GEAR: VERSATILE TRANSDUCTIVE DIFFUSION FOR TRAINING-FREE 3D OBJECT RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep 3D embeddings derived from pretrained models have demonstrated promising retrieval performance. However, achieving optimal results typically requires downstream adaptation with labeled 3D training data, which incurs significant annotation costs and limits their applicability in evolving real-world scenarios. In this paper, we introduce GEAR, a training-free test-time adaptation framework that efficiently performs transductive diffusion to improve the embedding discriminativeness without requiring labeled 3D data for adaptation. Specifically, GEAR only exploits the proximity information among unlabeled gallery objects and diffuses in a transductive manner. In this way, we enforce a cohesive embedding space for gallery samples suitable for retrieval. Unlike conventional transductive approaches that require a batch of queries for joint adaptation, GEAR enables single-query adaptation during inference by leveraging the refined gallery representations, making it applicable to standard online retrieval scenarios. Furthermore, we introduce an iterative anchored diffusion strategy to improve retrieval ranking while preserving the original query evidence. Extensive experiments on 3D object retrieval benchmarks demonstrate that GEAR effectively boosts the performance of various pretrained representations. Additional evaluations on sketch-based and scene-level retrieval further verify its flexibility and generalizability.

1 INTRODUCTION

We study 3D object retrieval (3DOR), which ranks a gallery of 3D assets according to their content similarity to a query. This task plays an essential role in a broad range of 3D applications, such as robotics, gaming, embodied AI, and e-commerce. The key to 3DOR is to encode diverse 3D data representations (*e.g.*, point clouds, voxels, and multi-view images) into a discriminative embedding space suitable for retrieval. Among existing methods, multi-view approaches Su et al. (2015); Kanezaki et al. (2018), which learn to encode multiple 2D projections of a 3D object captured from distinct viewpoints, have become the favored paradigm. Their key advantage is that rendered views can be processed by powerful 2D vision backbones such as CLIP Radford et al. (2021) and the DINO series Oquab et al. (2024); Siméoni et al. (2026), yielding generalizable embeddings with strong zero-shot performance. Nevertheless, the gap between rendered views and natural-image pretraining data can make these frozen embeddings suboptimal for a target gallery Zhang et al. (2024). Existing adaptation approaches typically require costly labeled 3D data He et al. (2025); Wang et al. (2025a); He et al. (2026) (Figure 1(a)), limiting their deployment in open-world and evolving scenarios.

Importantly, standard 3DOR is inherently transductive because gallery instances are fixed in advance. Their relational structure is therefore available at test time and can be exploited to handle distribution shifts without costly annotations. Yet, to our knowledge, this opportunity remains unexplored for test-time adaptation in 3DOR. Moreover, many retrieval-oriented transductive and test-time adaptation methods Shen et al. (2021); Feng et al. (2024); Wang et al. (2025b); Li et al. (2025) either require labeled training data or simultaneous access to multiple test queries, making them incompatible with online single-query retrieval. It thus remains unclear how to exploit gallery structure for independently arriving queries without supervision.

To this end, we introduce **G**allery-**E**nhanced **A**nchored **R**efinement (**GEAR**), a training-free framework that performs transductive refinement at test time. Unlike prior approaches, GEAR removes

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

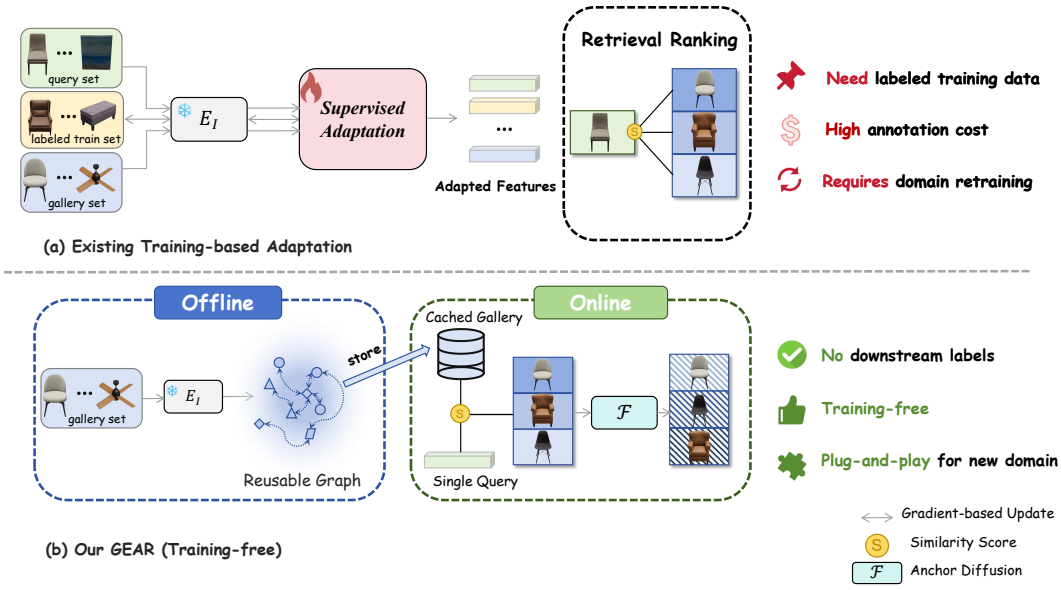

Figure 1: **Comparison between conventional adaptation paradigms and our GEAR.** (a) **Existing paradigms** require labeled 3D data for supervised adaptation or rely on batch-dependent joint optimization. (b) **Our GEAR** constructs a reusable graph from unlabeled gallery for offline diffusion, and calibrates each incoming query online at test time via anchored diffusion without training.

both costly annotation and batched-query requirements. It exploits intrinsic neighborhoods within the unlabeled gallery to refine gallery embeddings and caches the resulting graph offline (Figure 1(b)). Each incoming query is then calibrated independently against this reusable gallery prior, without information exchange between queries. We further introduce an iterative anchored diffusion scheme that propagates retrieval scores over the gallery graph while repeatedly preserving the original query evidence, thereby mitigating the over-smoothing of unconstrained graph diffusion. Comprehensive experiments on standard 3DOR benchmarks show that GEAR consistently improves diverse pretrained representations. Evaluations on sketch-based and scene-level retrieval further demonstrate its flexibility and generalization capability.

In summary, we make the following major contributions:

- We formulate adaptation in 3DOR as a transductive problem and introduce **GEAR**, a training-free framework that exploits unlabeled gallery neighborhoods to improve embedding discriminability.
- We design an asymmetric offline–online workflow that removes the batch-query dependency of conventional transductive methods and enables efficient single-query adaptation.
- We propose a simple but effective iterative anchored diffusion strategy to refine retrieval rankings online while preserving the original query evidence.

2 Related Work

2.1 Zero-Shot 3D Object Retrieval

Foundation models have enabled zero-shot 3D retrieval without closed-set supervision. Point-MAE Pang et al. (2022) and PointGPT Chen et al. (2023) pretrain point-cloud encoders with masked or autoregressive objectives, while ULIP Xue et al. (2023), OpenShape Liu et al. (2023), and Uni3D Zhou et al. (2024) align 3D representations with image–language spaces. A complementary approach aggregates multi-view features from frozen encoders such as CLIP Radford et al. (2021) and DINOv2 Oquab et al. (2024) into object descriptors without 3D annotations. Recent methods also mitigate projection-induced domain gaps Zhang et al. (2024). Training-based open-set 3DOR methods exploit part-level supervision through hypergraphs Xu et al. (2024). Yet, frozen descriptors

may be suboptimal for a downstream gallery, whereas supervised fine-tuning requires labeled 3D data. GEAR instead refines them using unlabeled gallery neighborhoods.

2.2 Test-Time Adaptation and Transductive Retrieval

Test-time adaptation (TTA) adapts pretrained models using unlabeled test data. Beyond entropy minimization with Tent Wang et al. (2021), vision–language methods reduce online optimization through cache- or augmentation-based designs Karmanov et al. (2024); Farina et al. (2024), while CLIPTTA Lafon et al. (2025) performs contrastive model updates. Recent 3D TTA focuses on point-cloud classification, understanding, and detection Mirza et al. (2023); Jiang et al. (2024); Chen et al. (2025b); TCR Li et al. (2025) addresses cross-modal query shift, while TeDA Wang et al. (2025b) applies gradient-based query–gallery adaptation to zero-shot 3D retrieval.

Transductive methods refine predictions or rankings using relations among unlabeled samples. RDP Bai et al. (2019) regularizes graph diffusion, whereas TransCLIP Zanella et al. (2024) uses joint test-set structure. Graph score propagation has also been explored for training-free 3D point-cloud OOD detection Chen et al. (2025a), but jointly models class prototypes and test samples. Existing TTA and transductive methods may require gradients, historical samples, or multiple queries, and can be sensitive to test-batch composition Zanella et al. (2025). Unconstrained diffusion can also weaken the original query evidence. GEAR instead caches the gallery structure offline, processes queries independently without gradients, and anchors diffusion to the initial scores.

3 Method

3.1 Problem Formulation

Let $\mathcal{G} = \{g_1, g_2, \ldots, g_N\}$ denote a gallery set consisting of N 3D objects. Given a test query object q, 3D object retrieval aims to rank the 3D objects in $\mathcal{G}$, such that objects that are semantically or geometrically similar to q receive smaller ranks. For accurate ranks, conventional methods typically rely on a labeled training set for downstream optimization. In contrast, we formulate this task as a transductive problem and proposed a fully training-free setting that bypasses these demands. In essence, we study the problem of adapting object embeddings at test-time when only the gallery set and the query are available. To this end, we only leverage representations from well-established pretrained models as the initial 3D embeddings, which have demonstrated great success He et al. (2026) in retrieval. Based on them, we aim to develop a 3D object retrieval framework applicable to many open-world scenarios, particularly those where the target domain lacks labeled training data.

3.2 Framework Overview

Following common multi-view-based methods Feng et al. (2024), each gallery object g_i is represented by M rendered views $\mathcal{V}_i = \{I_i^{(1)}, I_i^{(2)}, \ldots, I_i^{(M)}\}$. A frozen image encoder $E_I(\cdot)$ independently extracts their view-level features:

$$\mathbf{F}_i = E_I(\mathcal{V}_i) = [\mathbf{f}_{i,1}, \mathbf{f}_{i,2}, \ldots, \mathbf{f}_{i,M}], \tag{1}$$

where $\mathbf{f}_{i,m} \in \mathbb{R}^d$ denotes the feature of the m-th view and d is the embedding dimension.

The object representation is obtained by mean pooling, followed by a hyperbolic tangent activation and ℓ_2 normalization:

$$\mathbf{t}_i = \mathrm{Norm}\left(\tanh\left(\frac{1}{M}\sum_{m=1}^{M}\mathbf{f}_{i,m}\right)\right), \tag{2}$$

where $\mathrm{Norm}(\mathbf{x}) = \mathbf{x}/\|\mathbf{x}\|_2$. We denote the resulting gallery matrix by $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_N] \in \mathbb{R}^{d\times N}$. The query is processed identically to obtain $\mathbf{q} \in \mathbb{R}^d$, and the zero-shot baseline ranks the gallery using $\mathbf{s}_0 = \mathbf{q}^\top\mathbf{T}$. Complete descriptor construction details are provided in Appendix A.2.

Modern pretrained vision encoders such as CLIP Radford et al. (2021) and DINOv2 Oquab et al. (2024) provide strong generic representations, yet their direct application to downstream 3D retrieval remains affected by query–gallery distribution discrepancies. To improve the resulting ranking, we introduce GEAR, which consists of two decoupled stages.

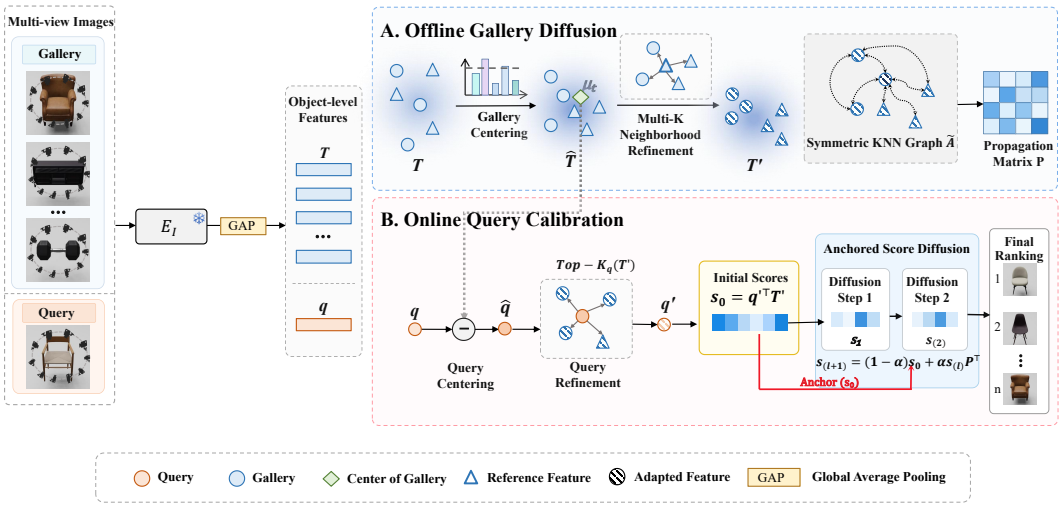

Figure 2: **Overview of GEAR.** GEAR follows an asymmetric offline–online pipeline. **(A) Offline Gallery Diffusion.** Frozen multi-view descriptors are centered and refined across multi-scale gallery neighborhoods, yielding cached gallery embeddings and a reusable transition graph. **(B) Online Query Calibration.** Each incoming query is independently adapted against the refined gallery, followed by anchored score diffusion that preserves its initial evidence.

1) Offline Gallery Diffusion. GEAR exploits structural relationships among unlabeled gallery objects to refine $\mathbf{T}$ into a more cohesive representation $\mathbf{T}'$ and constructs a reusable transition matrix P (Section 3.3).

2) Online Query Calibration. For each independently arriving query, GEAR performs single-query adaptation using the refined gallery to obtain $\mathbf{q}'$ (Section 3.4). The resulting query–gallery scores are then further refined through iterative anchored score diffusion (Section 3.4.2).

In the following, we elaborate on each stage.

3.3 Offline Gallery Diffusion

Large-scale pretrained backbones yield general but less discriminative 3D embeddings. To enhance them, we introduce a test-time training-free method to enhance intra-class compactness and correcting noisy outlier embeddings through neighborhood interactions among gallery embeddings.

3.3.1 Gallery Centering

To mitigate dataset-level shared biases (e.g., universal rendering backgrounds) and establish a relative coordinate system, we first center the representations. We first calculate the statistical mean of the gallery embedding matrix:

$$\boldsymbol{\mu}_t = \frac{1}{N}\sum_{i=1}^{N}\mathbf{t}_i, \tag{3}$$

and use it to center the gallery representations via subtraction, followed by ℓ_2 normalization (Norm):

$$\hat{\mathbf{t}}_i = \mathrm{Norm}\left(\mathbf{t}_i - \boldsymbol{\mu}_t\right). \tag{4}$$

Symmetrically, the query is centered using the same gallery statistics:

$$\hat{\mathbf{q}} = \mathrm{Norm}\left(\mathbf{q} - \boldsymbol{\mu}_t\right). \tag{5}$$

The above process projects the embeddings onto a hypersphere centered at the gallery mean, effectively improve embedding discriminativeness for retrieval.

3.3.2 MULTI-K NEIGHBORHOOD REFINEMENT

Following centering, we refine the gallery representations based on local similarity graphs. Formally, for a specific scale K at iteration $r \in \{0, 1\}$, we identify the K-nearest neighbors $\mathcal{N}_K^{(r)}(i)$ for the i-th gallery sample, excluding itself. Let $\hat{\mathbf{t}}_i^{(r)}$ denote the centered gallery embedding of the (i-th gallery sample at iteration r, where $r = 0$ corresponds to the initial embeddings $\hat{\mathbf{t}}_i$. The similarity between samples i and j: $s_{ij}^{(r)} = \left(\hat{\mathbf{t}}_i^{(r)}\right)^\top \hat{\mathbf{t}}_j^{(r)}$, are normalized within the neighborhood context:

$$w_{ij}^{(r)} = \frac{\exp\left(s_{ij}^{(r)}/\tau\right)}{\sum_{k\in\mathcal{N}_K^{(r)}(i)} \exp\left(s_{ik}^{(r)}/\tau\right)}, \qquad j \in \mathcal{N}_K^{(r)}(i), \tag{6}$$

where τ is the temperature.

The corresponding local context feature is computed as

$$\mathbf{z}_i^{(r)} = \sum_{j\in\mathcal{N}_K^{(r)}(i)} w_{ij}^{(r)} \hat{\mathbf{t}}_j^{(r)}. \tag{7}$$

To prevent the original features from being completely overridden by neighborhood contexts, we define an upper bound $\lambda_{\max} = 0.7$. The local reliability coefficient $\lambda_i^{(r)}$ is formulated as

$$\lambda_i^{(r)} = \lambda_{\max} \cdot \mathrm{clip}\left(\frac{1}{K} \sum_{j\in\mathcal{N}_K^{(r)}(i)} s_{ij}^{(r)}, 0, 1\right). \tag{8}$$

The gallery features are then updated through a residual formulation:

$$\hat{\mathbf{t}}_i^{(r+1)} = \mathrm{Norm}\left(\left(1 - \lambda_i^{(r)}\right)\hat{\mathbf{t}}_i^{(r)} + \lambda_i^{(r)}\mathbf{z}_i^{(r)}\right). \tag{9}$$

We perform exactly two rounds of updates ($r = 0, 1$). Since each update aggregates information from immediate 1-hop neighbors, two rounds allow each sample to indirectly absorb structural semantics from its second-order neighborhood. Finally, Four context scales (*i.e.*, $K \in \{5, 10, 20, 50\}$) are average pooled and ℓ_2-normalized as final gallery representation $\mathbf{t}'_i \in \mathbb{R}^d$.

3.4 ONLINE QUERY CALIBRATION

3.4.1 SINGLE QUERY EMBEDDING RECALIBRATION

We use the improved gallery embeddings to recalibrate the query embeddings in an on-line manner. Specifically, we calculate similarity between the query embedding $\hat{\mathbf{q}}$ and each refined gallery embedding $\mathbf{t}'_j$ as

$$u_j = \hat{\mathbf{q}}^\top \mathbf{t}'_j, \tag{10}$$

and select the top-K_q contextual neighbors $\mathcal{N}_{K_q}(q)$ from the gallery set, where $K_q = 10$. The contextual weights are normalized similarly as

$$w_j^q = \frac{\exp(u_j/\tau)}{\sum_{k\in\mathcal{N}_{K_q}(q)} \exp(u_k/\tau)}, \qquad j \in \mathcal{N}_{K_q}(q). \tag{11}$$

The context embedding for the query is then obtained by

$$\mathbf{z}'_q = \sum_{j\in\mathcal{N}_{K_q}(q)} w_j^q \mathbf{t}'_j. \tag{12}$$

Symmetrically to the gallery update, we estimate the query reliability coefficient λ_q based on the average similarity of these K_q neighbors:

$$\lambda_q = \lambda_{\max} \cdot \mathrm{clip}\left(\frac{1}{K_q} \sum_{j\in\mathcal{N}_{K_q}(q)} u_j, 0, 1\right). \tag{13}$$

The adapted query representation is then obtained as

$$\mathbf{q}' = \mathrm{Norm}\left((1-\lambda_q)\hat{\mathbf{q}} + \lambda_q \mathbf{z}'_q\right). \tag{14}$$

This dynamic injection effectively aligns the standalone query with the most probable gallery semantic cluster. The initial pairwise retrieval score is subsequently computed as

$$\mathbf{s}_0 = \mathbf{q}'^{\top}\mathbf{T}', \tag{15}$$

where $\mathbf{T}' = [\mathbf{t}'_1, \mathbf{t}'_2, \ldots, \mathbf{t}'_N] \in \mathbb{R}^{d\times N}$ denotes the embedding matrix of gallery set.

3.4.2 Query-Gallery Score Calibration

Anchored Score Diffusion. The pairwise score s_0 fails to capture higher-order structural consistency among gallery samples. To propagate query-conditioned retrieval scores across structurally related gallery samples, we construct a sparse K_g-NN graph, where $K_g = 20$, based on T' with adjacency matrix A.

To ensure reciprocal neighborhood relationships and eliminate the bias introduced by asymmetric directed edges, we symmetrize the graph as:

$$\widetilde{A} = (A + A^{\top})/2.$$

We then apply row normalization to obtain the transition probability matrix $P \in \mathbb{R}^{N\times N}$. Based on it, a straightforward graph diffusion Bai et al. (2019) can be employed:

$$s_{(l+1)} = s_{(l)}P^{\top}, \qquad l = 0, 1, \ldots, L-1, \tag{16}$$

where $s_{(l)} \in \mathbb{R}^{1\times N}$ denotes the diffused score vector at the l-th diffusion step, with $s_{(0)} = s_0$, and L is the diffusion step number. However, the above vanilla diffusion scheme progressively attenuates the original query evidence and ultimately converges to a query-agnostic stationary distribution. Motivated by restart-based personalized propagation Klicpera et al. (2019), we introduce its score-level counterpart for single-query retrieval:

$$s_{(l+1)} = (1-\alpha)s_0 + \alpha s_{(l)}P^{\top}. \tag{17}$$

where the initial query preference s_0 acts as a persistent anchor that is reinjected at every iteration. Empirically, two diffusion rounds suffice to attain strong performance at high efficiency.

3.5 Computational Complexity

Let N and d denote the gallery size and feature dimension, S the number of gallery scales, L_g the number of gallery-update rounds, K_q and K_g the query and graph neighborhood sizes, and L the number of diffusion steps. All gallery-dependent computations are performed once offline, requiring $\mathcal{O}((SL_g+1)N^2d)$ time with exact dense-similarity computation. The resulting $\boldsymbol{\mu}_t$, $\mathbf{T}'$, and sparse transition matrix P are cached for reuse. For each incoming query, GEAR requires $\mathcal{O}(Nd + K_qd + LNK_g + N\log N)$ time and $\mathcal{O}(N)$ transient memory, including exact ranking. Since K_q, K_g, and L are fixed, the online cost is near-linear in the gallery size. Unlike batch-transductive methods that jointly model multiple queries Zanella et al. (2024), GEAR introduces no query–query interaction term. The complete derivation is provided in Appendix E.4.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct experiments on four standard open-set 3DOR benchmarks (OS-ESB-core, OS-NTU-core, OS-ModelNet40-core, and OS-ABO-core) following Feng et al. (2024). For evaluation efficiency, frozen query descriptors are pre-extracted, while each query is adapted independently without query–query interaction. We further evaluate generalization on cross-modal retrieval (SHREC2013 Li et al. (2013), Sketchy Ext Liu et al. (2017), and TU-Berlin Ext Liu et al. (2017)) and scene-level retrieval (SUN397 Xiao et al. (2010)).

Table 1: **Comparison with state-of-the-art methods on four open-set 3DOR benchmarks.** ‡MV-CLIP denotes multi-view CLIP feature averaging with $\tanh$ saturation and ℓ_2 normalization. TeDA and GEAR are evaluated on the same backbone-matched frozen multi-view descriptors for controlled comparison. **Bold** and underlined numbers indicate the best and second-best.

Method	Backbone	OS-ESB-core			OS-NTU-core			OS-MN40-core			OS-ABO-core		
		mAP↑	NDCG↑	ANMRR↓	mAP↑	NDCG↑	ANMRR↓	mAP↑	NDCG↑	ANMRR↓	mAP↑	NDCG↑	ANMRR↓
Training-required													
SDML (SIGIR'19) Hu et al. (2019)	ResNet18, PointNet, 3DShapenets	49.59	21.75	52.36	40.16	21.52	60.49	50.75	65.70	50.22	47.44	52.79	52.42
CMCL (CVPR'21) Jing et al. (2021)		50.01	21.97	53.06	41.08	21.72	59.43	51.38	65.98	49.75	49.83	50.89	50.24
PROSER (CVPR'21) Zhou et al. (2021)		48.69	21.13	53.95	39.47	21.24	60.96	49.00	64.54	51.66	50.33	53.27	50.34
MCWSA (TMM'22) Zheng et al. (2023)		49.48	21.34	53.75	39.22	20.69	62.14	48.78	63.85	51.95	45.61	51.05	54.70
HGM^{2}R (TPAMI'24) Feng et al. (2024)		51.74	22.73	51.28	44.88	22.81	56.67	64.20	72.91	38.27	63.39	57.96	37.96
CLIP-AdaM (SIGIR'25) He et al. (2025)	CLIP ViT-L/14	59.90	24.69	44.80	61.52	27.65	41.72	63.54	74.21	38.76	64.77	58.75	37.25
DEC (CVPR'26) He et al. (2026)	DINOv2 ViT-B/14	61.82	24.55	42.74	61.56	27.27	41.62	67.62	77.67	34.95	65.04	59.34	37.32
Training-free													
ULIP (CVPR'23) Xue et al. (2023)	PointMLP-SLIP	33.81	16.71	68.18	20.76	13.46	78.26	24.56	38.23	72.41	34.00	39.79	64.94
OpenShape (NeurIPS'23) Liu et al. (2023)	PointBERT-CLIP ViT-L/14	38.58	18.81	65.10	24.71	15.02	75.18	29.64	44.79	67.64	38.65	45.56	60.90
Uni3D (ICLR'24) Zhou et al. (2024)	Uni3D-Giant	44.42	20.92	59.96	32.02	18.04	68.49	33.21	50.51	65.11	45.92	49.79	53.96
ULIP-2 (CVPR'24) Xue et al. (2024)	PointBERT-CLIP ViT-G/14	45.14	21.00	59.15	31.50	17.89	68.99	32.76	48.92	65.22	44.26	49.04	55.64
MV-CLIP‡ Song et al. (2025)	CLIP ViT-L/14	49.81	22.75	53.71	57.71	26.46	45.25	63.74	74.85	37.80	63.07	59.18	38.67
MV-CLIP‡ Song et al. (2025)	OpenCLIP ViT-L/14	58.11	24.29	44.49	57.23	26.25	45.00	60.66	72.76	40.68	59.74	57.11	41.88
TeDA (ICMR'25) Wang et al. (2025b)	CLIP ViT-L/14	60.92	25.13	43.99	61.63	27.44	41.77	63.94	73.83	38.27	65.59	57.80	37.02
GEAR (Ours)	CLIP ViT-L/14	63.14	**25.32**	42.24	64.32	27.53	38.90	66.61	74.41	35.60	68.55	57.61	33.48
TeDA (ICMR'25) Wang et al. (2025b)	DINOv2 ViT-B/14	61.76	25.22	43.42	62.91	27.69	40.66	70.12	77.25	33.42	67.83	59.14	35.58
GEAR (Ours)	DINOv2 ViT-B/14	**63.31**	25.12	**41.51**	65.44	28.00	**37.89**	74.40	80.12	**28.82**	**70.99**	**59.27**	**32.31**
TeDA (ICMR'25) Wang et al. (2025b)	DINOv3 ViT-B/16	57.33	23.97	48.07	60.07	27.15	43.44	68.14	76.77	35.00	66.38	57.62	36.79
GEAR (Ours)	DINOv3 ViT-B/16	62.39	24.51	43.16	**65.72**	**28.01**	38.40	**74.48**	**80.71**	29.25	69.25	58.25	33.96

Implementation Details. We adopt several representative pretrained models, including CLIP, DINOv2, and DINOv3, to extract the initial embeddings. For fair comparisons, all methods are evaluated using identical frozen backbone features without any downstream training or parameter optimization. During inference, GEAR exclusively exploits unlabeled gallery samples to refine gallery embeddings and construct a reusable transition graph, while adapting each query independently without accessing any category labels or annotations. Across all experiments, GEAR adopts a *unified* parameter configuration. Specifically, the gallery neighborhood size and query neighborhood size are set to $K_g = 20$ and $K_q = 10$, respectively. We set temperature τ to 0.1. The remaining hyperparameters are *fixed* across different datasets unless otherwise specified.

4.2 Comparison with State-of-the-Arts

Table 1 compares GEAR with representative state-of-the-art 3DOR methods, including both training-required and training-free approaches. As shown, GEAR consistently achieves the strongest training-free performance across all three backbones. With DINOv2 ViT-B/14, it attains the best ANMRR on all benchmarks and the best mAP and NDCG@100 on OS-ABO-core; with DINOv3 ViT-B/16, it further achieves the best mAP and NDCG@100 on OS-NTU-core and OS-MN40-core.

Notably, despite being training-free, GEAR outperforms recent training-required baselines, including CLIP-AdaM He et al. (2025) and DEC He et al. (2026). Compared with DEC using the same DINOv2 backbone, GEAR improves mAP by 1.49–6.78 percentage points and reduces ANMRR by 1.23–6.13 points across the four benchmarks. This indicates that offline diffusion over unlabeled gallery neighborhoods can recover, and even exceed, the discriminability typically acquired through supervised adaptation.

Furthermore, unlike the strongest transductive competitor TeDA, which strictly requires the *entire* query set for gradient optimization, our GEAR operates efficiently in a single-query mode, yet consistently surpasses TeDA by a measurable margin. The robust gains across CLIP, DINOv2, and DINOv3 demonstrate that GEAR can serve as a universal, plug-and-play refinement module for diverse pretrained 3D representations. Note that the improvements are especially pronounced in ANMRR, suggesting that our anchored diffusion not only retrieves more relevant objects but also effectively suppresses highly irrelevant neighbors. We attribute these achievements to two core design choices: shifting structural modeling to a cacheable unlabeled gallery prior, and strictly preserving the original query evidence via iterative anchored diffusion.

Table 2: **Generalization to point clouds and more retrieval tasks.** All entries are mAP; gains are absolute percentage points.

Task	Benchmark	Representation / baseline	Baseline	GEAR	Gain
Point cloud	ModelNet40	OpenShape-PointBERT	71.98	**80.49**	+8.51
	ModelNet40	Point-MAE	50.71	**56.99**	+6.28
Cross-modal	SHREC2013 zero-shot	MEHA-ZS Zhang et al. (2025)	35.30	**37.80**	+2.50
	SHREC2013 category	MEHA Zhang et al. (2025)	90.30	**90.70**	+0.40
	Sketchy Ext	Sketch_LVM Sain et al. (2023)	62.31	**72.32**	+10.01
	TU-Berlin Ext	Sketch_LVM Sain et al. (2023)	52.09	**69.57**	+17.48
Scene	SUN397	Places365-ResNet50 Zhou et al. (2018)	37.33	**47.01**	+9.68
	SUN397	CLIP ViT-L/14	38.41	**52.62**	+14.21
	SUN397	DINOv2 ViT-B/14	45.54	**51.78**	+6.24

Table 3: **Ablation study of the core components in GEAR.** Results are averaged over four open-set 3DOR benchmarks. All metrics are reported as percentages.

Gal. C.	Multi-K	Query A.	Anch. D.	mAP↑	NDCG@100↑	ANMRR↓
✗	✗	✗	✗	60.98	45.98	41.89
✓	✗	✗	✗	63.83	46.89	39.95
✓	✓	✗	✗	66.66	47.62	37.12
✓	✓	✓	✗	68.19	48.11	35.52
✓	✓	✓	✓	**68.53**	**48.13**	**35.13**

4.3 Generalization Analysis

We further evaluate GEAR beyond multi-view 3D descriptors across point-cloud, cross-modal, and scene-level retrieval, as summarized in Table 2.

Point-cloud representations. On ModelNet40 Wu et al. (2015), GEAR consistently improves both the self-supervised Point-MAE Pang et al. (2022) and language-aligned OpenShape Liu et al. (2023) representations.

Cross-modal retrieval. GEAR consistently improves both sketch-based shape retrieval and sketch-based image retrieval. In particular, it yields gains of 10.01 and 17.48 percentage points with Sketch_LVM Sain et al. (2023) on Sketchy Ext and TU-Berlin Ext, respectively.

Scene-level retrieval. On SUN397, GEAR improves all three evaluated representations, with the largest gain of 14.21 percentage points obtained using CLIP. Overall, these results demonstrate that GEAR generalizes across representation families, query modalities, and retrieval granularities. Complete task-specific protocols and metrics are provided in Appendix G.2.

4.4 Ablation and Analytic Study

We conduct ablation and analytic studies using DINOv2 and report macro-averaged results over the four open-set benchmarks unless otherwise specified.

Effect of Core Components. Table 3 shows consistent gains as components are progressively introduced. Gallery centering and Multi-K refinement provide the largest mAP gains of 2.85 and 2.83 percentage points, respectively, while query adaptation further contributes 1.53 points. Anchored diffusion yields the best overall trade-off, reaching 68.53% mAP and 35.13% ANMRR. These results verify the complementarity of gallery refinement, query calibration, and score propagation.

Effect of Multi-Scale Design. We further investigate the influence of different gallery neighborhood sizes. As detailed in Appendix D.2, retrieval performance varies with the choice of K, suggesting that a fixed neighborhood scale may not generalize uniformly across retrieval settings. The proposed Multi-K strategy achieves performance comparable to the best single-scale configuration while obtaining the best NDCG@100, without requiring dataset-specific selection of K.

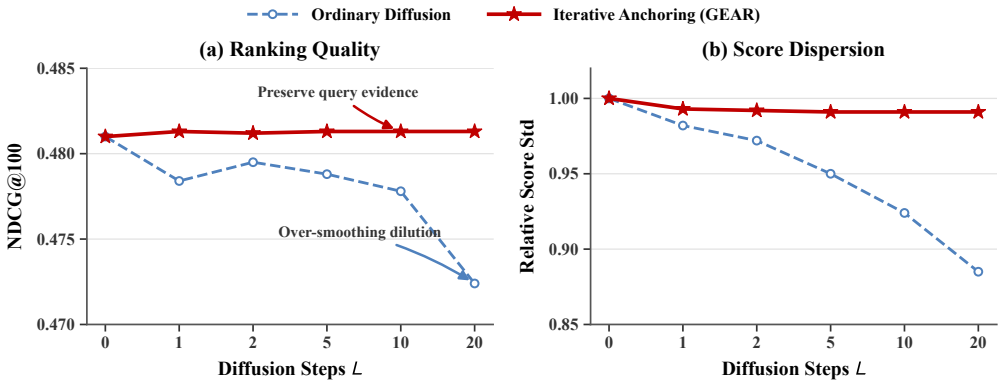

Figure 3: **Analysis of the anchored diffusion strategy.** NDCG@100 and relative score standard deviation for ordinary diffusion and iterative anchoring under different propagation steps L.

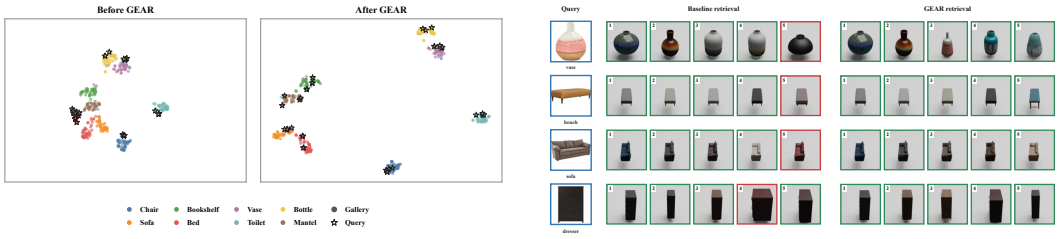

Figure 4: **Visual analysis.** (a) t-SNE van der Maaten & Hinton (2008) of frozen CLIP embeddings before and after GEAR on ModelNet40; circles and stars denote gallery and query samples. (b) DINOv2 top-5 retrieval on OS-ABO-core; green and red borders denote correct and incorrect matches.

Analysis of Anchored Diffusion. We compare iterative anchored diffusion with ordinary graph diffusion under different propagation depths L. As illustrated in Figure 3, ordinary diffusion becomes increasingly unstable as propagation proceeds, with both ranking quality and score dispersion gradually deteriorating. In contrast, iterative anchoring maintains substantially more stable behavior by repeatedly preserving the original query evidence during propagation.

Efficiency. With gallery-dependent representations and the sparse transition graph cached offline, GEAR requires only 2.76 ms per query on average, including ranking. The complete runtime breakdown is reported in Appendix E.3.

Embedding-Space Analysis. We further evaluate the refined gallery structure using neighborhood purity and the Fisher ratio, with class labels used solely for post-hoc evaluation. After gallery refinement, Purity@10 increases from 82.17% to 83.11%, while the Fisher ratio increases from 0.84 to 1.42, indicating improved local consistency and global separability.

Visual Analysis. Figure 4(a) shows that GEAR produces more compact and separated clusters while better aligning queries with relevant gallery neighborhoods. Consistently, the top-5 examples in Figure 4(b) exhibit fewer retrieval errors than the frozen baseline.

5 Conclusion

We have introduced GEAR, a training-free framework that transductively refines pretrained representations using the intrinsic structure of unlabeled galleries at test time. GEAR's decoupled offline-online design and anchored diffusion enable efficient single-query adaptation without gradient optimization or batch-query dependencies. Extensive experiments demonstrate consistent improvements across 3D, sketch-based, and scene-level retrieval tasks, validating the generalization of GEAR across diverse retrieval scenarios. Future work will extend this paradigm to dynamically evolving galleries and broader cross-modal scenarios.

AI USE STATEMENT

Generative AI tools were used to assist with experimental analysis and interpretation, manuscript drafting and language editing. All AI-assisted content was reviewed and verified by the authors against the original experimental results and source literature. The authors take full responsibility for the final content, claims, and artifacts presented in this work.

REPRODUCIBILITY STATEMENT

We detail GEAR and its experimental setup in the main paper. The supplementary materials provide additional implementation details, ablations, analyses, and anonymized code. The complete research code and evaluation scripts will be released upon publication.

REFERENCES

Song Bai, Xiang Bai, Qi Tian, and Longin Jan Latecki. Regularized diffusion process on bidirectional context for object retrieval. *IEEE TPAMI*, 41(5):1213–1226, 2019.

Guangyan Chen, Meiling Wang, Yi Yang, Kai Yu, Li Yuan, and Yufeng Yue. PointGPT: Auto-regressively generative pre-training from point clouds. In *NeurIPS*, volume 36, pp. 29667–29679, 2023.

Tiankai Chen, Yushu Li, Adam Goodge, Fei Teng, Xulei Yang, Tianrui Li, and Xun Xu. Exploiting vision language model for training-free 3D point cloud OOD detection via graph score propagation. In *ICCV*, pp. 28797–28807, 2025a.

Zhuoxiao Chen, Junjie Meng, Mahsa Baktashmotlagh, Yonggang Zhang, Zi Huang, and Yadan Luo. MOS: Model synergy for test-time adaptation on LiDAR-based 3D object detection. In *ICLR*, volume 2025, pp. 92784–92808, 2025b.

Matteo Farina, Gianni Franchi, Giovanni Iacca, Massimiliano Mancini, and Elisa Ricci. Frustratingly easy test-time adaptation of vision-language models. In *NeurIPS*, volume 37, pp. 129062–129093, 2024.

Yifan Feng, Shuyi Ji, Yu-Shen Liu, Shaoyi Du, Qionghai Dai, and Yue Gao. Hypergraph-based multi-modal representation for open-set 3D object retrieval. *IEEE TPAMI*, 46(4):2206–2223, 2024.

Xinwei He, Liang Ma, Yuxuan Cheng, Zhichuan Wang, Yulong Wang, Yang Zhou, and Xiang Bai. CLIP-AdaM: Adapting multi-view CLIP for open-set 3D object retrieval. In *ACM SIGIR*, pp. 1022–1032, 2025.

Xinwei He, Yansong Zheng, Qianru Han, Zhichuan Wang, Yuxuan Cai, Yang Zhou, Jingbo Xia, Yulong Wang, Jinhai Xiang, and Xiang Bai. DINO eats CLIP: Adapting beyond knowns for open-set 3D object retrieval. In *CVPR*, pp. 34704–34713, 2026.

Peng Hu, Liangli Zhen, Dezhong Peng, and Pei Liu. Scalable deep multimodal learning for cross-modal retrieval. In *ACM SIGIR*, pp. 635–644, 2019.

Jincen Jiang, Qianyu Zhou, Yuhang Li, Xinkui Zhao, Meili Wang, Lizhuang Ma, Jian Chang, Jian Jun Zhang, and Xuequan Lu. PCoTTA: Continual test-time adaptation for multi-task point cloud understanding. In *NeurIPS*, volume 37, pp. 96229–96253, 2024.

Longlong Jing, Elahe Vahdani, Jiaxing Tan, and Yingli Tian. Cross-modal center loss for 3D cross-modal retrieval. In *CVPR*, pp. 3142–3151, 2021.

Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. RotationNet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In *CVPR*, pp. 5010–5019, 2018.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *CVPR*, pp. 14162–14171, 2024.

Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized PageRank. In *ICLR*, 2019.

Marc Lafon, Gustavo Vargas Hakim, Clément Rambour, Christian Desrosiers, and Nicolas Thome. CLIPTTA: Robust contrastive vision-language test-time adaptation. In *NeurIPS*, volume 38, pp. 79939–79973, 2025.

Bo Li, Yijuan Lu, Afzal Godil, Tobias Schreck, Masaki Aono, Henry Johan, Jose M. Saavedra, and Shoki Tashiro. SHREC'13 track: Large scale sketch-based 3D shape retrieval. In *Eurographics Workshop on 3D Object Retrieval*, pp. 89–96, 2013.

Haobin Li, Peng Hu, Qianjun Zhang, Xi Peng, Xiting Liu, and Mouxing Yang. Test-time adaptation for cross-modal retrieval with query shift. In *ICLR*, volume 2025, pp. 76517–76544, 2025.

Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *CVPR*, pp. 2862–2871, 2017.

Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3D shape representation towards open-world understanding. In *NeurIPS*, volume 36, pp. 44860–44879, 2023.

M. Jehanzeb Mirza, Inkyu Shin, Wei Lin, Andreas Schriebl, Kunyang Sun, Jaesung Choe, Mateusz Kozinski, Horst Possegger, In So Kweon, Kuk-Jin Yoon, and Horst Bischof. MATE: Masked autoencoders are online 3D test-time learners. In *ICCV*, pp. 16709–16718, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024.

Yatian Pang, Wenxiao Wang, Francis E. H. Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *ECCV*, pp. 604–621, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, volume 139, pp. 8748–8763, 2021.

Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. Hubs in space: Popular nearest neighbors in high-dimensional data. *JMLR*, 11:2487–2531, 2010.

Aneeshan Sain, Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Subhadeep Koley, Tao Xiang, and Yi-Zhe Song. CLIP for all things zero-shot sketch-based image retrieval, fine-grained or not. In *CVPR*, pp. 2765–2775, 2023.

Xi Shen, Yang Xiao, Shell Xu Hu, Othman Sbai, and Mathieu Aubry. Re-ranking for image retrieval and transductive few-shot classification. *NeurIPS*, 34:25932–25943, 2021.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seung Eun Yi, Michael Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Herve Jegou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *TMLR*, 2026.

Dan Song, Xinwei Fu, Ning Liu, Wei-Zhi Nie, Wen-Hui Li, Lan-Jun Wang, You Yang, and An-An Liu. MV-CLIP: Multi-view CLIP for zero-shot 3D shape recognition. *IEEE TCSVT*, 35(9): 8767–8779, 2025.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3D shape recognition. In *ICCV*, pp. 945–953, 2015.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *JMLR*, 9:2579–2605, 2008.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021.

Zhichuan Wang, Yang Zhou, Zhe Liu, Rui Yu, Song Bai, Yulong Wang, Xinwei He, and Xiang Bai. Describe, adapt and combine: Empowering CLIP encoders for open-set 3D object retrieval. In *ICCV*, pp. 21026–21036, 2025a.

Zhichuan Wang, Yang Zhou, Jinhai Xiang, Yulong Wang, and Xinwei He. TeDA: Boosting vision-language models for zero-shot 3D object retrieval via testing-time distribution alignment. In *ACM ICMR*, pp. 1442–1451, 2025b.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3D ShapeNets: A deep representation for volumetric shapes. In *CVPR*, pp. 1912–1920, 2015.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pp. 3485–3492, 2010.

Yang Xu, Yifan Feng, Jun Zhang, Jun-Hai Yong, and Yue Gao. Assembly fuzzy representation on hypergraph for open-set 3D object retrieval. In *NeurIPS*, volume 37, pp. 125098–125115, 2024.

Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. ULIP: Learning a unified representation of language, images, and point clouds for 3D understanding. In *CVPR*, pp. 1179–1189, 2023.

Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. ULIP-2: Towards scalable multimodal pre-training for 3D understanding. In *CVPR*, pp. 27091–27101, 2024.

Maxime Zanella, Benoît Gérin, and Ismail Ben Ayed. Boosting vision-language models with transduction. In *NeurIPS*, volume 37, pp. 62223–62256, 2024.

Maxime Zanella, Clément Fuchs, Christophe De Vleeschouwer, and Ismail Ben Ayed. Realistic test-time adaptation of vision-language models. In *CVPR*, pp. 25103–25112, 2025.

Donglin Zhang, Changxing Li, and Xiao-Jun Wu. Multi-level encoding with hierarchical alignment for sketch-based 3D shape retrieval. In *ACM SIGIR*, pp. 1033–1043, 2025.

Zhihao Zhang, Shengcao Cao, and Yu-Xiong Wang. TAMM: TriAdapter multi-modal learning for 3D shape understanding. In *CVPR*, pp. 21413–21423, 2024.

Jiahao Zheng, Sen Zhang, Zilu Wang, Xiaoping Wang, and Zhigang Zeng. Multi-channel weight-sharing autoencoder based on cascade multi-head attention for multimodal emotion recognition. *IEEE TMM*, 25:2213–2225, 2023.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 40(6):1452–1464, 2018.

Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning placeholders for open-set recognition. In *CVPR*, pp. 4401–4410, 2021.

Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3D: Exploring unified 3D representation at scale. In *ICLR*, volume 2024, pp. 46766–46782, 2024.

APPENDIX

CONTENTS

A Supplementary Experimental Protocol **15**

A.1 Open-Set Benchmarks and Evaluation Splits 15

A.2 Frozen Descriptor Construction 15

A.3 Unified GEAR Configuration 15

A.4 Metrics and Statistical Analysis 15

A.5 Execution Environment 16

B Additional Implementation Details **16**

B.1 Numerical and Data Conventions 16

B.2 Multi-Scale Refinement and Graph Construction 16

B.3 Offline and Online Execution 16

C Additional Quantitative Results **16**

C.1 Improvement over Frozen Descriptors 16

D Detailed Ablation Studies **17**

D.1 Disentangling the Core Components 17

D.2 Gallery Refinement Design 17

D.3 Diffusion Strategy and Parameter Sensitivity 17

D.4 Neighborhood Sensitivity 17

D.5 Paired Statistical Analysis 18

E Single-Query Validation and Efficiency **18**

E.1 Sensitivity to Query Batch Size 18

E.2 Order Invariance and Execution Equivalence 19

E.3 Offline Preparation and Online Latency 19

E.4 Computational Complexity 19

F Post-Hoc Gallery Structure Analysis **20**

F.1 Diagnostic Definitions 20

F.2 Macro-Average Structural Changes 21

F.3 Per-Dataset Structural Results 21

G Backbone and Scenario Generalization **22**

G.1 Component Transfer across Visual Backbones 22

G.2 Transfer beyond Multi-View Object Retrieval 22

H Additional Sensitivity and Limitations **22**

H.1 Softmax Temperature 22
H.2 Propagation Depth across Backbones 23
H.3 Scope and Limitations 23

I Qualitative Protocol and Case Analysis **24**
I.1 Embedding-Space Visualization 24
I.2 Retrieval-Case Selection and Exact Rankings 24

Table 4: **Statistics of the open-set 3D retrieval benchmarks.** The listed query and gallery sets are used in all controlled backbone and ablation experiments.

Dataset	Queries	Gallery objects	Views per object
OS-ESB-core	120	452	24
OS-NTU-core	270	1,271	24
OS-MN40-core	160	9,329	24
OS-ABO-core	85	5,455	4

Table 5: **Unified hyperparameter configuration of GEAR.**

Component	Setting
Gallery neighborhood scales	$\{5, 10, 20, 50\}$
Gallery self-update rounds	$L_g = 2$ per scale
Multi-scale fusion	Uniform feature average
Query neighborhood size	$K_q = 10$
Gallery graph neighborhood size	$K_g = 20$
Softmax temperature	$\tau = 0.1$
Maximum feature mixing weight	0.7
Anchoring coefficient	$\alpha = 0.3$
Anchored diffusion steps	$L = 2$

A Supplementary Experimental Protocol

Unless stated otherwise, all experiments share one configuration across datasets and backbones. Category labels are used only after ranking to compute retrieval metrics and post-hoc diagnostics.

A.1 Open-Set Benchmarks and Evaluation Splits

We follow the established open-set 3D object retrieval protocol of Feng et al. (2024) on four benchmarks. GEAR uses neither their training splits nor gallery labels: its offline stage receives unlabeled gallery descriptors, and its online stage processes each query independently. Dataset statistics are reported in Table 4.

A.2 Frozen Descriptor Construction

We use frozen CLIP ViT-L/14 Radford et al. (2021), DINOv2 ViT-B/14 Oquab et al. (2024), and DINOv3 ViT-B/16 Siméoni et al. (2026). Queries and gallery objects follow the same multi-view aggregation in Equation 2, and every comparison with a given backbone uses identical cached descriptors. Feature extraction visits objects without shuffling and uses seed 2022; after caching, refinement and evaluation are deterministic.

A.3 Unified GEAR Configuration

Table 5 lists the global preset used across all four benchmarks and three image backbones. Sensitivity analyses vary only the declared factor and do not replace these values with dataset-specific optima.

A.4 Metrics and Statistical Analysis

We report mAP, NDCG@100, and ANMRR using the public open-set evaluator. Higher mAP and NDCG@100 are better, whereas lower ANMRR is better. Metrics are first averaged over queries within each dataset; cross-dataset results are equal-weight macro averages over the four benchmarks rather than averages pooled over all queries.

For comparisons between Full GEAR and key ablations, we use 10,000 paired bootstrap resamples with seed 2027. Queries are resampled with replacement within each dataset using matched indices

for both methods, after which the four dataset-level differences are macro-averaged. We report percentile 95% confidence intervals and orient every difference so that a positive value indicates an improvement by Full GEAR.

A.5 Execution Environment

Refinement and evaluation use Python 3.10.19, NumPy 2.2.6, and SciPy 1.15.3 on Linux 6.8 with glibc 2.35. Runtime measurements restrict OMP, MKL, OpenBLAS, and NumExpr to one thread; the timing protocol is given in Section E. The supplementary archive includes an anonymized implementation of the core GEAR pipeline and its fixed configuration; the complete research code and evaluation scripts will be released upon publication.

B Additional Implementation Details

This section records implementation choices that are not explicit in the method equations. The reference implementation operates on cached descriptors with NumPy and SciPy and performs no gradient or parameter update.

B.1 Numerical and Data Conventions

Although the main paper writes the gallery as $\mathbf{T} \in \mathbb{R}^{d\times N}$, the implementation stores one object per row in $T \in \mathbb{R}^{N\times d}$. All cached descriptors use single precision, and row normalization uses $\epsilon = 10^{-8}$. Neighborhood softmaxes subtract the largest selected similarity before exponentiation for numerical stability. Neighbors are obtained by partial sorting; gallery self-similarities are suppressed before selection. The effective neighborhood sizes are $\min(K, N-1)$ for gallery updates and $\min(K_q, N)$ for query calibration.

B.2 Multi-Scale Refinement and Graph Construction

The four scales $K \in \{5, 10, 20, 50\}$ are independent branches rather than a sequential chain. Each branch starts from the same centered gallery, performs two self-update rounds, and contributes equally to the normalized fused representation. Confidence values are computed per sample from the selected cosine similarities, clipped to $[0, 1]$, and scaled by $\lambda_{\max} = 0.7$.

The gallery graph is constructed once from the refined representation T'. After retaining the $K_g = 20$ strongest non-self neighbors per row, the weighted adjacency matrix is symmetrized and row-normalized. The resulting transition matrix is stored in compressed sparse row format. No query node or query–query edge is introduced.

B.3 Offline and Online Execution

Gallery centering, multi-scale refinement, and graph construction are performed once whenever the gallery changes. Their outputs form the immutable cache $\mathcal{C} = (\boldsymbol{\mu}_t, T', P)$. At inference, each query reads this cache, performs query calibration and anchored score diffusion, and sorts its final scores; neither T' nor P is modified. Multiple rows may be evaluated together for efficient linear algebra, but every output remains a function only of its own query and the shared gallery cache. Appendix E.2 empirically verifies order invariance and equivalence to independent execution.

C Additional Quantitative Results

C.1 Improvement over Frozen Descriptors

Table 6 isolates the effect of the complete GEAR pipeline for each frozen descriptor family. Results are equal-weight macro averages over the four open-set benchmarks and use the unified configuration in Table 5.

GEAR improves all three metrics for every backbone, raising macro mAP by 7.32–9.28 percentage points without updating the feature encoder.

Table 6: **Macro-average improvement over frozen descriptors.** Metric triplets report mAP/NDCG@100/ANMRR (%). All delta columns are oriented so that positive values indicate improvement.

Backbone	Frozen	Full GEAR	Δ mAP	Δ NDCG	Δ ANMRR ↓
CLIP ViT-L/14	56.38/43.43/45.67	65.66/46.22/37.56	+9.28	+2.79	8.11
DINOv2 ViT-B/14	60.98/45.98/41.89	68.54/48.13/35.13	+7.56	+2.15	6.76
DINOv3 ViT-B/16	60.64/45.80/42.56	67.96/47.87/36.19	+7.32	+2.07	6.37

Table 7: **Progressive component ablation with independent Gallery and Query refinement.** Dataset columns report mAP (%); the final three columns report macro mAP/NDCG@100/ANMRR (%). M is the multi-scale extension of G, not an independent gallery update.

ID	Variant	Active modules	ESB	NTU	MN40	ABO	mAP↑	NDCG↑	ANMRR↓
A	Frozen descriptors	–	59.15	59.80	62.65	62.31	60.98	45.98	41.89
B	Target centering	C	61.13	61.21	66.71	66.26	63.83	46.89	39.95
C	Single-K gallery	$C + G$	62.55	63.71	71.08	69.51	66.71	47.58	37.07
D	Equal Multi-K gallery	$C + G + M$	62.58	63.74	70.94	69.36	66.66	47.62	37.12
E	Single-query refinement	$C + G + M + Q$	63.11	65.11	73.90	70.62	68.19	48.11	35.52
F	Full GEAR	$C + G + M + Q + D$	**63.31**	**65.44**	**74.40**	**70.99**	**68.53**	**48.13**	**35.13**

D Detailed Ablation Studies

Unless noted otherwise, results use DINOv2 ViT-B/14 descriptors and report equal-weight macro averages over the four benchmarks under the unified configuration in Table 5.

D.1 Disentangling the Core Components

We denote target-mean centering, single-scale ($K = 20$) gallery update, Multi-K fusion, single-query refinement, and anchored diffusion by C, G, M, Q, and D, respectively. Gallery and query refinement are controlled independently.

Rows C–E separate single-scale gallery updating, Multi-K fusion, and query refinement. Table 8 further shows that centering and gallery updating improve the frozen representation both individually and jointly.

D.2 Gallery Refinement Design

Figure 5 compares individual neighborhood scales and update depths. No single K dominates all metrics, while Multi-K avoids dataset-specific scale selection. We use two update rounds because NDCG@100 peaks there and additional rounds require extra offline passes.

D.3 Diffusion Strategy and Parameter Sensitivity

Table 9 compares propagation rules on the same graph. Ordinary diffusion is strongest in mAP and ANMRR at $L = 2$, whereas the principal advantage of iterative anchoring is its stability at greater depths in Figure 3, rather than dominance on every shallow-step metric.

Figure 6 shows a total mAP spread of 0.58 points over the tested (α, L) grid. We retain the predeclared setting $\alpha = 0.3, L = 2$ rather than selecting test-set maxima.

D.4 Neighborhood Sensitivity

Figure 7 shows that query aggregation is stable for $K_q \in \{1, 5, 10\}$ and declines for broader neighborhoods. Larger K_g provides only modest gains while increasing graph storage and propagation cost, motivating the fixed values $K_q = 10$ and $K_g = 20$.

Table 8: **Interaction between target centering and gallery self-update.** The gallery update uses a single neighborhood scale $K = 20$; query refinement and diffusion are disabled.

Centering	Gallery update	mAP↑	NDCG@100↑	ANMRR↓
✗	✗	60.98	45.98	41.89
✓	✗	63.83	46.89	39.95
✗	✓	65.38	47.09	38.27
✓	✓	**66.71**	**47.58**	**37.07**

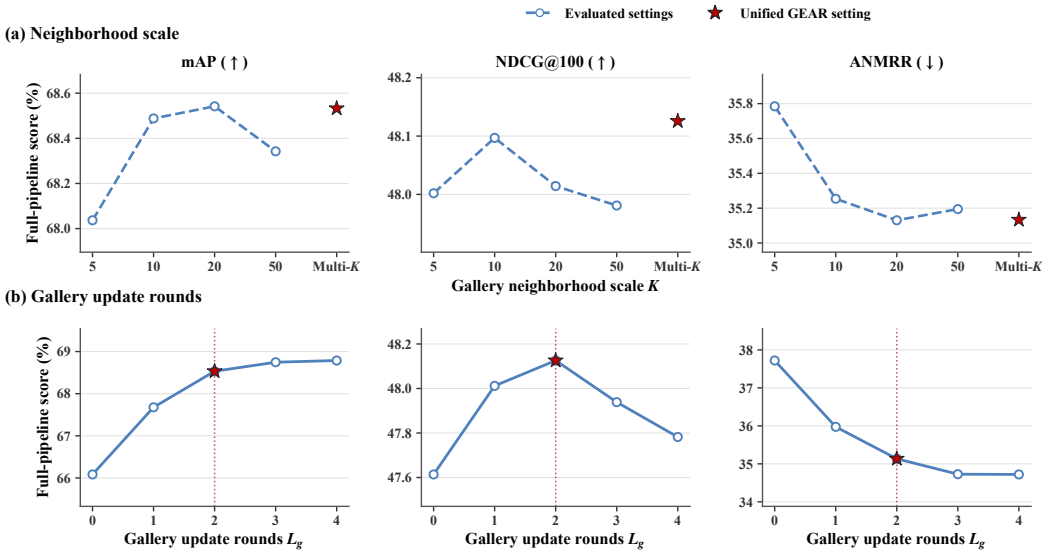

Figure 5: **Sensitivity of gallery refinement.** Top: complete-pipeline performance for four single neighborhood scales and equal Multi-K fusion. Bottom: complete-pipeline performance for gallery update rounds $L_g \in \{0, 1, 2, 3, 4\}$. Red stars mark the globally fixed GEAR settings; all values are four-dataset macro averages with DINOv2.

D.5 Paired Statistical Analysis

Under the protocol in Section A.4, the mAP and ANMRR intervals in Figure 8 are positive for all four comparisons. The NDCG@100 interval against no diffusion includes zero; evidence for anchoring therefore comes primarily from mAP/ANMRR and depth stability.

E Single-Query Validation and Efficiency

Using DINOv2 ViT-B/14 and the unified configuration, we verify query independence and separate one-time gallery preparation from online latency.

E.1 Sensitivity to Query Batch Size

For each $B \in \{2, 4, 8, 16, 32, 64\}$, we partition queries five times with seeds 2027–2031 and report four-dataset macro mAP. TeDA Wang et al. (2025b) uses the published dataset-specific temperatures; GEAR receives the same partitions but processes rows independently. Figure 9 shows that GEAR is invariant to B, whereas TeDA degrades with fewer queries. At $B = 1$, TeDA has no cross-query relation and returns the frozen baseline.

Table 9: **Score diffusion strategies with two propagation steps.** All variants use the same gallery graph and $\alpha = 0.3$; values are four-dataset macro averages (%).

Strategy	mAP↑	NDCG@100↑	ANMRR↓
No diffusion	68.19	48.11	35.52
Ordinary diffusion	**68.74**	47.96	**34.90**
Residual diffusion	68.65	48.12	35.04
Endpoint anchoring	68.60	**48.15**	35.06
Iterative anchoring (GEAR)	68.53	48.13	35.13

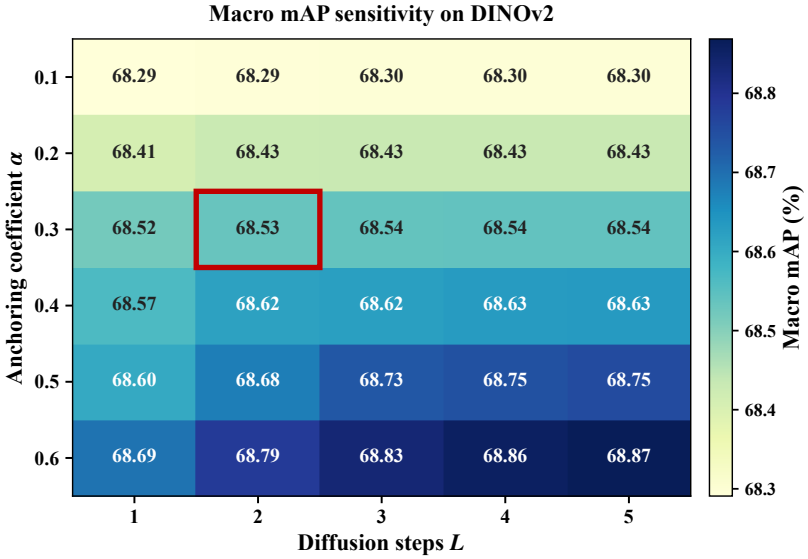

Figure 6: **Joint sensitivity to anchoring coefficient and propagation depth.** Cells report four-dataset macro mAP (%) using DINOv2. The red box marks the unified configuration $\alpha = 0.3, L = 2$, not a test-set optimum.

E.2 Order Invariance and Execution Equivalence

For each benchmark, we test five random query permutations and compare independent calls with grouped row-wise calls. Both checks give identical scores and rankings for all 635 queries. Fully vectorized execution differs only through BLAS accumulation order, with maximum score and metric differences of 3.46×10^{-6} and 4.20×10^{-7}, respectively (Table 10).

E.3 Offline Preparation and Online Latency

Using the single-thread environment in Section A.5, we repeat gallery preparation three times. Online timing uses three warm-up and five measured passes; Table 11 reports medians.

E.4 Computational Complexity

The offline expression in Section 3.5 consists of SL_g dense gallery similarity computations for multi-scale refinement and one for graph construction. The exact implementation uses $\mathcal{O}(N^2)$ working memory; the retained descriptors and sparse graph require $\mathcal{O}(Nd + NK_g)$ storage. Online, the terms respectively correspond to gallery scoring, aggregation over K_q neighbors, L sparse propagation steps, and exact sorting. Transient score memory is $\mathcal{O}(N)$ per query, and no term depends on the number of concurrently available queries.

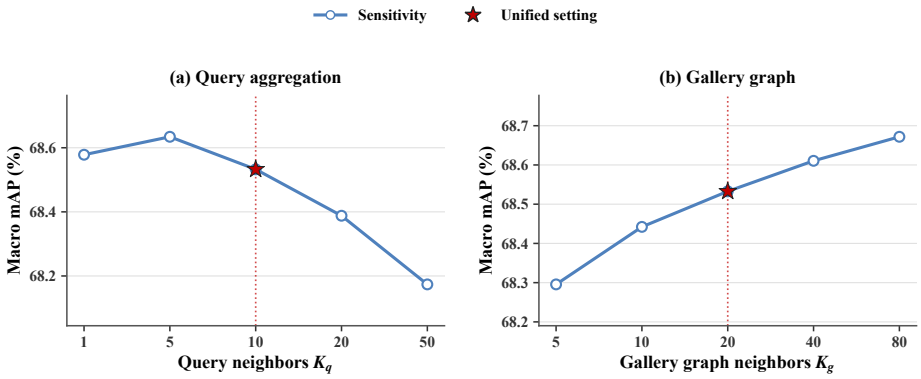

Figure 7: **Sensitivity to query and graph neighborhood sizes.** Curves show four-dataset macro mAP with DINOv2. Red stars mark the globally fixed values $K_q = 10$ and $K_g = 20$.

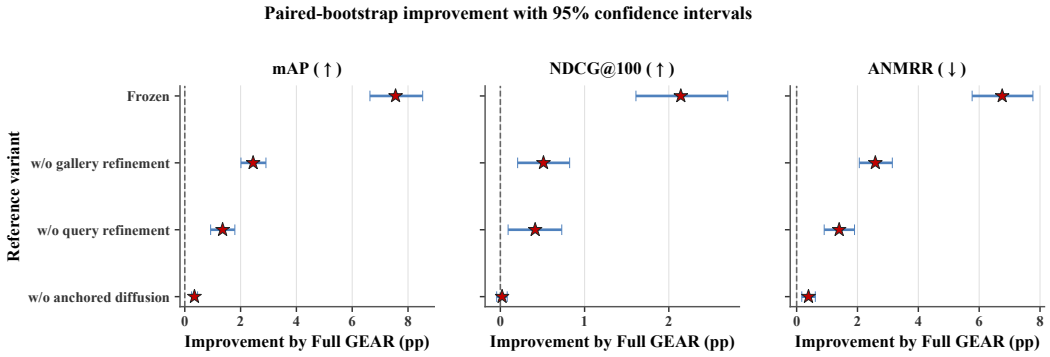

Figure 8: **Paired-bootstrap confidence intervals for key ablations.** Points show the four-dataset macro improvement of Full GEAR and bars show 95% confidence intervals from 10,000 paired query resamples. Positive values favor Full GEAR for all three panels.

F Post-Hoc Gallery Structure Analysis

We compare frozen, centered, and Multi-K refined gallery descriptors, using category labels only for post-hoc diagnostics. Query refinement and diffusion are excluded because they do not modify the cached gallery embedding T'.

F.1 Diagnostic Definitions

For gallery object i, let $\mathcal{N}_k(i)$ contain its k nearest gallery neighbors under cosine similarity. Neighborhood purity is

$$\text{Purity@}\,k = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{k}\sum_{j\in\mathcal{N}_k(i)}\mathbb{1}[y_i = y_j]. \tag{18}$$

We use $k = 10$; higher purity indicates greater local category consistency.

To evaluate global separability, we compute within-class and between-class scatter on ℓ_2-normalized descriptors. Let μ_c denote the mean of class c, μ the global mean, and n_c its sample count. The Fisher ratio is

$$\mathcal{F} = \frac{\sum_c n_c\|\mu_c - \mu\|_2^2}{\sum_c\sum_{i:y_i=c}\|t_i - \mu_c\|_2^2 + \epsilon}. \tag{19}$$

Higher $\mathcal{F}$ indicates stronger between-class separation relative to within-class scatter.

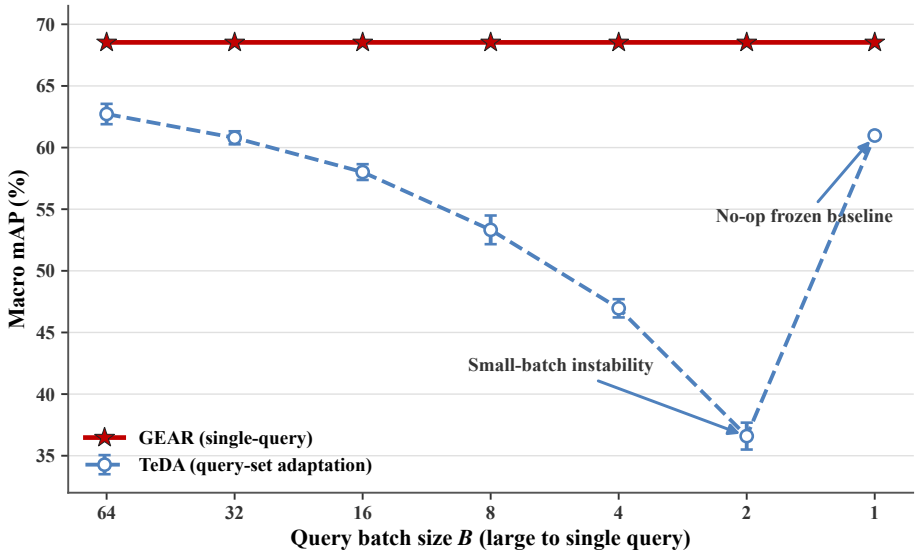

Figure 9: **Retrieval performance under decreasing query batch size.** Curves report four-dataset macro mAP using DINOv2. Error bars for TeDA show one standard deviation over five random query partitions; GEAR is deterministic and query-independent. At $B = 1$, TeDA has no cross-query relation and reduces to the frozen baseline rather than performing meaningful adaptation.

Table 10: **Validation of the single-query execution contract.** "Independent" compares sequential calls with grouped independent calls; "Vectorized" reports floating-point differences caused by batched BLAS evaluation.

Dataset	Queries	Order trials	Order score diff.	Independent score diff.	Vectorized score diff.	Metric diff.
OS-ESB-core	120	5	0	0	5.96×10^{-7}	0
OS-NTU-core	270	5	0	0	4.77×10^{-7}	0
OS-MN40-core	160	5	0	0	3.46×10^{-6}	4.20×10^{-7}
OS-ABO-core	85	5	0	0	5.36×10^{-7}	0

Following the standard k-occurrence formulation of hubness Radovanović et al. (2010), we count how frequently each gallery object appears in its peers' neighborhoods. For gallery object j, let $o_j = \sum_i \mathbb{1}[j \in \mathcal{N}_k(i)]$, with mean $\bar{o}$ and standard deviation σ_o. We report the standardized third moment

$$\mathrm{HubSkew@}\, k = \frac{1}{N} \sum_{j=1}^{N} \left(\frac{o_j - \bar{o}}{\sigma_o} \right)^3 . \tag{20}$$

Lower positive skewness indicates fewer dominant nearest-neighbor hubs.

We also report mean intra- and inter-class cosine similarities and the gallery centroid norm. The latter is only a centering sanity check, since subtracting the target mean reduces it by construction.

F.2 Macro-Average Structural Changes

Table 12 complements the Purity@10 and Fisher results in Section 4.4. Centering removes the positive common component in inter-class similarity, while gallery refinement increases intra-class similarity and reduces hubness skewness.

F.3 Per-Dataset Structural Results

Table 13 shows improved Fisher ratio and reduced hubness relative to frozen descriptors on all four datasets. Purity improves on ESB, NTU, and MN40 but is effectively unchanged on ABO, whose frozen purity is already high.

Table 11: **Offline gallery preparation and online single-query latency.** Preparation is measured in seconds; all online columns are milliseconds per query. Score generation includes query refinement, initial scoring, and two anchored diffusion steps.

Dataset	Gallery	Prep. (s)	Score (ms)	Rank (ms)	Total (ms)	P95 (ms)
OS-ESB-core	452	0.27	0.59	0.05	0.64	0.74
OS-NTU-core	1,271	1.03	0.95	0.14	1.09	1.26
OS-MN40-core	9,329	21.41	4.63	1.23	5.87	6.16
OS-ABO-core	5,455	8.95	2.76	0.67	3.44	3.70
Macro average	–	7.92	2.24	0.52	2.76	2.96

Table 12: **Complete macro-average gallery diagnostics.** Purity uses $k = 10$. Higher Purity and Fisher ratio are better; lower hubness skewness indicates fewer dominant nearest-neighbor hubs.

Gallery state	Purity@10 (%)	Intra-class cos.	Inter-class cos.	Fisher ratio	Hubness skew.	Centroid norm
Frozen	82.17	0.627	0.309	0.839	0.852	0.575
Centered	82.68	0.439	−0.033	0.868	0.746	0.031
Gallery refined	**83.11**	0.557	−0.041	**1.421**	**0.468**	0.038

These post-hoc results support improved gallery structure on average, not a claim that every class becomes uniformly compact.

G Backbone and Scenario Generalization

We test whether component effects transfer across visual encoders and report additional metrics for the scenarios summarized in the main paper.

G.1 Component Transfer across Visual Backbones

The controlled variants in Table 14 remove gallery refinement, query refinement, or anchored diffusion while retaining all other components.

All three removals reduce mAP across CLIP, DINOv2, and DINOv3, with gallery refinement giving the largest contribution.

G.2 Transfer beyond Multi-View Object Retrieval

The main paper reports mAP in Table 2; the tables below provide complementary task-specific metrics without repeating mAP.

MRR decreases in some sketch-based settings, so GEAR does not improve every metric on every task. All scenarios retain the fixed configuration without modality-specific tuning.

H Additional Sensitivity and Limitations

We report two additional sweeps using four-dataset macro averages without dataset-specific parameter selection.

H.1 Softmax Temperature

The temperature τ is shared by gallery refinement, query aggregation, and graph construction. Table 18 shows stable performance over $[0.1, 0.5]$; we retain $\tau = 0.1$, which gives the highest NDCG@100.

Table 13: **Per-dataset gallery structure diagnostics.** F, C, and R denote frozen, centered, and gallery-refined descriptors.

Dataset	State	Purity@10 (%)↑	Fisher ratio↑	Hubness skew.↓
OS-ESB-core	F	69.93	0.918	0.777
	C	71.00	0.949	**0.569**
	R	**71.08**	**1.380**	0.745
OS-NTU-core	F	73.04	0.917	0.666
	C	73.75	0.946	0.567
	R	**75.03**	**1.753**	**0.465**
OS-MN40-core	F	91.49	0.813	0.869
	C	91.71	0.812	0.844
	R	**92.05**	**1.441**	**0.252**
OS-ABO-core	F	94.21	0.710	1.096
	C	**94.28**	0.762	1.005
	R	94.26	**1.112**	**0.412**

Table 14: **Backbone-wise component ablation.** Entries are four-dataset macro averages (%) computed from unrounded per-query metrics. Higher mAP and NDCG@100 are better; lower ANMRR is better.

Backbone	Variant	mAP↑	NDCG@100↑	ANMRR↓
CLIP ViT-L/14	Frozen	56.38	43.43	45.67
	w/o Gallery refinement	63.24	45.73	40.21
	w/o Query refinement	64.16	45.57	39.07
	w/o Anchored diffusion	65.30	46.17	37.73
	Full GEAR	**65.65**	**46.22**	**37.55**
DINOv2 ViT-B/14	Frozen	60.98	45.98	41.89
	w/o Gallery refinement	66.08	47.61	37.72
	w/o Query refinement	67.18	47.71	36.53
	w/o Anchored diffusion	68.19	48.11	35.52
	Full GEAR	**68.53**	**48.13**	**35.13**
DINOv3 ViT-B/16	Frozen	60.64	45.80	42.56
	w/o Gallery refinement	65.43	47.29	38.55
	w/o Query refinement	66.92	47.49	36.85
	w/o Anchored diffusion	67.58	47.77	36.51
	Full GEAR	**67.96**	**47.87**	**36.19**

H.2 Propagation Depth across Backbones

Table 19 extends the depth analysis to all three encoders. Deeper propagation changes ranking metrics only marginally while increasing online cost, motivating the fixed depth $L = 2$.

H.3 Scope and Limitations

GEAR assumes a sufficiently stable gallery to amortize refinement and graph construction. Gallery changes require recomputing affected descriptors, neighborhoods, and transitions. The exact implementation also has quadratic offline similarity cost (Section E.4); approximate graph construction for much larger galleries is outside this study.

GEAR also relies on semantically consistent local neighborhoods. Systematically mixed frozen neighborhoods can propagate incorrect proximity; the bounded residual update mitigates but cannot eliminate this failure. Accordingly, the cross-modal experiments do not establish improvement on every metric in every setting.

Table 15: **Additional ModelNet40 metrics for pretrained point-cloud backbones.** Corresponding mAP results appear in the main paper.

Representation	Method	nDCG@All↑	Recall@10↑
OpenShape-PointBERT	Baseline	89.74	24.43
	+ GEAR (Ours)	**92.40**	**26.19**
Point-MAE	Baseline	79.52	18.22
	+ GEAR (Ours)	**81.33**	**18.47**

Table 16: **Additional cross-modal retrieval metrics.** We evaluate sketch-based shape retrieval (SBSR) and sketch-based image retrieval (SBIR); corresponding mAP results appear in the main paper.

Setting	Method	nDCG↑	MRR↑	NN↑	FT↑	ST↑
Zero-shot SBSR	MEHA-ZS	57.1	**46.2**	31.6	30.5	48.0
	+ GEAR	**57.3**	44.2	**31.7**	**32.9**	**52.9**
Category-level SBSR	MEHA	92.6	88.1	86.8	88.7	93.7
	+ GEAR	**92.9**	**88.2**	**87.2**	**89.1**	**93.9**

Sketch-Based Image Retrieval

Dataset	Method	Precision↑	nDCG↑	MRR↑
Sketchy Ext	Sketch_LVM Sain et al. (2023)	70.05	71.25	**80.68**
	+ GEAR	**75.66**	**76.04**	77.27
TU-Berlin Ext	Sketch_LVM Sain et al. (2023)	66.18	85.47	**76.34**
	+ GEAR	**71.61**	**89.47**	73.76

I Qualitative Protocol and Case Analysis

We document the protocol for Figure 4. Labels are used only after inference for sampling, coloring, and correctness annotations.

I.1 Embedding-Space Visualization

Figure 4(a) uses OS-MN40-core and CLIP ViT-L/14. For eight categories, we sample 40 gallery objects and five queries per class without replacement (seed 42), reusing identical indices before and after refinement.

All frozen and refined query/gallery descriptors are concatenated before one shared PCA(50) and t-SNE run (perplexity 30, 2,000 iterations, seed 43). Both panels share axis limits, avoiding independent projection or rescaling.

"After GEAR" contains centering, equal Multi-K gallery refinement, and single-query aggregation. Anchored diffusion is excluded because it changes ranking scores, not embeddings.

I.2 Retrieval-Case Selection and Exact Rankings

Figure 4(b) uses real OS-ABO-core queries and DINOv2. Cases are selected post hoc by requiring five correct GEAR results, then prioritizing top-5 and AP gains across distinct categories. They are illustrative, not a random sample; Table 20 reports their exact measurements.

The negative AP changes for sofa and dresser are retained to avoid overstating the top-5 visualization.

Table 17: **Additional SUN397 scene-retrieval metrics.** Results are mean ± standard deviation; corresponding mAP results appear in the main paper.

Backbone	Method	Hit@1↑	P@10↑	R@10↑	nDCG@10↑
Places365-ResNet50	Baseline	61.47±0.69	53.16±0.77	10.63±0.15	54.89±0.78
	+ GEAR	**62.76±0.70**	**58.03±0.76**	**11.61±0.15**	**59.02±0.76**
CLIP ViT-L/14	Baseline	70.07±0.26	59.96±0.44	11.99±0.09	62.10±0.39
	+ GEAR	**71.89±0.08**	**65.28±0.37**	**13.06±0.07**	**66.65±0.32**
DINOv2 ViT-B/14	Baseline	69.38±0.15	61.28±0.32	12.26±0.06	62.98±0.29
	+ GEAR	**69.49±0.05**	**63.59±0.39**	**12.72±0.08**	**64.82±0.33**

Table 18: **Sensitivity to the shared softmax temperature.** All entries are four-dataset macro averages (%) obtained with DINOv2 and the complete GEAR pipeline. The fixed paper setting is underlined.

τ	mAP↑	NDCG@100↑	ANMRR↓
0.05	67.58	47.97	36.12
<u>0.10</u>	68.53	**48.13**	35.13
0.20	**68.97**	48.12	34.72
0.50	68.93	48.11	**34.68**

Table 19: **Sensitivity of iterative anchoring to propagation depth.** Entries are four-dataset macro averages (%). The underlined column is the globally fixed GEAR depth.

Backbone	Metric	$L=1$	<u>$L=2$</u>	$L=5$	$L=10$
CLIP ViT-L/14	mAP↑	65.63	65.65	65.67	65.67
	NDCG@100↑	46.22	46.22	46.22	46.22
	ANMRR↓	37.58	37.55	37.55	37.55
DINOv2 ViT-B/14	mAP↑	68.52	68.53	68.54	68.54
	NDCG@100↑	48.13	48.13	48.13	48.13
	ANMRR↓	35.14	35.13	35.12	35.12
DINOv3 ViT-B/16	mAP↑	67.92	67.96	67.97	67.97
	NDCG@100↑	47.87	47.87	47.87	47.87
	ANMRR↓	36.22	36.19	36.18	36.18

Table 20: **Exact measurements for the qualitative retrieval cases.** AP is reported per query (%). Positive ΔAP favors GEAR.

Query category	Baseline AP	GEAR AP	ΔAP
Vase	59.14	89.29	+30.15
Bench	18.63	34.21	+15.58
Sofa	96.95	96.64	−0.31
Dresser	53.49	47.72	−5.77