2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

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 large-scale pretraining while efficiently learning in novel setups, such as environments or camera configurations. 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 in-domain geometry while preserving pretrained knowledge across various behaviors. Despite its simplicity, our comprehensive evaluation of real-world navigation suggests that our proposal effectively enables robust long-horizon navigation with minimal collisions and human intervention. Additionally, our offline analysis shows that the proposed strategy maintains or further improves action prediction capability beyond the fine-tuned dataset, providing a key insight into continual learning for general navigation.**
The project page: `https://toyotafrc.github.io/DCLING-Proj/`

***Index Terms*— Vision-Based Navigation, Collision Avoidance, Catastrophic Forgetting, Depth Estimation**

I. Introduction

Visual navigation based on a sequence of images has emerged as a fundamental paradigm in mobile robotics research, encompassing diverse tasks [1, 2]. One of the key enablers of them is *imitation learning*, where optimal navigation policies are learned through the expert demonstrations in an end-to-end manner [3, 4]. In particular, Navigation Foundation Models (NFMs), such as ViNT [5] and NoMaD [6], which employ imitation learning based on the large-scale demonstration across various environments and robots, have achieved reliable goal-reaching and obstacle avoidance. Importantly, the scale and diversity of the training dataset is critical for acquiring a diverse set of such reliable behaviors [7]. Thus, NFMs typically employ an image-to-action policy learning without additional sensory modalities to enable massive, low-cost, and standardized pretraining.

However, NFMs still struggle to adapt to a novel scene and robotic configuration, even when the setups are *seemingly* close to those seen during pretraining. We attribute this primarily to *domain-shift* in geometric perception, stemming from differences in camera configurations and/or scene geometry. Indeed, we observed that providing *more undistorted* images to the NFM than those used for pretraining, led to failure more frequently (Fig. 1). To address such failures, a well-known approach is full fine-tuning of the model, i.e., calibrating the entire policy backbone to new domains [5, 8]. 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: (1) a lack of accurate geometry awareness, given a novel scene and/or embodiment, and (2) insufficient behavioral diversity, which 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*

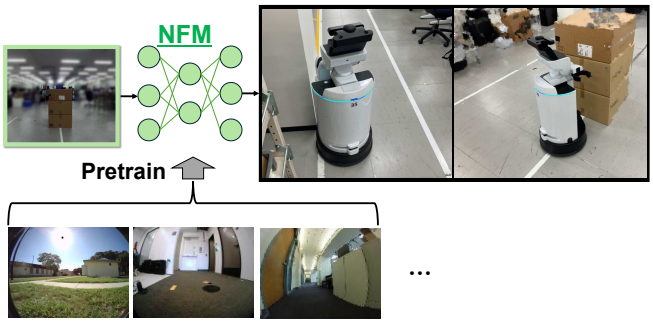


Fig. 1. **Failure scenes of zero-shot NoMaD [6].** In real-robot navigation, we observe failures such as unsafe clearance (left) and distance misestimation (right). In particular, the pretraining images [9, 10, 11, 12] exhibit strong distortion relative to our experimental condition, impairing geometric perception.

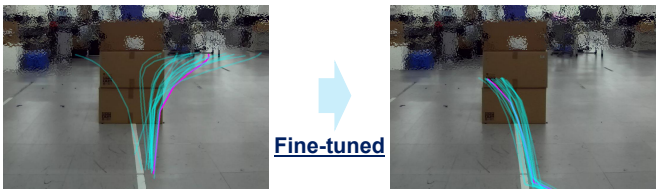


Fig. 2. **Less divergent trajectory generation after fine-tuning.** Based on a shared start–goal and an observation point, we visualize $N = 20$ trajectories sampled from each model: **Left**: the zero-shot *NoMaD* (before being fine-tuned); **Right**: the fine-tuned model. Fine-tuning yields markedly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

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.

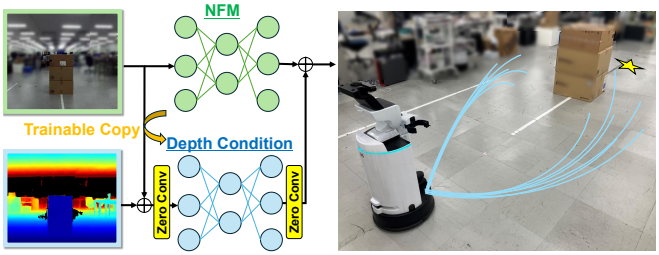


Fig. 3. **Overview of our proposal**: prior-preserving dense-depth conditioning, inspired by ControlNet [13]. Zero-initialized residual pathways inject per-pixel depth into intermediate features, preserving the pretrained prior and improving geometry-consistent obstacle avoidance across cameras.

***L**earn**I**ng for General **N**avi**G**ation Models*), that leverages large-scale pretraining while efficiently adapting to novel environments 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 [13]. 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 built upon NoMaD [6], 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 models, and the RGB-D fine-tuned models in a common (*early-fusion*) strategy [14]. Additionally, our offline analysis of both the fine-tuned and its pretrained domain shows that D-CLING not only adapts to the former *but is also* still powerful in the latter, indicating preservation of the pretrained prior and further extending it.

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

- **Prior-preserving framework of navigation model's fine-tuning:** We introduce a depth-conditioned adaptation that retains pretrained policy priors while explicitly injecting geometry-awareness.
- **Comprehensive validation:** We present comprehensive experiments, showing that the scheme of D-CLING achieves superior goal reachability and obstacle avoidance in both real-robot deployments and offline evaluations compared with typical baselines.
- **Experimental evidence for future extension:** We exhibit that our proposed fine-tuning further extends navigation performance beyond the fine-tuned domain, suggesting the potential for future extensions of NFMs.

II. Related Work

A. Visual Navigation

Traditional visual navigation pipelines were often modular [15], consisting of (1) mapping and localization [16], (2) visual perception, such as terrain traversability or object-category estimation [17], and (3) path planning and low-level control. While the modular strategy performs well in relatively simple environments [18], the hand-crafted interfaces between modules may discard critical cues, thereby reducing the robustness of the overall system.

In contrast, recent end-to-end strategies, such as reinforcement learning [19] and imitation learning [6], challenge its difficulty by acquiring critical queues from data with minimal assumptions. In particular, imitation learning is gaining significant traction because it avoids the laborious reward engineering [20] and offers higher sample efficiency [1].

In addition, it can train the navigation policy directly using real-world teleoperation logs, without relying on long-horizon exploration. Accordingly, imitation-learning–based navigation has been widely applied across diverse navigation tasks, including PointGoal [21, 22], ImageGoal [8, 23, 24], Object Navigation, Vision-and-Language Navigation [25, 26, 27], and Social Navigation [9, 28], among others.

Despite its practical advantages, imitation-learning-based navigation remains vulnerable to domain shift in novel domains, e.g., changes in layout/illumination [29] or camera extrinsics/intrinsics [30], which can sharply reduce the task success. To overcome these limitations, recent work has introduced NFMs [6, 22, 31, 32], end-to-end-trained models on large, heterogeneous datasets spanning multiple robots and environments. Relying only on RGB lowers hardware and data-collection costs, avoids multi-sensor calibration and synchronization, and keeps datasets comparable across platforms. Our work is built on such NFMs, and we have further extended it without additional large-scale training, yet with small-scale training.

B. Fine-tuning for Navigation Foundation Model

Fine-tuning is one of the most common practices for further boosting the capabilities of zero-shot models, as domain-specific knowledge is still a key source of task performance [5]. This strategy offers strong sample efficiency, enabling rapid adaptation from few demonstrations [33]. To fine-tune the visuomotor policy, a standard strategy prioritizes downsizing and maintaining a *same I/O* —since the model typically possesses (1) large-scale parameters to handle a large amount of data and (2) can suffer from catastrophic forgetting of the pretrained knowledge, such strategies are promising for feasible yet efficient real-robot adaptation. Indeed, based on the powerful zero-shot models [34, 35, 36], parameter-efficient adaptation (PEFT), such as LoRA and adding adapter layers, has been shown to preserve pretrained priors while enabling lightweight adaptation in manipulation settings, mitigating the priors' forgetting and overfitting [37, 38, 39, 40].

In contrast to that philosophy, fine-tuning NFMs need not always prioritize downsizing, as their parameter scale is typically negligible. This is because the navigation task must be *real-time*, and thus, the models are designed in a lightweight manner even for NFMs [5, 6]. This nature of NFM unlocks addressing the off-the-shelf use of the

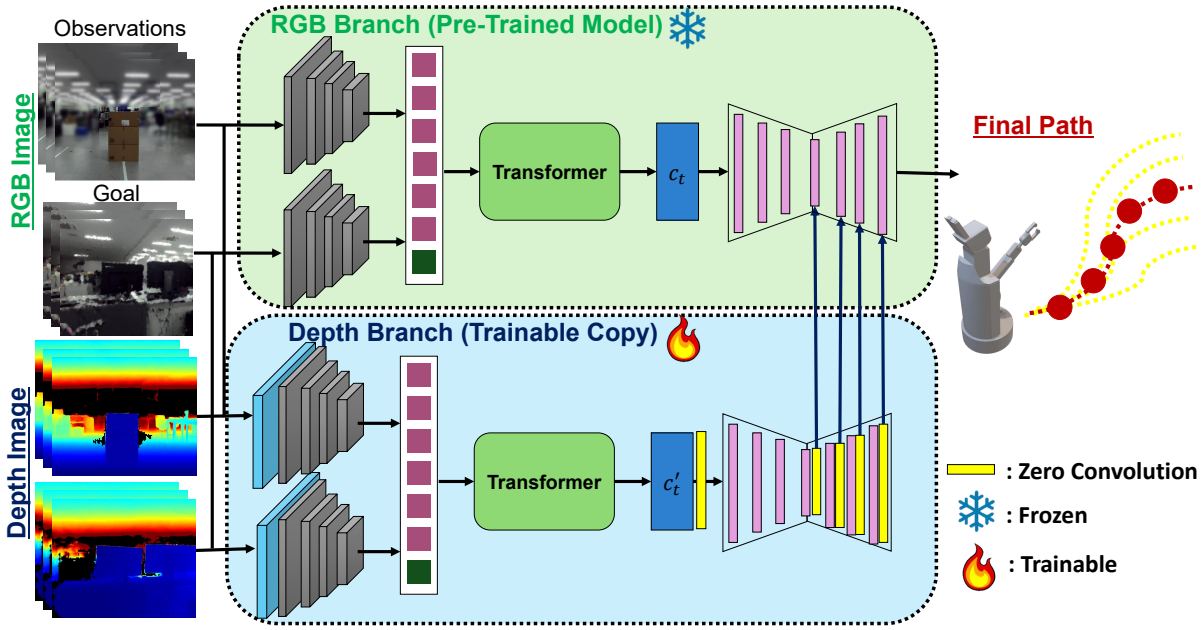


Fig. 4. **D-CLING architectural design, built upon NoMaD [6].** A frozen pretrained *RGB Branch* (identical to the NoMaD architecture) maps an RGB history and a goal image to intermediate features. In parallel, a trainable *Depth Branch* ingests RGB–D (with a $4 \to 3$ embedding) and injects zero-initialized residual features into the 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.

large-scale backbone, and/or even running in parallel for fine-tuning and inference. We thus employ ControlNet-like fine-tuning as a key strategy [13], which rather prioritizes *preserving* the diversity of data generation patterns over the model's memory footprint and its inference speed. Moreover, the ControlNet-based update of the backbone is localized rather than updating the entire backbone, which is less prone to overfitting, even with a small-scale dataset [37, 38, 39]. Therefore, policy tuning can be achieved without a costly dataset while maintaining pretrained knowledge.

Among recent works, LiReN is the most relevant to our study, where a combination of offline RL pretraining and autonomous online RL fine-tuning for NFMs is introduced [19]. Without relying on real-world exploration or an RGB-D pretrained NFM backbone, we show that the ControlNet-inspired strategy, D-CLING, efficiently robustified the NFM policy under human-controlled demonstrations. Moreover, we experimentally quantify that our proposal is more suitable for preserving the pretrained knowledge than the naive RGB-D fusion strategy (i.e., the *early-fusion*) [14], indicating the insight for the future fine-tuning design beyond this work.

III. Problem Formulation

The objective of this study is to learn a policy π_θ that,

$$\pi_\theta : \left(\tilde{\mathbf{o}}_{t-T:t},\, \mathbf{g}\right) \longmapsto p_\theta(\mathbf{a}_{t:t+H} \,|\, \tilde{\mathbf{o}}_{t-T:t},\, \mathbf{g}), \qquad (1)$$

where a sequence of observations $\tilde{\mathbf{o}}_{t-T:t} = (\tilde{\mathbf{o}}_{t-T}, \ldots, \tilde{\mathbf{o}}_t)$ and a goal image $\mathbf{g}$ provides a probability distribution $p_\theta(\cdot)$ over future action sequences $\mathbf{a}_{t:t+H} = (\mathbf{a}_t, \ldots, \mathbf{a}_{t+H})$, thereby enabling a mobile robot to reach its goal using only visual observations.

Here, $\tilde{\mathbf{o}}_t \in \mathbb{R}^{h\times w\times 4}$ denotes a single RGB-D observation obtained by concatenation of the raw RGB image with the depth map, assuming that a learning-based depth-estimation network provides it. The goal information is provided as a reference RGB image $\mathbf{g} \in \mathbb{R}^{h\times w\times 3}$ depicting the target viewpoint. h and w denote the image height and width, respectively, and $\mathbf{a}_t \in \mathbb{R}^2$ is a 2-D action vector that represents a waypoint in the current robot frame at time t. The hyper-parameter H denotes the prediction horizon, while T is the number of past frames used as input.

IV. Methodology

A. Preliminaries: NoMaD [6]

NoMaD [6] is illustrated in the upper half of Fig. 4. The model receives the current RGB frame together with T-frame RGB history $\mathbf{o}_t = o_{t-T:t}$, and predicts a distribution over the action sequence from the current step through H future steps $\mathbf{a}_t = a_{t:t+H}$. Visual features are extracted with the Visual Navigation Transformer (ViNT) [5]. The current frame o_t and the goal image o_g are embedded by an encoder $\phi(\cdot)$, whereas the history window $o_{t-T:t}$ is embedded by a separate encoder $\psi(\cdot)$. The resulting vectors are fed into a Transformer block $f_{\mathrm{tr}}(\cdot)$ together, yielding the context vector $\mathbf{c}_t$ as:

$$\mathbf{c}_t = f_{\mathrm{tr}}\big(\psi(o_{t-T:t}),\, \phi(o_t, o_g)\big). \qquad (2)$$

Conditioned on $\mathbf{c}_t$, a diffusion model with parameters θ defines the conditional distribution $p_\theta(\mathbf{a}_{t:t+H} \mid \mathbf{c}_t)$, from which a sample is drawn:

$$\hat{\mathbf{a}}_{t:t+H} \sim p_\theta(\cdot \mid \mathbf{c}_t). \qquad (3)$$

B. Overview of Proposed Method

Figure 4 presents our proposed framework adopted to a representative NFM, NoMaD [6]. 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.

C. 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:

$$y'_\ell = y_\ell + Z_\ell\left(u^d_\ell\,;\,\Theta^z_\ell\right). \tag{4}$$

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

$$\Theta^z_\ell = \mathbf{0} \implies \forall x:\ Z_\ell(x;\Theta^z_\ell) = \mathbf{0} \tag{5}$$

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 [41], externally providing a 3D map [42], can be integrated for future extensions, we intentionally exclude them from this paper.

V. Experiments

A. Fine-tuning Setups

Dataset construction. We collected synchronized RGB-odometry sequences using a Toyota Human Support Robot (HSR) [43] equipped with a ZED 2. For dense depth estimation, we used a learning-based stereo-to-depth estimator pretrained on in-house datasets [44]. 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. The dataset also contains scenarios requiring avoidance of dynamic obstacles, in addition to static obstacles. This dedicated fine-tuning dataset, RealHSRNav, is achieved 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.

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 [6], we train with AdamW [45] using a cosine learning-rate schedule with warm-up, optimizing the parameters with respect to the unmodified NoMaD loss function.

Importantly, our choice of the checkpoint and RealHSRNav dataset poses a domain-shift, particularly 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.

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: naive fine-tuning of the zero-shot model is insufficient, and, in fact, degrades its original ability.

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 [14]: 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

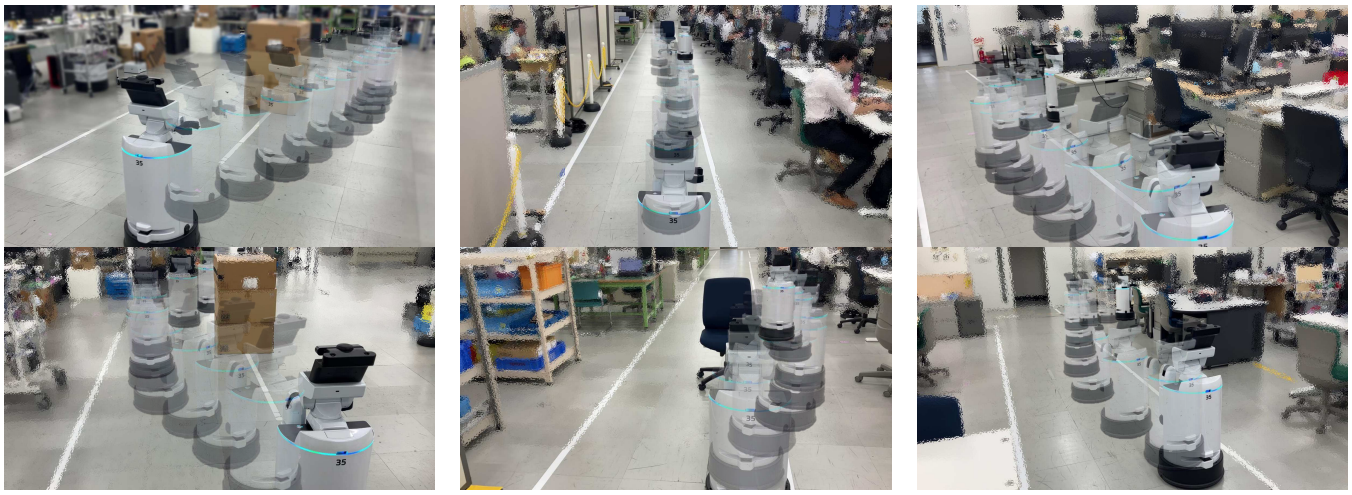
Scenarios. We evaluate all the methods in the following three scenarios, which align with real-world scenarios of navigation tasks (Fig. 5):

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

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 [6]	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. 5. **Representative frames of our proposed method from real-world experiments with a physical overlaid robot 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.

(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 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 m trajectory that covers about half a circuit of 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 to collect the fine-tuning dataset. Its linear and angular speeds are limited to 0.45 m/s and 1.0 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 $H = 7$, following NoMaD [6].

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 without collisions or human intervention. For scenario (iii), we ran 5 trials, and recorded the number of detected *safety triggers* from operator interventions. We report the mean interventions per trial (lower is better), along 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 of the model, which in turn improves obstacle avoidance and long-horizon goal reachability (Fig. 5).

TABLE II

BENCHMARK ON OFFLINE DATA. ADE / FDE / DTW PROVIDES AN ERROR METRIC, WHERE LOWER VALUES ARE BETTER.

Method	*F.T. Dataset*			*NoMaD Dataset*											
	RealHSRNav			Recon [10]			GoStanford [11]			Sacson [9]			Scand [12]		
	ADE	FDE	DTW	ADE	FDE	DTW	ADE	FDE	DTW	ADE	FDE	DTW	ADE	FDE	DTW
NoMaD	1.326	2.160	0.917	1.691	**2.996**	**1.301**	2.267	4.448	1.888	2.508	**4.285**	2.003	2.035	2.990	1.283
NoMaD-FT	2.138	5.484	2.255	1.810	4.775	1.956	2.097	5.436	2.216	**1.861**	4.435	**1.913**	**1.622**	4.115	1.659
NoMaD-EF	1.897	4.244	1.674	2.222	4.676	1.991	2.540	6.246	2.592	2.737	5.497	2.455	2.428	5.063	2.045
D-CLING (Ours)	**1.298**	**1.443**	**0.726**	**1.502**	3.037	1.312	**1.812**	**4.275**	**1.739**	2.521	4.385	1.929	1.839	**2.401**	**1.065**

In contrast, zero-shot NoMaD remains problematic, particularly on (ii), even though the model was originally pretrained on similar *indoor* datasets [9, 11]. 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 they are trained on in-domain data. In particular, in NoMaD-EF, where the learning for a novel domain is forcibly applied to the RGB-pretrained policy (i.e., off-the-shelf NoMaD), intervention is required the most frequently to execute the task (iii). We anticipate that input sequences from a novel modality, i.e., dense depth, have eroded pretrained knowledge.

Limitations. We observed the following failure case in our method: even if the robot successfully avoids the collision *initially*, it immediately returns to the original path and hits the obstacles. We hypothesize that this failure stems from the limited awareness of the temporal context. As our NoMaD-based policy conditions on only four (current plus three past) frames without persistent memory mechanisms, obstacles that move off-screen can no longer be tracked by the policy. The effect is becoming more pronounced in our HSR setup, where a narrower field of view camera is employed than the fisheye [6], further reducing the visible workspace. Maintaining awareness of the off-screen state via multiframe fusion or auxiliary sensing might be a promising future study to reduce such collisions.

D. Offline Evaluations

To further analyze the impact of our proposal, we evaluate the capability of action prediction, a key subtask of navigation, in an offline setup. Since the action prediction corresponds to the regression of future waypoints in a *metric*-space, the higher awareness of geometry is expected to reflect a more accurate prediction. We used the fine-tuning dataset RealHSRNav, as well as the NoMaD pretraining dataset [9, 10, 11, 12], to verify (1) its prediction capability and (2) how our method preserves *previously learned* domain knowledge. Note that the depth maps for the NoMaD datasets are synthesized using Depth-Anything V2 [46] to enable RGB-D methods.

Experimental details. In all experiments, we use a $T{+}1 = 4$ frames history (from $t-3$ to t) and generate $H{+}1 = 8$ waypoints including the current step (i.e., $t, \ldots, t{+}7$), consistent with the real-world experiments. For comprehensiveness, we randomly sampled 100 independent sequences per dataset, and then averaged the error score (detailed later) across the sampled sequences. In each trial, we (1) set a fixed random seed for reproducibility, (2) randomly sampled observation window of $T + 1 = 4$ frames and a goal image, performed a single forward pass to predict the following $H + 1 = 8$ actions, (4) logged the metrics, and (5) proceeded to the next trial with a newly sampled observation-window and goal pair.

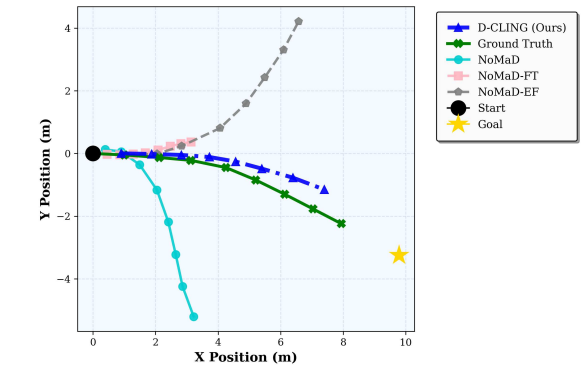
Metrics. We report *Average Displacement Error* (ADE) and *Final Displacement Error* (FDE), where ADE is the mean Euclidean position error over the prediction horizon and FDE is the Euclidean error at the terminal step $t{+}H$. To account for temporal misalignments while assessing trajectory fidelity, we additionally report the (normalized) Dynamic Time Warping (nDTW) distance [47], which has been shown to correlate well with human judgment of trajectory similarity.

Results and discussion. Table II shows quantitative results, and Figure 6 shows qualitative results of the evaluation. On RealHSRNav (the dataset used for fine-tuning), D-CLING attains the lowest ADE/FDE/DTW, outperforming the other baselines. This indicates that our proposed dense depth adaptation strategy offers the best future action prediction in the target environment. Intriguingly, the evaluation *on NoMaD datasets* demonstrates that fine-tuned D-CLING *on RealHSRNav* provides competitive or *even better* performance compared with the zero-shot NoMaD. We conjecture that our proposed strategy improved transferability to various scene domains, beyond the fine-tuning domain.

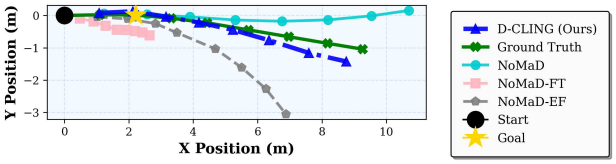
On the other hand, NoMaD-FT and NoMaD-EF, which lack a specific mechanism to alleviate catastrophic forgetting, yield inferior results on most domains compared with ours and the zero-shot model. These findings align with the discussion in Section V-C, where fine-tuning *rather degraded* the original zero-shot capability.

E. Benefit of Depth Conditioning (RGB vs RGB-D)

Finally, we investigated the effect of using depth, rather than RGB, given the conditioning strategy that D-CLING



(a) RealNaVHSR.



(b) NoMaD Dataset (GoStanford [11])

Fig. 6. **Bird's-eye-view predicted waypoint sequences**. These figures show waypoint predictions of length $H + 1 = 8$ generated from frame windows of length $T + 1 = 4$ on a representative subset of the offline evaluation set. Each panel overlays D-CLING (blue) and baselines—NoMaD (cyan), NoMaD-FT (pink), NoMaD-EF (gray)—with the ground-truth path in green and start/goal markers. D-CLING most closely follows the ground truth and maintains the most consistent heading toward the goal image point.

employed. We expected that the comparison between those two conditions would more clearly elucidate the importance of the geometric cue for novel-domain learning. Specifically, we implemented the RGB baseline by (1) dropping the depth-input channel from D-CLING, and (2) training with the same protocol (Sec. IV-C).

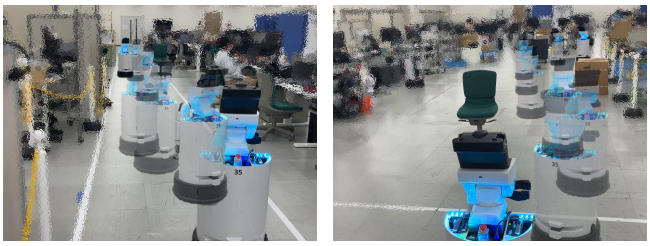
Scenarios. We compared the two strategies across two scenarios, which are fundamental to real-world navigation:

(i) **Single obstacle avoidance (*Single Obstacle*).** The robot traverses a corridor while avoiding a single stationary chair placed along its route.This scenario is similar to *Dynamic Corridor* in Section V-C and to scenarios contained in the training dataset.

(ii) **Multi obstacle avoidance (*Multi Obstacle*).** The robot navigates through a $15\,\mathrm{m} \times 5\,\mathrm{m}$ area in which three obstacles are placed at approximately uniform intervals in an alternating left-right arrangement along its direction of travel. The robot must avoid these obstacles while maintaining forward progress. Notably, this obstacle configuration is not represented in the training dataset and therefore lies outside the training distribution.

Metrics. For scenarios (i) and (ii), we run 10 trials each and report the success rate (SR).

Results and discussion. Table III reports real-robot results on two scenarios. **D-CLING outperforms the RGB baseline** in both cases. The largest gain is observed in scenario (ii), which is not represented in the training data. Contrarily, the baseline in (ii) usually stemmed from a delay in avoiding action. We underline that the result supports our



(i) *Single Obstacle* (ii) *Multi Obstacle*

Fig. 7. **Representative real-world navigation examples of the RGB-D modality (Ours) in two scenarios.** (i) *Single Obstacle*: corridor traversal with a single stationary chair. (ii) *Multi Obstacle*: avoidance of three obstacles in a zigzag trajectory.

TABLE III

RGB VS. RGB-D CONDITIONING. SUCCESS RATE (SR) FOR TWO SCENARIOS.

	(i) *Single Obstacle*	(ii) *Multi Obstacle*
Modality	SR(%)↑	SR(%)↑
RGB	60	10
RGB-D (Ours)	**80**	**100**

hypothesis, i.e., depth conditioning in our proposed manner strengthens geometric awareness and enables more robust navigation, even in a novel configuration.

VI. CONCLUSION AND FUTURE WORK

Zero-shot adoption of NFMs still suffers from novel scene observation, camera parameters, etc. Nevertheless, fine-tuning on a 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-conditioning strategy for NFM fine-tuning to enable novel scene learning. The comprehensive analysis demonstrated the efficacy of our proposal for robust navigation, as well as higher accuracy in action prediction. Moreover, our proposal enables more accurate action prediction beyond the fine-tuned domains, thereby further improving the zero-shot performance of NFMs.

Our experiments in this paper are limited to NoMaD, which is one example of NFMs. The proposed framework is designed in a largely model-agnostic manner and could potentially be applied to other NFMs. It remains an important direction for future work to evaluate the proposed method across a broader range of NFMs.

VII. ACKNOWLEDGMENTS

The authors used ChatGPT for limited language editing and grammar checking, and reviewed all AI-assisted text.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] 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.

[7] L. Suomela, N. Takahata, S. K. Arachchige, H. Edelman, and J.-K. Kämäräinen, "Data scaling for navigation in unknown environments," *arXiv*, 2026.

[8] 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.

[9] 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.

[10] D. Shah, B. Eysenbach, G. Kahn, N. Rhinehart, and S. Levine, "Rapid exploration for open-world navigation with latent goal models," *arXiv*, 2021.

[11] 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.

[12] H. Karnan, A. Nair, X. Xiao, G. Warnell, S. Pirk, A. Toshev, J. Hart, J. Biswas, and P. Stone, "Socially compliant navigation dataset (scand): A large-scale dataset of demonstrations for social navigation," *IEEE RA-L*, vol. 7, no. 4, pp. 11 807–11 814, 2022.

[13] 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.

[14] 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.

[15] J. Thoma, D. P. Paudel, A. Chhatkuli, T. Probst, and L. V. Gool, "Mapping, localization and path planning for image-based navigation using visual features and map." In *Proc. of the CVPR*, 2019, pp. 7383–7391.

[16] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[17] M. V. Gasparino, A. N. Sivakumar, Y. Liu, A. E. B. Velasquez, V. A. H. Higuti, J. Rogers, H. Tran, and G. Chowdhary, "WayFAST: Navigation With Predictive Traversability in the Field," *IEEE RA-L*, vol. 7, no. 4, pp. 10 651–10 658, 2022.

[18] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to explore using active neural slam." In *Proc. of the ICLR*, 2020.

[19] K. Stachowicz, L. Ignatova, and S. Levine, "Lifelong autonomous improvement of navigation foundation models in the wild." In *Proc. of the CoRL*, 2024.

[20] Y. Qiu, A. Pal, and H. I. Christensen, "Learning hierarchical relationships for object-goal navigation." In *Proc. of the CoRL*, 2020.

[21] A. Zhang, H. Sikchi, A. Zhang, and J. Biswas, "Creste: Scalable mapless navigation with internet scale priors and counterfactual guidance." In *Proc. of the RSS*, 2025.

[22] X. Liu, J. Li, Y. Jiang, N. Sujay, Z. Yang, J. Zhang, J. Abanes, J. Zhang, and C. Feng, "Citywalker: Learning embodied urban navigation from web-scale videos." In *Proc. of the CVPR*, 2025, pp. 6875–6885.

[23] Z. Feng, X. Chen, C. Shi, L. Luo, Z. Chen, Y.-H. Liu, and H. Lu, "Image-goal navigation using refined feature guidance and scene graph enhancement," *arXiv*, 2025.

[24] G. Bono, L. Antsfeld, B. Chidlovskii, P. Weinzaepfel, and C. Wolf, "End-to-end (instance)-image goal navigation through correspondence as an emergent phenomenon," *arXiv*, 2023.

[25] 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 *Proc. of the CVPR*, 2018, pp. 3674–3683.

[26] A. Kamath, P. Anderson, S. Wang, J. Y. Koh, A. Ku, A. Waters, Y. Yang, J. Baldridge, and Z. Parekh, "A new path: Scaling vision-and-language navigation with synthetic instructions and imitation learning." In *Proc. of the CVPR*, 2023, pp. 10 813–10 823.

[27] N. Hirose, C. Glossop, A. Sridhar, D. Shah, O. Mees, and S. Levine, "Lelan: Learning a language-conditioned navigation policy from in-the-wild videos," *arXiv*, 2024.

[28] D. Song, J. Liang, A. Payandeh, A. H. Raj, X. Xiao, and D. Manocha, "Vlm-social-nav: Socially aware robot navigation through scoring using vision-language models," *IEEE RA-L*, 2024.

[29] Y. Zhang, H. Tan, and M. Bansal, "Diagnosing the environment bias in vision-and-language navigation." In *Proc. of the IJCAI*, 2020, pp. 890–897.

[30] X. Y. Tianqi Tang Heming Du and Y. Yang, "Monocular camera-based point-goal navigation by learning depth channel and cross-modality pyramid fusion." In *Proc. of the AAAI*, 2022, 36(5), 5422–5430.

[31] W. Cai, J. Peng, Y. Yang, Y. Zhang, M. Wei, H. Wang, Y. Chen, T. Wang, and J. Pang, "Navdp: Learning sim-to-real navigation diffusion policy with privileged information guidance," *arXiv*, 2025.

[32] H. Wang, A. H. Tan, A. Fung, and G. Nejat, "X-nav: Learning end-to-end cross-embodiment navigation for mobile robots," *arXiv*, 2025.

[33] T. L. Team, "A careful examination of large behavior models for multitask dexterous manipulation," *arXiv*, 2025.

[34] A. O' Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, et al., "Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0." In *Proc. of the ICRA*, IEEE, 2024, pp. 6892–6903.

[35] K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al., "$\pi_{0.5}$: A vision-language-action model with open-world generalization," *arXiv*, 2025.

[36] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, et al., "Gr00t n1: An open foundation model for generalist humanoid robots," *arXiv*, 2025.

[37] Z. Liu, J. Zhang, K. Asadi, Y. Liu, D. Zhao, S. Sabach, and R. Fakoor, "Tail: Task-specific adapters for imitation learning with large pretrained models." In *Proc. of the ICLR*, 2024.

[38] K. Lu, K. T. Ly, W. Hebberd, K. Zhou, I. Havoutis, and A. Markham, "Learning generalizable manipulation policy with adapter-based parameter fine-tuning." In *Proc. of the IROS*, 2024.

[39] M. Sharma, C. Fantacci, Y. Zhou, S. Koppula, N. Heess, J. Scholz, and Y. Aytar, "Lossless adaptation of pretrained vision models for robotic manipulation," *arXiv*, 2023.

[40] K. Bousmalis, G. Vezzani, D. Rao, C. Devin, A. X. Lee, M. Bauzá, T. Davchev, Y. Zhou, A. Gupta, A. Raju, et al., "Robocat: A self-improving generalist agent for robotic manipulation," *arXiv*, 2023.

[41] 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.

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

[43] 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.

[44] 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.

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

[46] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," *NeurIPS*, vol. 37, pp. 21 875–21 911, 2024.

[47] G. Ilharco, V. Jain, A. Ku, E. Ie, and J. Baldridge, "General evaluation for instruction conditioned navigation using dynamic time warping," *arXiv*, 2019.