Robust Scene Transfer for PointGoal Navigation via Privileged-Sensor–Guided Contrastive Learning

Amirhossein Zhalehmehrabi, Tiziano Tezze, Alberto Castellini, and Alessandro Farinelli

***Abstract*—We propose a sensor-guided adaptive contrastive learning framework for visual representation learning in PointGoal navigation. During training, privileged LiDAR sensing guides the contrastive objective through a geometry-aware similarity metric and adaptive temperature scaling, encouraging visual embeddings to capture navigation-relevant structure rather than scene-specific appearance. The resulting encoder is pretrained independently, frozen, and used as the perceptual backbone for reinforcement learning, decoupling representation learning from policy optimization. We further introduce a cross-stage domain mismatch between representation pretraining and policy learning to suppress environment-specific shortcuts and promote reliance on task-relevant features. Extensive experiments in high-fidelity simulation demonstrate that our approach significantly improves policy-level scene transfer across diverse indoor and outdoor environments. At deployment, the agent relies only on monocular RGB observations together with standard task-related inputs such as goal position and proprioceptive signals, without access to LiDAR or other privileged sensors. Our method outperforms large pretrained vision models, navigation-specific representation learning approaches, domain randomization methods, and standard contrastive baselines under severe appearance and semantic shifts. We also release a multimodal dataset to support future research on privileged-guided visual representation learning for navigation.**

***Index Terms*—Reinforcement learning, representation learning, vision-based navigation.**

I. Introduction

VISION-BASED mobile navigation promises scalable autonomy by enabling robots to operate using only onboard cameras, avoiding reliance on additional sensing modalities such as depth or LiDAR. Despite this appeal, vision-based navigation policies often fail when deployed in visually different environments, even when task dynamics and robot embodiment remain unchanged [1]. This lack of *scene transfer*—the ability to reliably perform the same task across visually distinct environments—remains a major barrier to real-world deployment of learning-based navigation systems.

At the core of this challenge lies representation learning. To generalize across environments, a navigation policy must rely on visual representations that suppress scene-specific appearance while preserving task-relevant structure such as goal geometry, free space, and obstacle layout. While prior work has demonstrated strong performance within fixed environments [2], [3], [4], learning representations that generalize across environments without retraining remains challenging, particularly under large visual distribution shifts.

In contrast to robotic manipulation, where scene transfer has been extensively studied [5], vision-based navigation methods are typically evaluated in visually similar environments or rely on explicit geometric representations, such as mapping, SLAM, or depth sensing [6], [7], [8], [9].

In another line of research, contrastive representation learning provides a principled mechanism for enforcing invariance by aligning semantically similar observations [10]. However, standard contrastive methods rely on heuristic augmentations and do not explicitly incorporate task structure, often resulting in representations that remain entangled with scene appearance under large cross-environment shifts [11].

Recent work [12] incorporates privileged supervision into contrastive learning for agile flight. However, this approach is trajectory-centric and demonstrated only for executing predefined motion patterns, limiting its applicability to general closed-loop navigation. Moreover, reliance on imitation learning reduces robustness in long-horizon tasks due to limited recovery outside the training distribution [13].

These limitations highlight a key gap: policy-level scene transfer requires representations that support scene-agnostic decision-making. More broadly, this work explores a research direction in which privileged sensing modalities guide the representation learning of deployable sensors. Rather than relying on these sensors at inference time, we use them during training to shape the feature extractor of the target sensor. One promising direction is to leverage sensing modalities that provide direct access to geometric structure, which is inherently less sensitive to appearance variations across environments. In this work, we instantiate this idea by using LiDAR observations to guide the learning of an RGB encoder. In this work, we use privileged geometric sensing available during training to guide representation learning, without requiring such sensors at deployment. Specifically, we incorporate geometric observations into the contrastive objective as an additional supervisory signal, encouraging the

Received 16 March 2026; accepted 2 August 2026. Date of publication 28 August 2026; date of current version 9 September 2026. This article was recommended for publication by Associate Editor Z. Wu and Editor P. Vasseur upon evaluation of the reviewers' comments. This work was supported by PNRR iNEST Project under Grant ECS00000043. *(Corresponding author: Amirhossein Zhalehmehrabi.)*

The authors are with the Department of Computer Science, University of Verona, 37129 Verona, Italy (e-mail: amirhossein.zhalehmehrabi@univr.it; tiziano.tezze@univr.it; alberto.castelini@univr.it; alessandro.farinelli@univr.it).

The code is available at: https://github.com/zhalehmehrabi/privileged-sensor-contrastive-nav.

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2026.3728340, provided by the authors.

Digital Object Identifier 10.1109/LRA.2026.3728340

2377-3766 © 2026 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

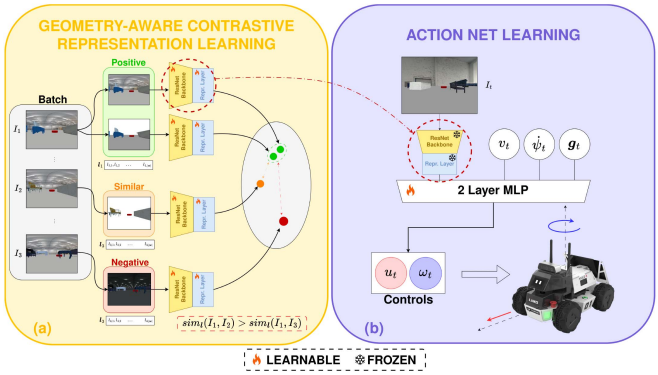


Fig. 1. Overall pipeline. (a) A ResNet backbone is first trained with geometry-aware contrastive learning, (b) then frozen and used alongside robot state inputs to train an MLP policy that outputs navigation controls.

learned representation to encode the underlying geometric structure of the scene. This geometry-guided regularization steers the representation away from appearance-specific features and toward features that remain stable across environments, resulting in more robust and transferable policies.

Specifically, we propose a *sensor-guided adaptive contrastive learning* framework in which privileged LiDAR observations modulate the contrastive objective through geometry-aware adaptive temperature scaling. The resulting visual encoder is pretrained independently, frozen, and used as the perceptual backbone for reinforcement learning (RL). At deployment, the navigation policy operates using only deployable sensing modalities, including monocular RGB observations and standard task inputs such as goal-relative pose and proprioceptive measurements, without access to the privileged sensing used during representation pretraining.

We summarize our contributions as follows:

- *Sensor-guided adaptive contrastive learning:* We propose a framework in which privileged sensors guide the representation learning of a target sensor by modulating the contrastive objective. We instantiate this paradigm using LiDAR observations to guide an RGB encoder.
- *Policy-level scene transfer with decoupled pretraining and RL:* We demonstrate that freezing a sensor-guided pretrained visual-based encoder and training navigation policies via RL yields robust closed-loop generalization across visually distinct environments.
- *Comprehensive evaluation and dataset release:* We demonstrate strong cross-scene generalization on PointGoal navigation and release a multimodal dataset to support future research on privileged-guided visual representation learning.

II. Related Works

A. End-to-End Policy Learning

End-to-end sensorimotor learning jointly optimizes perception and control directly from raw sensory observations, bypassing explicit geometric modeling [14]. In PointGoal navigation, map-based approaches integrate spatial representations and planning, while mapless approaches learn policies directly from sensory inputs. However, mapless methods typically rely on rich sensing and task rewards as primary supervision, which does not explicitly enforce scene-invariant representations and often leads to visual overfitting.

Our work explicitly targets scene transfer in vision-based navigation by decoupling representation learning from policy optimization—learning task-relevant visual features first, then training a policy on the frozen representation. Unlike prior geometry-driven approaches [15], [16], we learn transferable features through contrastive pretraining.

B. Visual Pre-Training for Robotics

Visual pre-training has emerged as an effective strategy for improving generalization and sample efficiency in robotic learning by enabling reusable representations [17]. Large-scale pretrained models such as CLIP [18] and Masked Autoencoders (MAE) [19] have demonstrated strong cross-task transfer in robotic perception and control [20].

Self-supervised and contrastive learning methods provide a complementary approach by shaping representations according to task dynamics rather than raw appearance. Contrastive pretraining, popularized by methods such as SimCLR [10], as well as subsequent extensions in robotics [21], and contrastive RL approaches [22], improve robustness by learning invariant, task-relevant features. Representative paradigms include trajectory-based contrastive learning, which pulls future states closer [23], and augmentation-based methods that enforce consistency across transformed observations [7].

[12] proposed an adaptive contrastive learning framework guided by privileged signals in a controlled flight setting. However, this approach relies on predefined trajectories and teacher supervision, limiting its applicability to autonomous navigation. In contrast, our method uses auxiliary sensory information available only during training to adaptively modulate the contrastive objective, without requiring privileged environmental states or teacher policies.

C. Privileged Visual Representation Learning

Privileged supervision leverages sensors or state information available during training but not at deployment to facilitate representation learning and policy optimization. In robotics, this paradigm has been widely used in teacher–student or "learning-by-cheating" frameworks, where controllers trained with full-state or geometric inputs are distilled into vision-based policies [14].

More recently, privileged information has been integrated into self-supervised and contrastive learning by guiding sampling strategies or modulating contrastive objectives [12]. These approaches typically rely on trajectory structure, expert supervision, or explicit state regression, which restricts their applicability to general closed-loop navigation.

In contrast, our approach uses privileged geometric sensing solely during training to adaptively scale a self-supervised contrastive objective, without teacher policies or fixed trajectories.

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

III. Methodology

We propose a vision-based robot navigation framework that separates visual representation learning from policy optimization. The visual encoder is pretrained with a privileged-sensor–guided contrastive objective using replay correspondences and geometric observations available only during representation learning. Once pretraining is complete, the encoder is frozen and used as a feature extractor for a lightweight policy trained via model-free RL. No privileged observations, replay trajectories, or cross-scene correspondences are used during policy training or evaluation.

A. Task Formulation

We study the PointGoal Navigation task, where a robot must reach a target relative to its start, avoiding obstacles. At each timestep, it receives partial observations and outputs continuous linear and angular velocities. Episodes end on success or collision. During deployment, the robot observes the goal in its local frame, velocities, and an RGB image, without access to obstacle geometry or absolute position. The aim is to learn a navigation policy that generalizes across visually diverse environments using only these sensors.

B. Geometry-Aware Contrastive Representation Learning

The objective of the representation learning stage is to train a vision encoder that produces embeddings aligned with navigation-relevant geometry rather than scene appearance. We build upon standard contrastive learning, but incorporate privileged geometric information available during training to modulate the loss in a task-aware manner.

Training batches are sampled uniformly. Positives are augmented views of scene-invariant counterparts, while all other samples in the batch are treated as negatives. Instead of altering this pair assignment, we adjust the contribution of each pair to the contrastive objective through a geometry-adaptive temperature mechanism.

1) Contrastive Learning Objective: Let $\phi_{\boldsymbol{\theta}}$ denote a vision encoder parameterized by $\boldsymbol{\theta}$ that maps an RGB image I_i to a latent embedding $\mathbf{z_i} = \phi_{\boldsymbol{\theta}}(I_i) \in \mathbb{R}^d$. Given a random batch of N images, we train $\phi_{\boldsymbol{\theta}}$ to minimize the *InfoNCE objective* [24] for each positive pair in the batch:

$$\mathcal{L}_{i,j} = -\log \frac{\exp\left(\mathrm{sim}(\mathbf{z_i}, \mathbf{z_j})/\tau\right)}{\sum_{k=1}^{N} \mathbb{1}_{[k\neq i]} \exp\left(\mathrm{sim}(\mathbf{z_i}, \mathbf{z_k})/\tau\right)} \tag{1}$$

where $\mathrm{sim}(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature parameter. In contrast to standard formulations, τ is not fixed but adaptively modulated based on geometric similarity, as described in Section III-B3.

2) Task-Aware Similarity Metrics: Although pair assignment follows standard contrastive practice, uniform treatment of negatives is suboptimal for navigation. Visually distinct observations may correspond to nearly identical local geometric configurations, and forcing strong repulsion between such embeddings conflicts with the objective of learning geometry-aligned representations.

To address this, we compute a geometry-based similarity using LiDAR readings $\boldsymbol{l}$ available during training. Given two observations I_1 and I_2 with corresponding vectors $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$, we define their *normalized geometric similarity* as

$$sim_{\boldsymbol{l}}(I_1, I_2) = 1 - \frac{\sqrt{\sum_{i=1}^{\ell} w_i (\boldsymbol{l}_{1,i} - \boldsymbol{l}_{2,i})^2}}{\sqrt{\sum_{j=1}^{\ell} w_j}} \tag{2}$$

where $\ell = |\boldsymbol{l}|$ is the number of LiDAR beams, and $w_i \geq 0$ is the weight assigned to the i-th beam; their selection is detailed in Section V-A1.

The subtracted term corresponds to a normalized weighted Euclidean distance in LiDAR space, bounded in [0,1], such that higher values of $sim_{\boldsymbol{l}}$ indicate greater similarity in the local spatial structure surrounding the robot.

3) Geometry-Adaptive Temperature: Contrastive learning objectives are known to be sensitive to the temperature parameter [12], [25], [26], which controls the strength of attraction and repulsion in embedding space. We therefore modulate the temperature on a per-pair basis according to geometric similarity, allowing repulsive forces to vary continuously with navigation-relevant structure.

Given a contrastive pair (I_1, I_2) with geometric similarity $sim_{\boldsymbol{l}}(I_1, I_2)$, we define the adaptive temperature as

$$\tau(I_1, I_2) = \begin{cases} \alpha & \text{if}(I_1, I_2)\text{is a positive pair,} \\ sim_{\boldsymbol{l}}(I_1, I_2) & \text{if}(I_1, I_2)\text{is a negative pair.} \end{cases} \tag{3}$$

The resulting value is mapped to a bounded interval $[\tau_{\min}, \tau_{\max}]$ to ensure numerical stability:

$$\tau'(I_1, I_2) = \tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \tau(I_1, I_2) \tag{4}$$

This adaptive formulation ensures that geometrically similar observations, even when treated as negatives, apply weaker repulsive forces, while geometrically dissimilar observations are pushed apart more strongly. As a result, the learned latent space reflects navigation-relevant structure rather than purely visual appearance. In our framework, privileged LiDAR observations are used exclusively during simulation-based representation pretraining and are never available at deployment. Since privileged sensing operates entirely within the simulator, geometric measurements are noise-controlled by construction, and we assume perfect sensing during training. The effect of degraded privileged sensing is therefore outside the scope of this work.

C. Action Net Learning

The navigation policy is composed of a frozen vision encoder and a lightweight action network. At each timestep t, the policy receives an observation o_t consisting of the visual embedding $\mathbf{z_t} = \phi_{\boldsymbol{\theta}}(I_t)$ extracted from the RGB image I_t, the robot's linear and angular velocities, $v_t, \dot{\psi}_t$ respectively, and the relative position of the goal in polar coordinates $\mathbf{g}_t$. Formally, the observation is defined as:

$$o_t = \left\{\mathbf{z_t},\, v_t,\, \dot{\psi}_t,\, \mathbf{g}_t\right\} \tag{5}$$

The action network maps observation to continuous control commands: linear velocity u_t and angular velocity Ω_t.

The action network is optimized using a model-free RL algorithm. By fixing the visual representation during policy learning, the action network is trained on a stable and scene-invariant input distribution, which helps reduce overfitting to the training

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

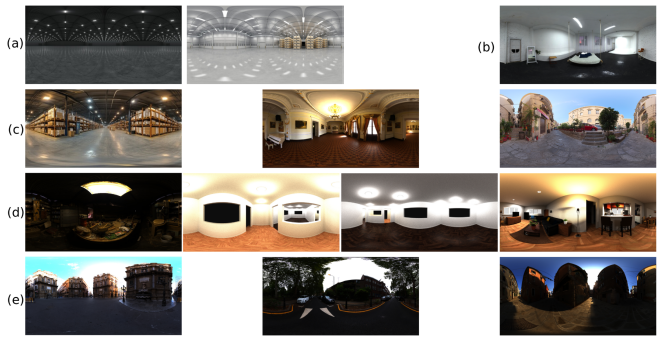


Fig. 2. Overview of datasets and evaluation environments. Visual appearance differences are shown; obstacle counts and placements are not depicted. (a) Environments for visual encoder pre-training, Warehouse-1&2 (including variants without background and without floor/background). (b) RL training environment, Photo-studio. (c) Environments for representation analysis, Warehouse-3, Ballroom, Palermo. (d) Indoor test environments, Carpentry-shop, Probe-2&3, Rs. (e) Outdoor test environments, Quattro-canti, Urban-street, Venetian-crossroads.

environments and isolates the effect of representation learning on policy generalization.

The reward function is inspired by [27] and is defined as:

$$R(s_t, a_t, s_{t+1}) = \begin{cases} r_{\text{suc}} & \text{if success,} \\ r_{\text{collide}} & \text{if collision,} \\ -\alpha_1 \Delta_d - \alpha_2 r_{\text{back}} - \alpha_3 r_{\text{angular}} & \text{otherwise,} \end{cases} \quad (6)$$

where Δ_d denotes the change in geodesic distance to the goal between consecutive states, $r_{\text{back}} = |\min(u_t, 0)|$ penalizes backward motion, and $r_{\text{angular}} = |\Omega_t|$ penalizes angular velocities. Penalizing angular velocity encourages smoother trajectories and more stable visual observations, which improves robustness when operating with vision-based perception. The reward is kept identical across all environments.

D. Cross-Stage Domain Mismatch (CSDM)

The environments used for representation pretraining (Fig. 2(a)) differ from those used during action net learning (Fig. 2(b)). This cross-stage domain mismatch is intentional and serves to isolate the effect of representation learning on policy generalization. By freezing the pretrained encoder, the visual feature space remains fixed during RL. Training the policy in a distinct environment discourages reliance on scene-specific correlations that may arise during pretraining, as such correlations do not transfer across environments. Consequently, policy optimization is forced to exploit features that remain predictive under domain shift, providing a stronger test of geometry-aligned representation learning.

IV. Dataset and Data Collection

We collect navigation episodes by replaying trajectories from a privileged *state-based* agent—trained with full geometry access but no visual inputs—across visually distinct environments sharing identical geometry, yielding observations of the same navigation-relevant configurations under varying appearances.

TABLE I
Dataset Specifications

Parameter	Value
#Environments	8
#Images per Env.	36k
#Objects in Env.	10
#Room Settings	20
#Object Categories	13
#Distinct Objects	126
#Source Models	4
#Trajectories per Model	100

At each timestep, we record RGB images, LiDAR-derived vectors, depth images, and semantic segmentation masks. For pretraining, only the LiDAR-derived vector is used to define task-relevant similarity in the contrastive objective; the remaining modalities are included for completeness.

The dataset spans diverse indoor and outdoor environments, including three warehouse layouts, Photo-studio, ballroom, Palermo-sidewalks, and no-background/floor variants (Table I). To increase scene diversity, we generated 20 room configurations for each environment. Each configuration specifies the spatial arrangement, orientation, and object category assigned to each placement location. Every configuration contains 10 obstacle objects. During data collection, object instances are sampled from a pool of 126 assets belonging to 13 categories, allowing different object appearances to occupy the same semantic role across episodes while preserving the underlying room layout. Object placements therefore vary across room configurations, while object instances are randomized within each category. Four environments are used for representation pretraining (Fig. 2(a)), where one warehouse serves as the anchor and others provide scene-invariant counterparts. The Photo-studio environment is reserved for RL training (Fig. 2(b)), and three environments are held out entirely for out-of-distribution validation (Fig. 2(c)). All privileged information is used solely during representation learning and is never available to the navigation policy at deployment. [1]

V. Experiments

We evaluate our approach through representation analysis and downstream navigation under distribution shift.

A. Implementation Details and Model Training

1) Vision Encoder Pre-Training: Using the high-fidelity simulator *iGibson* [28], we collected a dataset for pretraining. We conduct simulation experiments using TurtleBot platforms with a differential-drive configuration. The visual encoder is trained using the geometry-adaptive contrastive objective described in Section III-B, including the scene-invariant pairing strategy. No additional image augmentations are applied. Architecture details and optimization hyperparameters are reported in Table II. For

[1]The **G**eometry-Guided **R**epresentation for **A**utonomous **N**avigation (GRAN) dataset is publicly available for download at the following link:https://huggingface.co/datasets/Tr3itz/Geometry-Guided_Representation_for_Autonomous_Navigation

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

TABLE II
TRAINING HYPERPARAMETERS FOR VISUAL PRETRAINING, REPRESENTATION ANALYSIS AND POLICY LEARNING

Stage	Parameter	Value
Pretrain	RGB Input dim	224×224
	Backbone	ResNet-50 (IN)
	#Projection layers	2
	Projection dim d	256
	Batch size	8192
	Optimizer	Adam
	LR	1e−4 (linear decay)
	$\tau_{\min}/\tau_{\max}$	0.1/10
	α (pos. temp.)	0.1
Representation analysis	k-NN k	10
RL (SAC)	r_{suc} / $r_{collide}$	10 / -10
	α_1 / $\alpha_{2,3}$	1 / -0.1
	Actor / Critic LR	3e−4 / 3e−4
	Batch size	1024
	Replay buffer	1.5e6
	Discount γ	0.99
	Target update τ	0.005
	Entropy coeff.	automatic

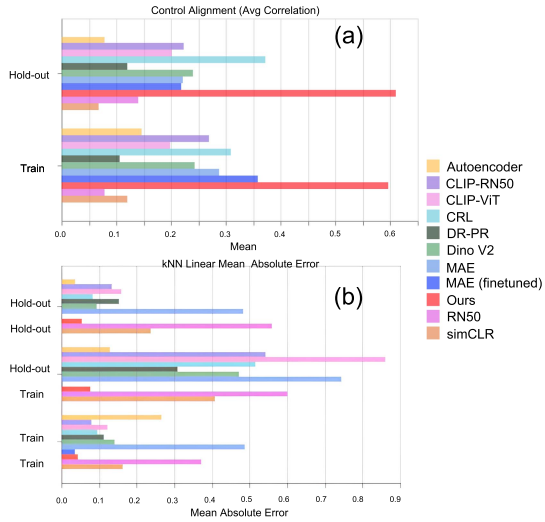


Fig. 3. Control alignment and local consistency across scenes. (a) Correlation between embedding distances and control differences (higher is better). (b) k-NN action prediction mean absolute error across scene pairs (lower is better), evaluating local control consistency and transferability.

weights $\boldsymbol{w}$ in (2), beams inside the FOV are assigned unit weight $w_i = 1$. Outside the FOV, weights decay smoothly and symmetrically toward 0.1 via a sigmoid.

2) Action Network Learning: The navigation policy consists of the frozen visual encoder and a two-layer MLP with hidden dimension 256. No recurrent components are used to isolate the effect of representation quality. Policy optimization uses Soft Actor-Critic (SAC) [29]. Policy training is conducted exclusively in the Photo-studio environment (Fig. 2(b)), which is not seen during encoder pretraining.

For evaluation, we use seven environments unseen during visual pretraining, representation analysis, and policy learning: four indoor scenes (carpentry-shop, probe-2, probe-3, Rs; Fig. 2(d)) and three outdoor scenes (quattro-canti, urban-street-01, venetian-crossroads; Fig. 2(e)). Each environment is tested under uniform- and textured-floor variants to induce controlled appearance and semantic shifts.

3) Baselines: We compare against three categories of visual representation baselines: (i) frozen pretrained models not exposed to our dataset, including ResNet50 [30], CLIP-ResNet/ViT [18], MAE [19], DINOv2 [31], and CRL [7], a navigation-specific curiosity-driven contrastive representation; (ii) representations trained on our custom dataset, namely AutoEncoder (AE) [32] with a ResNet50 backbone, MAE with pretrained weights, and SimCLR [10], which incorporate domain randomization at the representation-learning stage; and (iii) two SimCLR-based policy-stage domain randomization baselines, DR-PR (trained on the same environments used for pretraining) and DR-CSDM (trained on held-out environments, inducing a cross-stage domain mismatch). Direct comparison with [12] was not possible, as their approach assumes predefined trajectories for both training and evaluation.

B. Representation Analysis

We evaluate representations on both training and hold-out scenes illustrated in Figs. 2(a) and 2(c). Hold-out scenes are used to validate cross-scene generalization.

1) Embedding–Action Distance Correlation: To quantify whether the learned representation preserves task-relevant geometry, we measure the correlation between distances in the embedding space and corresponding differences in control commands. The idea behind this analysis is that a good encoder should generate distant embeddings for pairs of images requiring different actions, and close embeddings for pairs of images requiring similar actions. This evaluation is inspired by representational similarity analysis [33] and prior studies examining geometric structure in learned embeddings [34]. Unlike classification-based probes, this metric directly evaluates alignment between latent structure and downstream control signals, which is critical for sensorimotor transfer.

As shown in Fig. 3(a), our method achieves substantially higher control alignment than all baselines in both training and hold-out scenes. Importantly, the correlation remains stable under scene shift, whereas general-purpose visual encoders exhibit significant degradation. These results indicate that the proposed objective induces a representation whose geometry reflects control-relevant structure rather than purely visual similarity, enabling transfer beyond the visual distribution observed during training.

2) K-Nearest Neighbor Action Prediction: We further assess local control consistency using a non-parametric k-nearest neighbor (kNN) regression probe. For a query embedding, control commands are predicted as the average of its k nearest neighbors retrieved via cosine similarity, and performance is measured using mean absolute error. This frozen kNN protocol is commonly used to evaluate representation quality independent of task-specific fine-tuning [35], [36].

Results in Fig. 3(b) show that our representation achieves consistently low kNN mean absolute error across scene combinations, with limited degradation under scene shift. The MAE-finetuned encoder may reach impressive kNN error rates, but these gains are deceptive. The representation lacks a robust embedding–action correlation (Fig. 3(a)) and, as evidenced in Section V-C, ultimately fails to generalize.

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.



Fig. 4. UMAP projection of learned visual embeddings from trajectories collected in three held-out environments.

This discrepancy suggests that the finetuned encoder may overfit to dataset-specific local statistics, yielding favorable nearest-neighbor regression performance without preserving globally consistent control-aware geometry. In contrast, our method maintains both strong global alignment and stable local structure, supporting robust transfer.

3) U-Map Visualization: We visualize embeddings from held-out environment trajectories using UMAP. The three environments share identical trajectory configurations, differing only in visual appearance. Fig. 4 shows that observations from visually distinct scenes organize along a coherent latent manifold, with embeddings from different environments interleaved along shared branches rather than forming scene-specific clusters. Notably, trajectory-like structure emerges across environments without explicit temporal or topological regularization. We attribute this to the adaptive temperature mechanism: temporally nearby observations tend to share high geometric similarity in LiDAR space, causing adaptive τ to soften their mutual repulsion and keeping them naturally close in the latent space. This suggests the representation captures navigation-relevant geometry that generalizes beyond scene appearance.

C. Navigation Performance Under Distribution Shift

We evaluate downstream navigation using a frozen visual encoder for all methods and train a single policy architecture with identical hyperparameters to isolate representation quality. Each method trains with a frozen visual encoder and identical hyperparameters (five seeds) in Photo-studio (Fig. 2(b)), except DR-PR and DR-CSDM which additionally use training and held-out environments respectively. Evaluation uses test environments (Fig. 2(d) & (e)) over 100 episodes per environment. We report Success Rate (SR) and Success weighted by Path Length (SPL) [37]. Results are shown in Table III and Table IV.[2]

1) In-Distribution Performance: All methods achieve high SR and SPL (typically $> 80\%$ SR) in the training environment (see column "Train" in Tables III or IV), showing the task is solvable with sufficient interaction data. However, strong in-distribution performance does not imply robustness: methods with similar training SR behave very differently under distribution shift.

[2]Since domain mismatch III-D was also used for the MAE fine-tuned, the model overfit to the pretraining environments, resulting in lower performance than the base MAE model even in the training environment.

TABLE III
NAVIGATION PERFORMANCE UNDER INCREASING APPEARANCE AND SCENE SHIFTS. POLICIES ARE TRAINED IN A SINGLE ENVIRONMENT AND EVALUATED ON UNSEEN TEST SCENES. SR AND SPL ON THE TEST ENVIRONMENTS ARE AVERAGED OVER 5 SEEDS AND ALL ENVIRONMENTS IN EACH CLASS. THE BEST PERFORMANCE ARE HIGHLIGHTED IN BOLD, AND THE SECOND BEST ARE UNDERLINED.

Method	Custom Dataset	Train		Test Simple		Test Hard Simple Floor		Test Hard Textured Floor	
		SR↑	SPL↑	SR↑	SPL↑	SR↑	SPL↑	SR↑	SPL↑
CL-RN [18]	×	<u>92.78</u> ±11.47	<u>91.14</u> ±12.76	38.80 ±13.12	33.00 ±13.11	33.72 ±10.0	28.46 ±9.42	10.12 ±8.81	8.48 ±7.66
CL-VT [18]	×	87.16 ±5.19	83.25 ±6.90	<u>52.00</u> ±8.97	<u>44.66</u> ±4.40	24.07 ±9.95	19.41 ±8.77	6.9 ±7.16	5.59 ±6.07
MAE [19]	×	**93.40** ±2.31	**91.48** ±2.31	37.20 ±15.87	30.22 ±12.18	18.22 ±9.64	15.28 ±8.33	6.82 ±7.54	5.93 ±6.62
Resnet [30]	×	87.40 ±3.43	82.83 ±5.45	44.4 ±26.25	34.97 ±20.23	25.40 ±15.27	18.85 ±13.11	5.3 ±5.26	3.97 ±4.15
DINOv2 [31]	×	82.00 ±3.16	77.16 ±4.60	42.80 ±14.02	34.15 ±14.11	37.35 ±5.96	32.33 ±6.35	15.2 ±8.6	12.77 ±7.3
CRL [7]	×	88.4 ±5.68	84.47 ±5.27	0.0 ±0.0	0.0 ±0.0	9.45 ±8.23	8.34 ±7.38	9.77 ±6.83	8.59 ±6.33
AE [32]	✓	85.2 ±6.18	81.95 ±6.31	0.0 ±0.0	0.0 ±0.0	0.4 ±0.89	0.26 ±0.58	0.62 ±1.16	0.39 ±0.73
MAE [19] [2]	✓	90.8 ±4.43	87.18 ±3.64	0.4 ±0.54	0.11 ±0.19	2.27 ±4.56	1.91 ±3.95	0.85 ±1.59	0.71 ±1.43
SimCLR [10]	✓	82.2 ±9.98	77.86 ±10.29	39.4 ±14.77	33.36 ±10.24	51.07 ±8.61	44.76 ±7.37	<u>45.7</u> ±12.7	<u>39.39</u> ±11.43
DR-PR	✓	76.2 ±7.25	74.71 ±6.65	45.0 ±19.44	37.98 ±15.62	44.9 ±11.94	38.73 ±9.54	30.02 ±10.45	25.73 ±9.32
DR-CSDM	✓	77.4 ±8.98	75.66 ±9.42	28.2 ±5.01	24.46 ±4.50	<u>52.52</u> ±7.82	<u>44.86</u> ±7.51	42.22 ±9.10	36.47 ±8.43
Ours	✓	83.8 ±7.46	79.62 ±7.19	**82.6** ±5.68	**78.82** ±5.99	**77.7** ±4.28	**73.86** ±4.08	**49.42** ±11.11	**43.36** ±9.67

TABLE IV
GENERALIZATION ACROSS INDOOR AND OUTDOOR ENVIRONMENTS. POLICIES ARE TRAINED IN A SINGLE INDOOR SCENE AND EVALUATED ON UNSEEN ENVIRONMENTS. RESULTS ARE AVERAGED OVER 5 SEEDS (MEAN ± STD).

Method	Custom Dataset	Train		Indoor		Outdoor	
		SR↑	SPL↑	SR↑	SPL↑	SR↑	SPL↑
CL-RN [18]	×	<u>92.78</u> ±11.47	<u>91.14</u> ±12.76	19.15 ±10.04	16.02 ±8.92	24.7 ±8.84	20.92 ±8.17
CL-VT [18]	×	87.16 ±5.19	83.25 ±6.9	17.52 ±10.10	14.29 ±8.50	13.45 ±7.01	10.71 ±6.34
MAE [19]	×	**93.40** ±2.31	**91.48** ±2.31	19.22 ±9.69	16.59 ±8.97	5.82 ±7.49	4.62 ±5.97
Resnet [30]	×	87.40 ±3.43	82.83 ±5.45	17.97 ±11.7	13.43 ±9.91	12.72 ±8.82	9.4 ±7.35
DINOv2 [31]	×	82.00 ±3.16	77.16 ±4.60	38.37 ±7.87	32.85 ±7.68	14.17 ±6.69	12.25 ±5.96
CRL [7]	×	88.4 ±5.68	84.47 ±5.27	12.63 ±8.39	11.23 ±7.76	8.02 ±7.40	7.04 ±6.74
AE [32]	✓	85.2 ±6.18	81.95 ±6.31	0.7 ±1.49	0.49 ±1.05	0.32 ±0.55	0.16 ±0.26
MAE [19]	✓	90.8 ±4.43	87.18 ±3.64	1.27 ±2.27	0.97 ±1.8	1.85 ±3.89	1.66 ±3.57
SimCLR [10]	✓	82.2 ±9.98	77.86 ±10.29	<u>50.14</u> ±10.17	<u>43.56</u> ±9.25	47.3 ±11.92	41.51 ±10.43
DR-PR	✓	76.2 ±7.25	74.71 ±6.65	41.50 ±9.97	35.96 ±9.25	35.91 ±10.57	30.81 ±9.46
DR-CSDM	✓	77.4 ±8.98	75.66 ±9.42	46.82 ±8.79	39.86 ±7.87	<u>49.82</u> ±8.14	<u>43.40</u> ±7.99
Ours	✓	83.8 ±7.46	79.62 ±7.19	**68.65** ±7.18	**63.63** ±6.58	**60.89** ±8.5	**55.79** ±7.6

2) Appearance and Scene Shifts: We first evaluate robustness to controlled visual perturbations (See "Test Simple" in Table III). Under simple appearance changes, the floor and background are replaced with two uniform colors, introducing a global appearance shift while preserving scene geometry, all the baselines lose 35%–55% in SR relative to training. Methods finetuned on the custom dataset (AE, MAE) collapse entirely. In contrast, our method retains 82.6% SR, maintaining performance close to the training regime.

Under more severe scene changes with fixed floor appearance (Fig. 2(d) & (e), but with uniform color as floor as *Test Hard – Simple Floor*), baseline performance further degrades, but simCLR got 51.07% SR. Our approach achieves 77.7% SR, substantially outperforming all baselines and reducing the

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

TABLE V
ABLATION STUDY OF REPRESENTATION LEARNING COMPONENTS

RL DR	Adaptive τ	CSDM	Scene Inv.	Vis. Augs.	SR ↑	SPL ↑
×	×	×	×	✓	48.38 ±10.6	42.07 ±9.4
×	×	×	✓	✓	43.07 ±9.18	37.77 ±8.67
×	×	×	✓	×	50.20 ±5.78	47.08 ±5.87
×	×	✓	✓	×	59.86 ±7.95	55.31 ±7.47
×	✓	✓	✓	×	**63.56** ±7.7	**58.61** ±6.87
✓	✓	✓	✓	×	54.27 ±6.24	49.56 ±5.81
✓	✓	×	✓	×	51.30 ±5.22	46.04 ±4.46

generalization gap by more than half. Notably, neither of the domain-randomization baselines (DR-PR or DR-CSDM) reached the performance of our method, suggesting that increasing scene diversity alone is insufficient to achieve the level of transfer provided by geometry-guided representation learning.

The most challenging setting (*Test Hard – Textured Floor*) introduces the textured floor and additional lighting and reflection changes. All methods degrade, but ours remains markedly more robust (49.4% SR), achieving more than 3× the performance of most pretrained models (i.e. DINOv2) and clearly outperforming SimCLR (45.7% SR). Importantly, textured floors are never observed during training, indicating that the learned representation captures navigation-relevant structure rather than superficial visual statistics.

3) Indoor-to-Outdoor Generalization: We next evaluate semantic domain shift by averaging results across indoor and outdoor environments (Table IV). Unlike the controlled appearance perturbations above, indoor-to-outdoor transfer introduces a semantic domain shift. Here, performance differences reflect the extent to which representations encode scene-agnostic navigation structure rather than environment-specific semantics. While pretrained baselines show substantial degradation across domains, our method maintains consistent performance across both indoor and outdoor settings (68.7% vs. 60.9% SR), with a significantly smaller generalization gap.

4) Ablation Study: Table V evaluates our framework components. We denote: "RL DR" (RL phase domain randomization), "Adaptive τ" (geometry-adaptive temperature, Section III-B3), "CSDM" (environment mismatch between pretraining and policy learning, Section III-D), "Scene Inv." (scene-invariant positive pairs), and "Vis. Augs." (standard SimCLR augmentations). The SimCLR baseline achieves 48.4% SR. Scene invariance alone improves over visual augmentations in isolation, but combining them degrades performance, suggesting standard SimCLR augmentations may be misaligned with geometry-oriented contrastive learning for navigation. CSDM consistently yields substantial gains across configurations. Comparing rows 3–4 and rows 6–7, CSDM improves performance by +9.7% SR and +3.0% SR, respectively, demonstrating that viewpoint diversity via cross-stage domain mismatch is robust to other design choices. This consistency underscores the fundamental importance of domain mismatch in learning robust representations. Adaptive temperature further improves to 63.6% SR and 58.6% SPL. Overall, robustness is driven primarily by domain mismatch and adaptive weighting, rather than conventional augmentations.



Fig. 5. Composite image from the real-world experiments showing the robot navigation. (a) outdoor, (b) indoor.

5) Discussion: Cross-Stage Domain Mismatch Vs. Domain Randomization: We investigated domain randomization as an alternative approach to robustness, testing it on both our final model and the SimCLR baseline (Table IV and V). Surprisingly, DR does not improve performance: on our model, DR reduces performance from 63.6% to 54.27% SR; on SimCLR, DR-PR underperforms the baseline (48.38% → 47.37% on average of all unseen scenes). This reveals a limitation: domain randomization's effectiveness depends on whether the randomization covers the actual domain shift. In our case, training environments contained uniform-colored floors, so the randomization lacked textured floor configurations. This mismatch caused overfitting to constant colors, degrading performance on textured floors—a shift larger than the randomization could encompass. In contrast, CSDM works through an implicit mechanism that does not require domain knowledge. The frozen encoder forces the policy to rely on geometry-consistent features rather than appearance-specific shortcuts. CSDM thus acts as an implicit regularizer promoting usefulness-based feature selection, explaining why it consistently improves performance (rows 3–4, 6–7, Table V) regardless of domain randomization.

D. Sim2Real Transfer

To evaluate sim-to-real transfer, the trained policy was deployed on a Limo Pro mobile robot equipped with an RGB camera and onboard computing. Due to computational limitations, policy inference was performed on an external laptop, and the generated actions were transmitted to the robot in real time.[3]

The pretrained policy demonstrated zero-shot transfer capability and was directly deployed on the physical robot without any real-world fine-tuning. We evaluated the final model in ten navigation trials across indoor and outdoor environments, as shown in Fig. 5. The policy achieved success rates of 60% indoors and 40% outdoors. Given the limited number of trials, these results should be interpreted as a proof-of-concept demonstration rather than a statistically exhaustive evaluation. Failures were primarily caused by collisions with previously unseen obstacles, such as a small square curb in the outdoor environment. Additionally, occasional backward motion was observed near the goal. We attribute this behavior to the reward

[3]The supplementary video is available at https://www.youtube.com/watch?v=ggAfytPQWQM.

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.

design in (6), where equal penalties for backward and angular motion can make reverse correction locally preferable to reorientation. Overall, the results demonstrate the feasibility of zero-shot sim-to-real transfer while highlighting the importance of obstacle diversity during training and reward coefficient tuning for robust real-world deployment.

VI. Conclusion

We presented a sensor-guided contrastive representation learning framework for robust scene transfer in vision-based PointGoal navigation. By leveraging privileged geometric sensing only during training to adaptively modulate the contrastive objective, our method learns navigation-relevant visual representations invariant to scene appearance. Decoupling representation learning from policy optimization further promotes reliance on task-relevant features.

Extensive experiments demonstrate substantial policy-level generalization across unseen indoor/outdoor environments under severe appearance shifts. At deployment, the agent uses only monocular RGB observations and standard inputs (goal position, proprioceptive signals), without privileged sensors. These results validate privileged-guided contrastive learning as an effective mechanism for improving robustness in vision-based navigation.

Future work will explore: (1) reducing reliance on explicit task inputs like goal position, (2) interaction between geometry-aware objectives and image augmentation strategies, and (3) which privileged sensors most effectively guide representation learning.

References

[1] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Müller, V. Koltun, and D. Scaramuzza, "Champion-level drone racing using deep reinforcement learning," *Nature*, vol. 620, no. 7976, pp. 982–987, Aug. 2023.

[2] C. Fifty, E. Amid, Z. Zhao, T. Yu, R. Anil, and C. Finn, "Efficiently identifying task groupings for multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 27503–27516.

[3] M. f. Shridhar, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Proc. Conf. Robot Learn.*, 2023, pp. 785–799.

[4] R. Rahmatizadeh, P. Abolghasemi, L. Bölöni, and S. Levine, "Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 3758–3765.

[5] A. Zeng et al., "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," *Int. J. Robot. Res.*, vol. 41, no. 7, pp. 690–705, 2022.

[6] P. R. Wurman et al., "Outracing champion Gran Turismo drivers with deep reinforcement learning," *Nature*, vol. 602, pp. 223–228, 2022.

[7] Y. Du, C. Gan, and P. Isola, "Curious representation learning for embodied intelligence," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10388–10397.

[8] L. Bonanni, D. Meli, A. Castellini, and A. Farinelli, "Monte Carlo tree search with velocity obstacles for safe and efficient motion planning in dynamic environments," in *Proc. 2025 Int. Conf. Auton. Agents Multiagent Syst.*, 2025, pp. 371–380.

[9] F. Taioli et al., "Collaborative instance object navigation: Leveraging uncertainty-awareness to minimize human-agent dialogues," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2025, pp. 18781–18792.

[10] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[11] F. Wang and H. Liu, "Understanding the behaviour of contrastive loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2495–2504.

[12] J. Xing, L. Bauersfeld, Y. Song, C. Xing, and D. Scaramuzza, "Contrastive learning for enhancing robust scene transfer in vision-based agile flight," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 5330–5337.

[13] J. Luo, C. Xu, J. Wu, and S. Levine, "Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning," *Sci. Robot.*, vol. 10, 2025, Art. no. eads5033.

[14] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res.*, vol. 17, pp. 1–40, 2016.

[15] A. Zhalehmehrabi, D. Meli, F. Dal Santo, F. Trotti, and A. Farinelli, "Depth-constrained ASV navigation with deep RL and limited sensing," *IEEE Robot. Automat. Lett.*, vol. 10, no. 12, pp. 13011–13017, Dec. 2025.

[16] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Motion planning and control for mobile robot navigation using machine learning: A survey," *Auton. Robots*, vol. 46, pp. 569–597, 2022.

[17] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," in *Proc. Conf. Robot Learn.*, 2023, pp. 416–426.

[18] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[19] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16000–16009.

[20] M. Shridhar, L. Manuelli, and D. Fox, "CLIPort: What and where pathways for robotic manipulation," in *Proc. Conf. Robot Learn.*, 2022, pp. 894–906.

[21] N. Messikommer, Y. Song, and D. Scaramuzza, "Contrastive initial state buffer for reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 2866–2872.

[22] S. Qiu, L. Wang, C. Bai, Z. Yang, and Z. Wang, "Contrastive UCB: Provably efficient contrastive self-supervised learning in online reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 18168–18210.

[23] B. Eysenbach, T. Zhang, R. Salakhutdinov, and S. Levine, "Contrastive learning as goal-conditioned reinforcement learning," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 35603–35620, 2022.

[24] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[25] F. Wang and H. Liu, "Understanding the behaviour of contrastive loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2495–2504.

[26] H. Zhuang, W. E. Zhang, J. Yang, W. Chen, and Q. Z. Sheng, "Not all negatives are equally negative: Soft contrastive learning for unsupervised sentence representations," in *Proc. 33 rd ACM Int. Conf. Inf. Knowl. Manage.*, 2024, pp. 3591–3601.

[27] R. Partsey, E. Wijmans, N. Yokoyama, O. Dobosevych, D. Batra, and O. Maksymets, "Is mapping necessary for realistic pointgoal navigation?," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17232–17241.

[28] B. Shen et al., "iGibson 1.0: A simulation environment for interactive tasks in large realistic scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 7520–7527.

[29] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1861–1870.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[31] M. Oquab et al., "DINOV2: Learning robust visual features without supervision," 2023, *arXiv:2304.07193*.

[32] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Sci.*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[33] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3519–3529.

[34] N. Frosst, N. Papernot, and G. Hinton, "Analyzing and improving representations with the soft nearest neighbor loss," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2012–2020.

[35] M. Caron et al., "Emerging properties in self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9650–9660.

[36] P. Sermanet et al., "Time-contrastive networks: Self-supervised learning from video," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1134–1141.

[37] N. Yokoyama, S. Ha, and D. Batra, "Success weighted by completion time: A dynamics-aware evaluation criteria for embodied navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 1562–1569.

Authorized licensed use limited to: Universita degli Studi di Verona. Downloaded on November 19,2026 at 17:49:48 UTC from IEEE Xplore. Restrictions apply.