scale." In *Proc. of the CVPR*, 2022, pp. 5173–5183.

[7] D. Shah, A. Sridhar, N. Dashora, K. Stachowicz, K. Black, N. Hirose, and S. Levine, "Vint: A foundation model for visual navigation," *arXiv*, 2023.

[8] A. Sridhar, D. Shah, C. Glossop, and S. Levine, "Nomad: Goal masked diffusion policies for navigation and exploration." In *Proc. of the ICRA*, 2024, pp. 63–70.

[9] J. Wan, C. Zhou, J. Liu, X. Huang, X. Chen, X. Yi, Q. Yang, B. Zhu, X.-Q. Cai, L. Liu, et al., "Pig-nav: Key insights for pretrained image goal navigation models," *arXiv*, 2025.

[10] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models." In *Proc. of the CVPR*, 2023, pp. 3836–3847.

[11] S. Gode, A. Nayak, D. N. Oliveira, M. Krawez, C. Schmid, and W. Burgard, "Flownav: Combining flow matching and depth priors for efficient navigation," *arXiv*, 2024.

[12] J. Kim, J. Sim, W. Kim, K. Sycara, and C. Nam, "Enhancing safety of foundation models for visual navigation through collision avoidance via repulsive estimation," *arXiv*, 2025.

[13] K. Honda, T. Ishita, Y. Yoshimura, and R. Yonetani, "Gsplatvnm: Point-of-view synthesis for visual navigation models using gaussian splatting," *arXiv*, 2025.

[14] T. Yamamoto, K. Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase, "Development of Human Support Robot as the research platform of a domestic mobile manipulator," *ROBOMECH J.*, vol. 6, no. 1, p. 4, 2019.

[15] K. Shankar, M. Tjersland, J. Ma, K. Stone, and M. Bajracharya, "A learned stereo depth system for robotic manipulation in homes," *IEEE RA-L*, vol. 7, no. 2, pp. 2305–2312, 2022.

[16] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv*, 2017.

[17] N. Hirose, F. Xia, R. Martín-Martín, A. Sadeghian, and S. Savarese, "Deep visual mpc-policy learning for navigation," *IEEE RA-L*, vol. 4, no. 4, pp. 3184–3191, 2019.

[18] N. Hirose, D. Shah, A. Sridhar, and S. Levine, "Sacson: Scalable autonomous control for social navigation," *IEEE RA-L*, vol. 9, no. 1, pp. 49–56, 2023.