Tango3D: Towards Alignment for Global and Local 2D-3D Correspondence

Zebin He[1,2*] **Mingxin Yang**[2] **Hanxiao Sun**[1,2] **Shuhui Yang**[2]
Xintong Han[2] **Chunchao Guo**[2†] **Wenhan Luo**[1†]

[1]HKUST [2]Tencent Hunyuan



Figure 1: Our framework Tango3D unifies fine-grained correspondence and global retrieval, supporting diverse cross-modal and intra-modal matching tasks from pixel-level grounding to instance-level retrieval. More results can be found on the webpage `https://zebinhe.github.io/Tango3D/`.

Abstract

Existing 3D foundation models typically align point clouds to frozen vision-language spaces like CLIP, which achieve strong cross-modal retrieval by compressing 3D shape into a global vector. However, this global-only alignment cannot establish fine-grained pixel-to-point correspondence. To solve this, we present Tango3D, a foundation model that unifies dense correspondence and global retrieval. We use a geometry-aware 2D visual backbone and a pretrained 3D VAE to encode images into 2D patches and point clouds into 3D tokens. These are mapped into a single shared space to achieve both local pixel-to-point alignment and global semantic alignment. To stabilize the joint learning of dense and global

*Work done during internship at Tencent Hunyuan.
†Corresponding Author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

objectives, we introduce a three-stage progressive training strategy. Experiments show our model successfully achieves object-level pixel-to-point alignment while maintaining competitive global retrieval, a joint capability not offered by existing 3D foundation models. By establishing a fine-grained alignment feature space, Tango3D injects rich semantics into purely geometric 3D tokens, paving the way for a wide range of dense 3D downstream tasks.

1 Introduction

Aligning 3D point clouds with 2D visual foundations has become the standard approach for 3D representation learning. By training 3D encoders against frozen vision-language spaces like CLIP [1], recent 3D foundation models have achieved remarkable success. This alignment paradigm successfully injects open-vocabulary semantics into 3D representations, establishing a strong foundation for zero-shot classification and shape retrieval.

However, this paradigm suffers from a fundamental limitation: the alignment is exclusively global. Existing methods compress an entire 3D shape into a single feature vector. They typically use purely semantic spaces like CLIP as alignment anchors, which are designed for semantic richness but inherently lack spatial structure. Consequently, while sufficient for retrieval, this compression discards the spatial details needed for pixel-to-point correspondence. Such dense alignment is critical for fine-grained tasks like 3D part segmentation. Although scene-level methods [2, 3] have explored dense matching, object-level 2D-3D dense alignment remains an open problem.

To solve this, we present Tango3D, a foundation model that unifies dense correspondence and global retrieval. Instead of relying on purely semantic spaces, we use the geometry-aware 2D visual backbone VGGT [4] as our alignment anchor to provide the necessary spatial structure. Furthermore, rather than compressing 3D shapes into a single vector, we utilize a 3D VAE to encode point clouds into 3D tokens. By establishing a shared token space, 2D feature patches and 3D tokens can interact directly. This design naturally supports fine-grained pixel-to-point alignment while preserving strong global semantic capabilities.

However, jointly optimizing both dense correspondence and global retrieval presents a significant challenge. To stabilize this learning process, we introduce a three-stage progressive training strategy. This strategy guides the model from establishing robust fine-grained local alignments to integrating global semantic representations, facilitating joint learning of dense correspondence and global retrieval. Experiments demonstrate that Tango3D successfully achieves object-level pixel-to-point correspondence while maintaining competitive global retrieval performance. Our approach also enables zero-shot semantic matching at both the object and point levels. By bridging the gap between 2D and 3D modalities, our approach lays the foundation for exploring robust and unified representations, which could potentially benefit future complex 3D downstream tasks.

In summary, our main contributions are:

Local alignment. We propose Tango3D, a foundation model that achieves object-level pixel-to-point dense correspondence by mapping 2D patches and 3D tokens into a shared space. The learned space supports bidirectional queries between pixels and points, cross-instance semantic matching at the point level, and 3D-to-3D correspondence without 2D input.

Global alignment. We unify dense correspondence and instance-level retrieval within a single framework. The aggregated global descriptors support both image-to-shape and shape-to-shape retrieval, achieving competitive performance against leading global-only baselines.

Progressive training. We introduce a three-stage progressive strategy to stabilize the joint optimization. It first optimizes geometric correspondence, then integrates global semantics, and finally refines both at higher resolution, leading to a more coordinated optimization of the two objectives.

2 Related Work

Recent research on 3D modeling spans shape and material generation [5, 6, 7], controllable editing [8, 9], and human animation [10]. Related efforts also explore 3D and 4D reconstruction from images [11, 12, 13, 14, 15] and monocular 3D visual grounding [16].

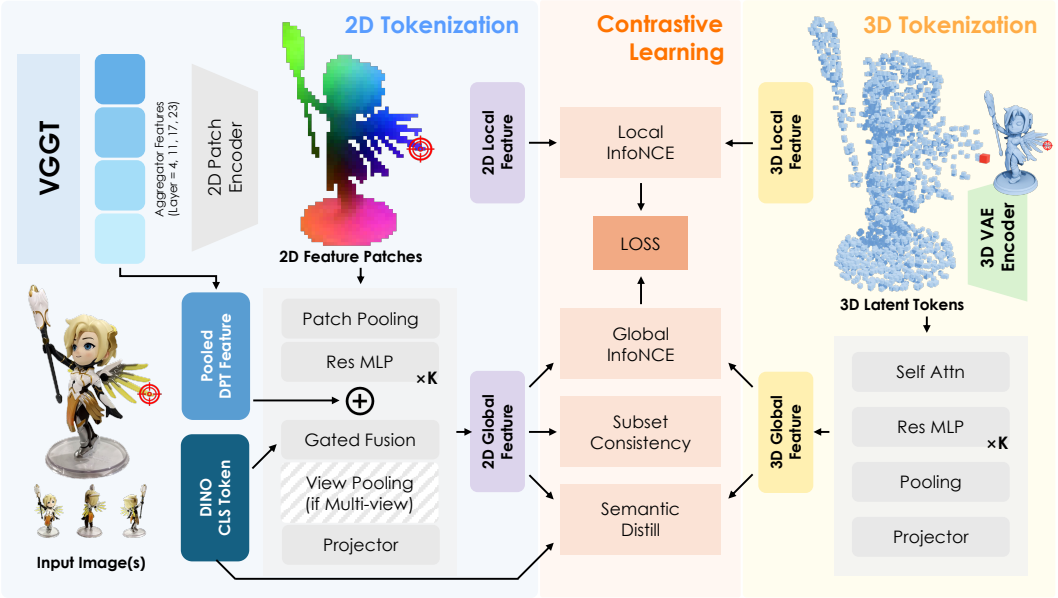

Figure 2: **Overview of Tango3D.** We project 2D patch features from a frozen VGGT backbone and 3D latent tokens from a trainable VAE into a shared embedding space, jointly optimized by a local branch for pixel-to-point correspondence and a global branch for instance-level retrieval.

Within this broader landscape, our work focuses on representations that connect images and 3D geometry at both the object and point levels. We therefore review two closely related directions: multi-modal 3D representation learning for global semantic alignment, and dense 2D-3D correspondence for local geometric matching.

Multi-modal 3D representation learning. CLIP [1] learns a joint image and text embedding space, and subsequent works [17, 18] further improve efficiency. Recent advances in 3D representation learning have rapidly evolved from designing robust unimodal architectures [19, 20, 21] to multi-modal alignment frameworks. Early approaches adopt lightweight 3D encoders [22, 23, 24, 25] or depth encoders [26], and align their outputs to the frozen CLIP embedding space. ULIP [27] and ULIP-2 [28] construct image, language, and 3D triplets and train a Point-BERT backbone. OpenShape [29] focuses on data curation by ensembling multiple 3D datasets with filtered and enriched text descriptions. All of these methods use a frozen CLIP ViT (typically ViT-bigG-14) as the 2D encoder and only train the 3D branch. Uni3D [30] directly reuses a 2D ViT as the 3D encoder and achieves strong zero-shot classification. Beyond instance-level alignment, MRD [31] distills relational structure from the pretrained CLIP space, preserving both intra-modal and cross-modal similarity distributions. MixCon3D [32] combines multi-view CLIP image features with point-cloud features into a holistic embedding for text alignment. OpenView [33] further introduces a cross-modal fusion encoder that jointly processes point clouds and multi-view images.

Despite their success, these methods are limited to global-level alignment and cannot establish dense pixel-to-point correspondences, as both the single-vector compression and the purely semantic CLIP anchor discard the spatial structure required for fine-grained geometric grounding.

2D-3D dense correspondence. Traditional works establish 2D-3D correspondences via multi-stage SfM pipelines [34, 35, 36, 37, 38, 39]. More recent efforts instead learn these correspondences end-to-end in a differentiable manner [40, 41, 42, 43, 44, 45, 46, 47, 48, 49]. DUSt3R [50] predicts dense per-pixel 3D coordinates from image pairs, MASt3R [51] adds explicit local descriptors trained with a matching loss to ground correspondences in 3D, Fast3R [52] generalizes the pairwise design to process many views in a single forward pass, and Sem-MASt3R [53] injects DINOv2 semantics to disambiguate textureless regions. While achieving impressive 2D-to-3D correspondences, these methods target scene-level reconstruction and do not address cross-modal alignment between image and point clouds. Other scene-level efforts further explore pixel-to-point matching via contrastive distillation or implicit cross-attention [2, 3]. In contrast, our work targets object-level 2D-3D alignment across modalities, and simultaneously maintains a global retrieval capability.

Table 1: **Pixel-to-point local correspondence.** We evaluate dense geometric alignment by retrieving the nearest 3D tokens for randomly sampled 2D foreground pixels. LocAcc@k is reported under both single- and multi-view configurations.

	Baselines		**Tango3D (ours)**				
Input views	VGGT [4]@1	VGGT-OBJ [54]@1	@1	@2	@3	@5	@10
S=1, random	–	–	77.43	80.24	82.02	84.23	87.14
S=4, random	–	–	77.37	80.26	82.07	84.37	87.38
S=4, ortho	72.04	72.16	**76.37**	79.49	81.42	83.85	87.00

3 Method

As illustrated in Fig. 2, given image(s) $\mathcal{I} = \{I_s\}_{s=1}^{S}$ ($S \geq 1$) and a surface point cloud $\mathcal{P} \in \mathbb{R}^{N\times 6}$, our framework learns a unified embedding space that simultaneously supports fine-grained 2D-3D correspondence and instance-level cross-modal retrieval. To achieve this, we propose a dual-branch architecture built upon a shared token space: a local branch for geometry-grounded dense alignment, and a global branch for object-level semantic aggregation.

3.1 2D and 3D tokenization

Our framework leverages two pretrained models to extract geometry-aware representations. For the 2D and 3D modalities, we employ VGGT [4] and the encoder of the Hunyuan3D 2.1 variational autoencoder [55], respectively. To extract 2D features, each view I_s is processed by the frozen VGGT backbone and its DPT refinement head in a single forward pass, producing four intermediate feature maps and one fused DPT feature map. These feature maps are resized to the patch grid of the current training stage and concatenated into a 3072-dimensional feature vector $x_{s,u,v} \in \mathbb{R}^{3072}$ per grid position. The fused DPT map is also globally averaged to yield a 256-dimensional per-view context c_s. During Stage I and Stage II, VGGT operates at 518×518 input resolution, producing a 37×37 patch grid; in Stage III, the resolution is increased to 1022×1022, giving a 73×73 grid. To construct geometry-aware 2D queries for local supervision, we sample each rendered position map at the same grid centers. We aggregate valid patches across views and retain at most 512 queries via farthest-point sampling. For each retained query, we extract its 3D coordinate q_m, its corresponding feature vector x_m, and record its source-view index $\pi(m)$.

For the 3D modality, the point cloud is processed by a trainable 3D VAE encoder, yielding 1024 latent tokens $z_n \in \mathbb{R}^{D_{\text{vae}}}$ and their spatial centers $p_n \in \mathbb{R}^3$. We use the encoder features before posterior sampling, which retain richer geometric detail than the bottleneck latent.

We freeze VGGT to preserve its pretrained geometric features. The VAE's FPS queries already provide spatial anchors; fine-tuning its reconstruction-pretrained features allows the encoding stage to adapt to cross-modal semantic alignment, while keeping the subsequent alignment heads lightweight.

Finally, both modalities are mapped into a shared 1024-dimensional space via residual MLP encoders:

$$h_m^{\text{2D}} = f_{\text{2D}}(x_m,\ c_{\pi(m)}), \qquad h_n^{\text{3D}} = f_{\text{3D}}(z_n),$$

where the 2D encoder injects the per-view context. This shared token space serves as the unified foundation for the entire framework, directly supervised by the local branch and aggregated by the global branch.

3.2 Local branch

We project the shared tokens h_m^{2D} and h_n^{3D} into modality-specific local descriptors via linear heads and ℓ_2-normalization. For each 2D query q_m, we identify its positive 3D token via nearest-neighbor assignment in the geometric space: $n^\star(m) = \arg\min_n \|q_m - p_n\|_2$.

Because the projected 2D queries may not perfectly align with the discrete 3D token centers, we apply a Gaussian confidence weight $w_m = \exp(-\|q_m - p_{n^\star}\|_2^2/2\sigma^2)$ to each query and exclude 3D tokens within a spatial distance δ of q_m from the negative set. We optimize these descriptors using a

Table 2: **Surface-distance localization.** SurfAcc with four input views. The Stage I model uses random views, while the VGGT pointmap reference uses orthographic views.

Method	Input views	SurfAcc@1	SurfAcc@5	SurfAcc@10
VGGT reference	S=4, ortho	57.0	–	–
Tango3D (Stage I)	S=4, random	61.8	74.3	79.5

bidirectional InfoNCE contrastive loss:

$$\mathcal{L}_{\text{local}} = \frac{1}{2}\big(\mathcal{L}_{\text{2D}\to\text{3D}} + \mathcal{L}_{\text{3D}\to\text{2D}}\big),$$

where each query's contribution to the loss is weighted by w_m. Instead of relying solely on the forward matching, the reverse direction explicitly aggregates all 2D queries assigned to the same 3D token. This bidirectional formulation prevents the network from collapsing into a many-to-one mapping. During later stages, the loss is augmented with top-k hard negatives [29, 33] in each direction, where both the number of hard negatives k and their loss weight are increased across stages to progressively sharpen discriminability.

3.3 Global branch

To aggregate the shared tokens into instance-level representations, the global branch extracts a single descriptor per modality. For the 2D modality, we first pool the shared 2D tokens within each view to form preliminary view tokens $\tilde{r}_s$. To inject rich semantics without compromising the established geometric grounding, we enrich these view tokens using the DPT context c_s and the DINO class token d_s [56] from the VGGT backbone:

$$r_s = \tilde{r}_s + \lambda_c\, U c_s + \lambda_d\, \gamma_s \odot W_d d_s, \qquad \gamma_s = \sigma\big(\text{MLP}([\tilde{r}_s;\, W_d d_s])\big),$$

with $\lambda_c = 0.5$ and $\lambda_d = 1.0$. By applying this gated DINO fusion strictly after local evidence is pooled, we ensure the DINO signal acts as semantic enrichment rather than a shortcut that bypasses local alignment. A residual MLP then refines these view tokens before a second stage pools them across all valid views to produce the normalized 2D global descriptor g^{2D}.

For the 3D modality, the shared 3D tokens first pass through a self-attention [57] layer and a feed-forward network. This step enables spatially distant parts of the shape to exchange context. The contextualized 3D tokens are subsequently pooled, projected, and normalized to yield the 3D global descriptor g^{3D}.

To align these global representations, we apply a symmetric InfoNCE cross-modal contrastive loss over the distributed batch of size B:

$$\mathcal{L}_{\text{global}} = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{\exp(g_i^{\text{2D}}\cdot g_i^{\text{3D}}/\tau_g)}{\sum_{j=1}^{B}\exp(g_i^{\text{2D}}\cdot g_j^{\text{3D}}/\tau_g)} + \log\frac{\exp(g_i^{\text{3D}}\cdot g_i^{\text{2D}}/\tau_g)}{\sum_{j=1}^{B}\exp(g_i^{\text{3D}}\cdot g_j^{\text{2D}}/\tau_g)}\right]$$

, where τ_g is the contrastive temperature, and g_i denotes the global descriptor for the i-th sample in the batch. We further regularize the 2D descriptor against missing views via a subset-consistency loss $\mathcal{L}_{\text{sub}}$, requiring random view subsets to match the full-view descriptor.

Finally, inspired by relation distillation [31], we introduce a semantic distillation loss $\mathcal{L}_{\text{sd}}$ to transfer semantic knowledge from the 2D teacher. Instead of regressing the teacher features directly, we distill their relational structure. Given the mean DINO class token t_i (where t_i is the average of the class tokens d_s over valid views of sample i), we compute a teacher similarity matrix and align it with the student's cross-modal similarity:

$$\mathcal{L}_{\text{sd}} = \text{KL}\Big(\text{softmax}\big(\frac{TT^\top}{\tau_d}\big)\;\Big\|\;\text{softmax}\big(\frac{G^{\text{2D}}(G^{\text{3D}})^\top}{\tau_d}\big)\Big),$$

where τ_d is the distillation temperature. This formulation encourages the joint feature space to inherit the semantic neighborhood structure of the teacher while remaining inherently cross-modal. Relational targets convey which objects should be similar without requiring the student to reproduce the teacher's feature coordinates.

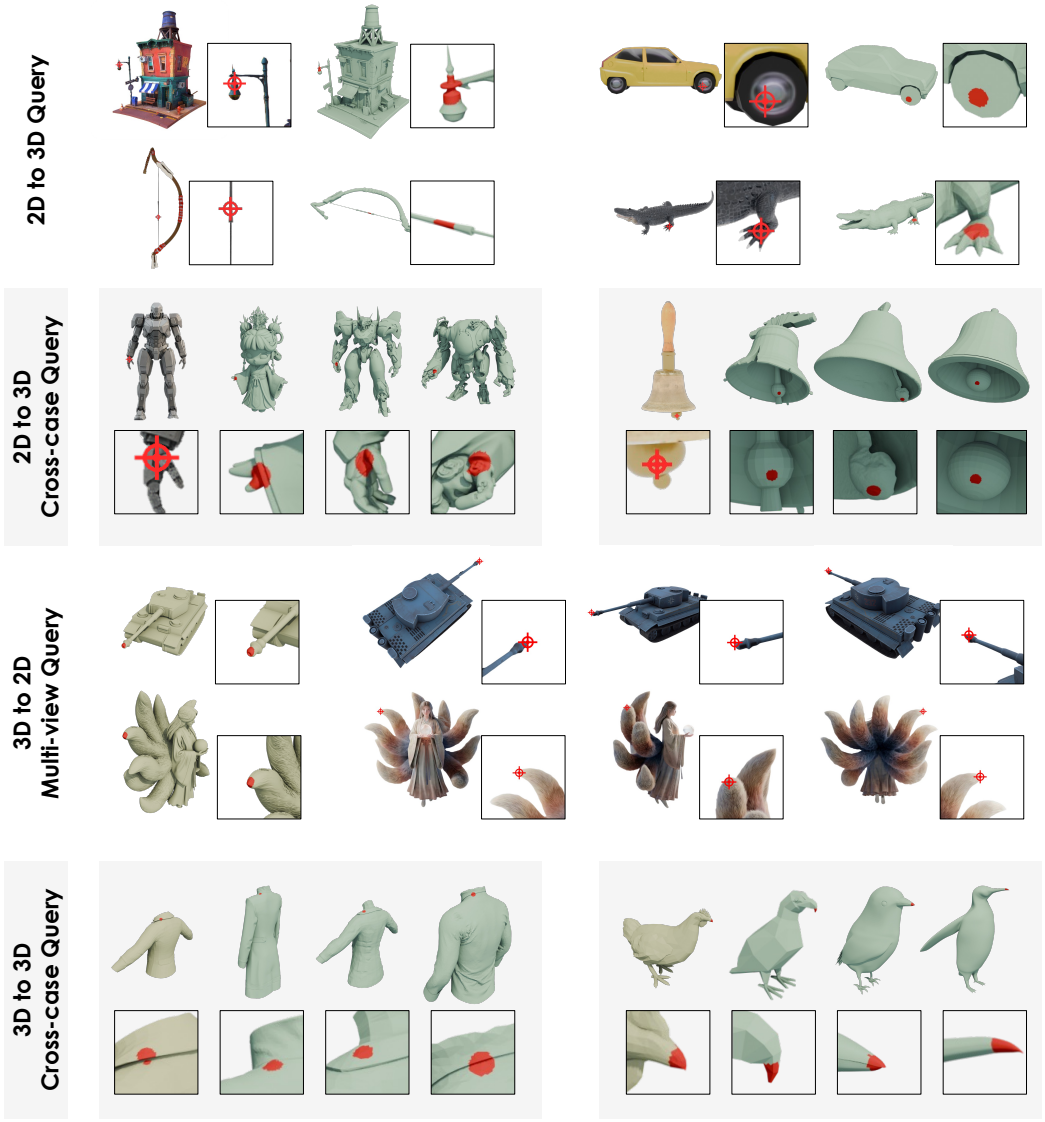

Figure 3: **Fine-grained local alignment.** The shared token space establishes accurate dense correspondences across diverse cross-modal and intra-modal scenarios.

3.4 Progressive training

To stably couple the dense and global objectives, we train the model in three progressive stages. In Stage I, we disable the global branch and train only the shared encoders, local heads, and the 3D VAE encoder using $\mathcal{L}_{\text{local}}$ without hard-negative mining. This establishes the foundational pixel-to-point geometric alignment. In Stage II, we enable the global branch to integrate semantic representations. The model is jointly optimized, and the full objective combines $\mathcal{L}_{\text{local}}$, $\mathcal{L}_{\text{global}}$, $\mathcal{L}_{\text{sub}}$, and $\mathcal{L}_{\text{sd}}$. To preserve the correspondence structure learned in Stage I, we apply a reduced learning rate to the local modules, while introducing hard-negative mining to the local loss. In Stage III, we scale up the 2D input resolution and apply strengthened hard-negative mining. All modules are jointly fine-tuned to sharpen overall feature discriminability. Training configurations and evaluation details are provided in Appendix A.

4 Experiments

Training data. We train Tango3D on a large-scale 2D-3D dataset comprising approximately 650,000 objects sourced from Objaverse [58, 59], including a portion of the Objaverse-LVIS dataset. For the

Table 3: **Image-to-shape retrieval.** Category-level Recall@k and MRR on our ModelNet40 benchmark. Baselines use a single random view (S=1); our model is evaluated under both single-view and multi-view settings.

Method	Input views	R@1	R@2	R@3	R@5	MRR
Uni3D-g [30]	S=1	32.60	41.68	45.63	53.43	42.70
OpenShape [29]	S=1	**36.44**	**47.50**	**52.74**	59.02	**48.19**
ULIP-2 [28]	S=1	33.88	42.96	48.54	54.60	43.98
Tango3D (ours)	S=1	35.04	45.17	52.50	**62.28**	47.17
Tango3D (ours)	S=4, ortho	37.60	49.94	**56.23**	63.10	49.67
Tango3D (ours)	S=4, random	**39.12**	**50.64**	55.88	**63.21**	**50.69**

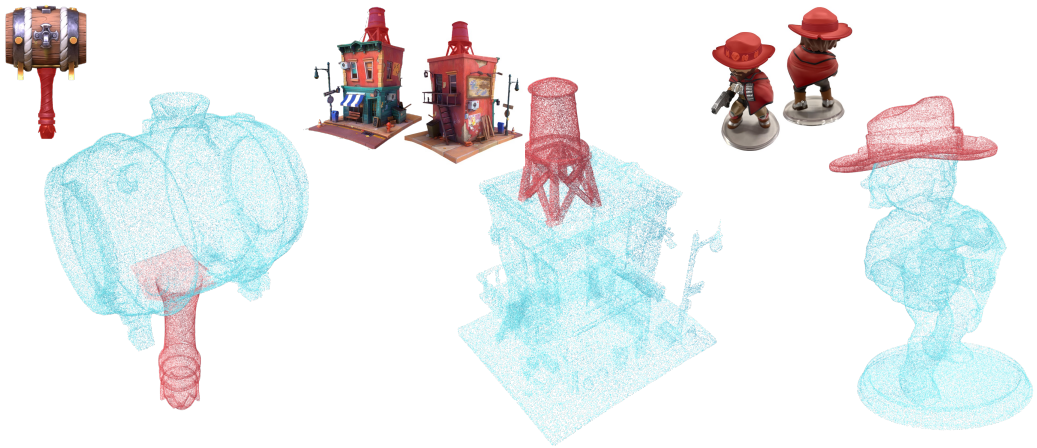

Figure 4: **2D-to-3D part transfer.** 2D regions selected via SAM is mapped onto the 3D point clouds using our local descriptor space.

3D modality, we extract a surface point cloud of $20{,}480$ points with coordinates and normals for each object. For the 2D modality, we render 60 views (6 orthographic and 54 random) per object, pairing each view with a RGBA position map. During training, we uniformly sample $S \in \{1, 2, 3, 4\}$ views per object.

Evaluation benchmark. To prevent data leakage from the training set's inclusion of Objaverse-LVIS, we construct a custom benchmark rather than relying on standard LVIS evaluation splits. This benchmark comprises $3{,}055$ objects covering the 40 ModelNet40 [60] categories. Since the original ModelNet40 meshes lack the textures required for multi-view rendering, we source textured instances from a mixture of generated assets and public 3D collections. Each test object includes 10 rendered views and a paired surface point cloud.

Baselines and metrics. For pixel-to-point correspondence, no prior work addresses this task at the object level. As a reference baseline, we repurpose VGGT [4] and input four orthographic views (VGGT requires a known reference frame anchored by the front view) and directly read its pointmap prediction at each query pixel. We evaluate both the original VGGT and VGGT-OBJ (fine-tuned on object data, proposed in [54]). We measure this spatial alignment using LocAcc@k (Localization Accuracy), defined as the mean of $(1 - d_k^\star / d_{\text{norm}}) \times 100\%$. Here, $d_k^\star$ is the minimum Euclidean distance between the ground-truth 3D coordinate and the centers of the top-k retrieved 3D tokens, and $d_{\text{norm}} = \sqrt{3}L$ normalizes the error by the 3D bounding box diagonal. For image-to-shape retrieval, we compare against three leading global alignment models: Uni3D-g [30], OpenShape [29], and ULIP-2 [28], reporting standard Recall@$\{1, 2, 3, 5\}$ and Mean Reciprocal Rank (MRR).

4.1 Pixel-to-point correspondence

To evaluate fine-grained local alignment, we randomly sample up to 20 foreground pixels per view and decode their ground-truth 3D coordinates using the corresponding position maps. For each sampled

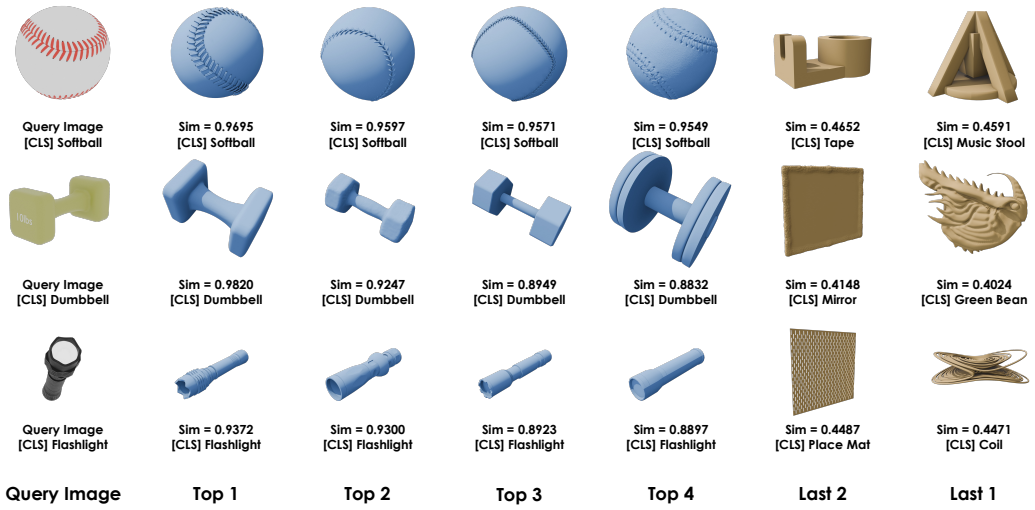

Figure 5: **Image-to-shape retrieval.** Each row shows a query image (left), the top-4 retrieved 3D shapes ranked by global descriptor similarity, and the bottom-2 shapes for contrast. Similarity scores and predicted category labels are annotated below each result.

pixel, we retrieve the 3D token with the highest cosine similarity in the shared local descriptor space. Table 1 reports the LocAcc performance under three view configurations. Our model achieves a LocAcc@1 above 75 across all settings, peaking at 77.37 given four random views.

Surface-distance evaluation. LocAcc measures Euclidean proximity, which can under-penalize matches on opposite sides of a thin structure. We therefore also evaluate localization with SurfAcc, using distances along a graph of surface samples. We construct an undirected eight-nearest-neighbor graph on the 20,480 surface points, with Euclidean edge lengths, and assign each ground-truth query position and retrieved token center to its nearest graph node. For the query set $\mathcal{Q}$ of one object, let $\nu(x)$ denote this assignment, d_G the graph shortest-path distance, and $\mathcal{K}_k(q)$ the top-k tokens ranked by descriptor similarity. We compute

$$d_k^G(q) = \min_{n \in \mathcal{K}_k(q)} d_G\big(\nu(q), \nu(p_n)\big), \qquad \mathrm{SurfAcc@}k = \frac{100}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \left(1 - \frac{d_k^G(q)}{D_G}\right).$$

The per-object normalizer D_G is the largest finite shortest-path distance reached from eight randomly sampled source nodes with a fixed seed, providing an estimate of the graph diameter. If all retrieved candidates are disconnected from a query, its error is set to D_G. We average query scores within each object and then across objects. SurfAcc approximates surface distance using the sampled point cloud: graph edges may connect nearby surface sheets, so it neither gives exact mesh geodesics nor establishes semantic part correctness. Its graph-diameter normalization differs from LocAcc's bounding-box normalization.

For this evaluation, we use the Stage I model at 518×518 and sample up to 20 foreground pixels per view, with an alpha threshold of 0.99 and a four-pixel silhouette erosion. Tango3D takes four random views; the VGGT pointmap reference takes four orthographic views, as required by its evaluation setup. Table 2 summarizes surface-distance localization for Tango3D and the VGGT reference across retrieval ranks. The different view configurations preclude a matched-view comparison, and these Stage I measurements are distinct from the Stage III LocAcc results in Table 1.

Beyond quantitative metrics, Figure 3 demonstrates the versatility of our learned local descriptor space across diverse matching scenarios. In standard 2D-to-3D intra-object matching, a single query pixel accurately localizes the corresponding geometric region on the 3D mesh, confirming robust fine-grained spatial grounding. This capability naturally extends to cross-case 2D-to-3D queries, where a pixel on one object successfully retrieves the semantically equivalent 3D region on an entirely different instance, indicating that the shared space captures strong category-level semantics rather than merely overfitting to exact shapes. Furthermore, reverse 3D-to-2D queries show that a single 3D point correctly grounds back to corresponding pixels across diverse camera poses, verifying multi-view

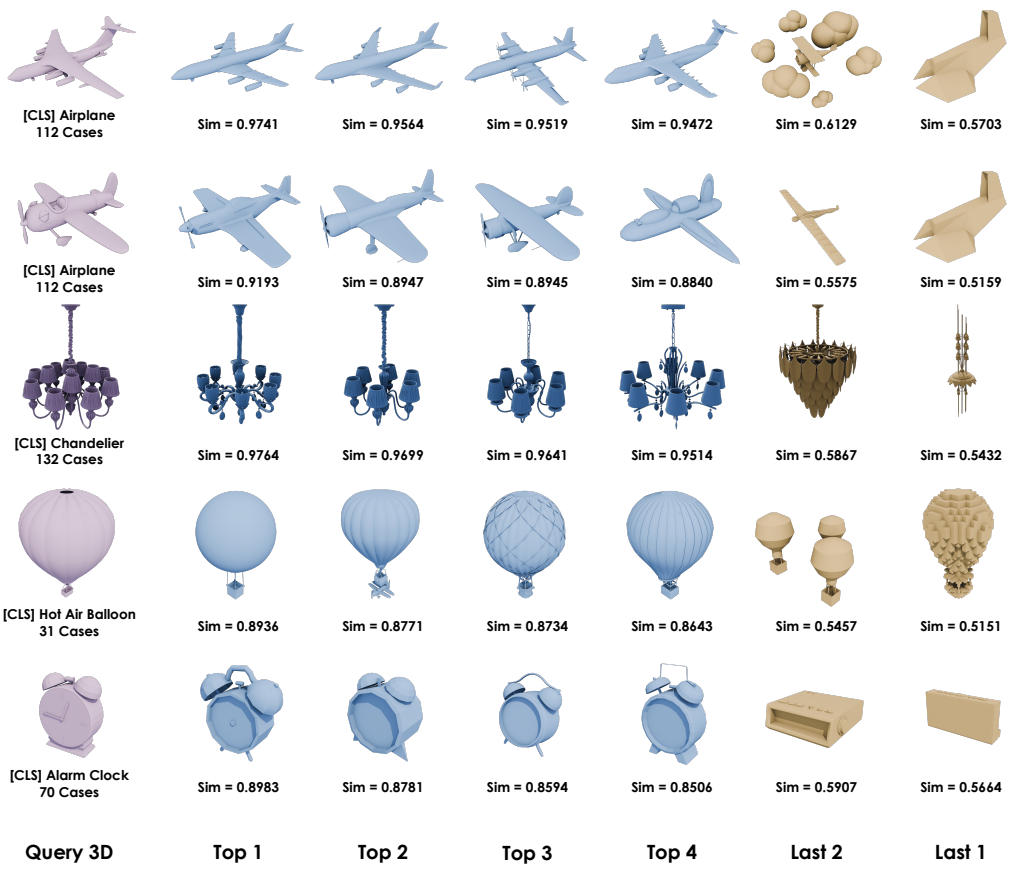

Figure 6: **3D-to-3D shape retrieval.** Each row shows a query shape (left), the top-4 retrieved shapes, and the bottom-2 shapes using only the 3D global descriptor, without any 2D input. Retrieved shapes share fine-grained geometric and topological structure beyond basic category labels.

consistency. Finally, pure 3D-to-3D cross-case queries reveal that the 3D tokens themselves acquire semantic part-level alignment, accurately matching corresponding structural parts across different shapes without relying on any 2D inputs during inference.

Our pixel-to-point alignment naturally extends from individual points to contiguous semantic parts. By coupling a SAM [61] with our local descriptor space, we can transfer an arbitrary 2D part mask onto the 3D mesh without any part-level supervision. As shown in Figure 4, given a user-selected 2D region, our pipeline identifies the corresponding 3D surface region with high fidelity. This demonstrates that the dense alignment learned by Tango3D is coherent enough to support region-level transfer, bridging pixel-level matching and part-level understanding. Details of the transfer pipeline are provided in Appendix C.

4.2 Global retrieval

For category-level image-to-shape retrieval, we use the 2D global descriptor of the query image(s) to search against the 3D global descriptors of all test objects. A match is considered correct if the retrieved shape belongs to the same category as the query. Table 3 compares our performance against the global-only baselines. Across both single-view and multi-view configurations, Tango3D achieves highly competitive retrieval accuracy alongside established models like OpenShape.

Qualitative results in Figure 5 further illustrate this capability. The top-ranked 3D shapes align with the 2D queries not only in broad category semantics but also in fine-grained structural details, demonstrating the benefit of grounding global representations on local geometric features.

Table 4: **Progressive training ablation.** We evaluate the model at the end of each training stage. The improvement in both pixel-to-point localization and image-to-shape retrieval demonstrates that progressively introducing global aggregation and scaling resolution steadily enhances both geometric and semantic representations.

	Local (pixel→point)		Global (image→shape)		
Stage	LocAcc@1	LocAcc@10	R@1	R@10	MRR
Stage I	76.47	86.42	-	-	-
Stage II	76.99	87.35	36.79	71.94	47.91
Stage III	77.37	87.38	39.12	74.16	50.69

Beyond cross-modal matching, our 3D global descriptors naturally support pure shape-to-shape retrieval without any 2D input. As shown in Figure 6, the retrieved shapes share not only category semantics but also fine-grained geometric and topological similarities.

4.3 Ablation studies

To validate our three-stage progressive training strategy, we evaluate the checkpoint at the end of each stage on both tasks (Table 4). Stage I, trained with only the local objective, establishes strong pixel-to-point correspondence but provides no global retrieval capability. After Stage II introduces the global branch, the model gains competitive retrieval performance while the local metrics are largely preserved or improved. Stage III further boosts both tasks through higher-resolution input and strengthened hard-negative mining. The monotonic improvement across all metrics validates the effectiveness of each stage. Complete per-stage results, component ablations, and computational cost analyses are provided in Appendix B.

5 Conclusion and limitation

We presented Tango3D, a foundation model that unifies dense correspondence and global retrieval within a single shared token space. We adopt a geometry-aware 2D backbone as the alignment anchor and encode point clouds into discrete 3D tokens via a pretrained VAE. This design enables fine-grained 2D-3D alignment at the object level. To stabilize the joint optimization of dense and global objectives, we introduce a three-stage progressive training strategy. Experiments confirm that Tango3D achieves strong pixel-to-point localization while remaining competitive with leading global retrieval baselines. Furthermore, the originally non-semantic 3D encoder acquires discriminative semantic structure through cross-modal training alone, showing strong potential for a wide range of downstream tasks.

Our current local alignment operates at a compressed resolution on both sides: VGGT compresses images into patch-level features where each token covers a 14×14 pixel region, and the 3D VAE compresses point clouds into 1024 discrete tokens. Both compression steps discard sub-patch and sub-token spatial details, placing an inherent error on point-level localization accuracy.

Acknowledgments and Disclosure of Funding

This work was supported in part by the General Research Fund (Grant No. 16205526) from Hong Kong Research Grants Council (RGC), in part by the National Natural Science Foundation of China (Grant No. 62372480), in part by Qingdao Key Technology Tackling and Industrialization Demonstration Project (Project No. 25-1-1-gjgg-94-hz), and in part by 2025 Tencent AI Lab Rhino-Bird Focused Research Program (No. TAL26IS01).

References

[1] 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 Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021.

[2] Yifan Zhang and Junhui Hou. Fine-grained image-to-lidar contrastive distillation with visual foundation models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 128396–128429. Curran Associates, Inc., 2024.

[3] Xinjun Li, Wenfei Yang, Jiacheng Deng, Zhixin Cheng, Xu Zhou, and Tianzhu Zhang. Implicit correspondence learning for image-to-point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16922–16931, June 2025.

[4] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

[5] Hanxiao Sun, Mingxin Yang, Shuhui Yang, Zebin He, Xintong Han, Hongbo Fu, Chunchao Guo, and Wenhan Luo. ROAR-3D: routing arbitrary views for high-fidelity 3d generation. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari, editors, *Computer Vision - ECCV 2026 - 19th European Conference, Malmö, Sweden, September 8-12, 2026, Proceedings, Part VII*, volume 17007 of *Lecture Notes in Computer Science*, pages 266–284. Springer, 2026.

[6] Zebin He, Mingxin Yang, Shuhui Yang, Yixuan Tang, Tao Wang, Kaihao Zhang, Guanying Chen, Yuhong Liu, Jie Jiang, Chunchao Guo, and Wenhan Luo. Materialmvp: Illumination-invariant material generation via multi-view pbr diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 26294–26305, 2025.

[7] Zebin He, Mingxin Yang, Shuhui Yang, Zeqiang Lai, Xiangyu Yue, Chunchao Guo, and Wenhan Luo. Materialmvp++: High-quality material generation with neural bump estimation via multi-view pbr diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[8] Huaizhi Qu, Ruichen Zhang, Shuqing Luo, Luchao Qi, Zhihao Zhang, Xiaoming Liu, Roni Sengupta, and Tianlong Chen. Editcast3d: Single-frame-guided 3d editing with video propagation and view selection, 2025.

[9] Peng Li, Suizhi Ma, Jialiang Chen, Yuan Liu, Congyi Zhang, Wei Xue, Wenhan Luo, Alla Sheffer, Wenping Wang, and Yike Guo. CMD: controllable multiview diffusion for 3d editing and progressive generation. In Ginger Alford, Hao (Richard) Zhang, and Adriana Schulz, editors, *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference, SIGGRAPH Conference Papers 2025, Vancouver, BC, Canada, August 10-14, 2025*, pages 81:1–81:10. ACM, 2025.

[10] Peng Li, Rawal Khirodkar, Junxuan Li, Yuan Dong, Chen Cao, Yuan Liu, Wenhan Luo, Yike Guo, and Shunsuke Saito. LUNA: learning universal 3d human animation beyond skinning. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari, editors, *Computer Vision - ECCV 2026 - 19th European Conference, Malmö, Sweden, September 8-12, 2026, Proceedings, Part IV*, volume 17004 of *Lecture Notes in Computer Science*, pages 360–380. Springer, 2026.

[11] Jingke Zhou, Chenhang Ma, Zhizhou Zhong, Mingkai Liu, Zhuang Zhou, Yicheng ji, Binghua Su, Bo Cai, and Xianliang Huang. Think locally, refine globally for memory-efficient 3d reconstruction. *arXiv preprint arXiv:2609.21437*, 2026.

[12] Jiaxin Guo, Tongfan Guan, Wenzhen Dong, Wenzhao Zheng, Wenting Wang, Yue Wang, Yeung Yam, and Yun-Hui Liu. Salon3r: Structure-aware long-term feedforward 3d reconstruction from unposed images. *International Journal of Computer Vision*, 134(8):383, 2026.

[13] Mengfei Li, Xiaoxiao Long, Yixun Liang, Weiyu Li, Yuan Liu, Peng Li, Wenhan Luo, Wenping Wang, and Yike Guo. Multi-view large reconstruction model via geometry-aware positional encoding and attention. *IEEE Trans. Vis. Comput. Graph.*, 31(10):8564–8577, 2025.

[14] Peng Li, Wangguandong Zheng, Yuan Liu, Tao Yu, Yangguang Li, Xingqun Qi, Xiaowei Chi, Siyu Xia, Yan-Pei Cao, Wei Xue, Wenhan Luo, and Yike Guo. Pshuman: Photorealistic single-image 3d human reconstruction using cross-scale multiview diffusion and explicit remeshing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 16008–16018. Computer Vision Foundation / IEEE, 2025.

[15] Yanqin Jiang, Tengfei Wang, Zhengwei Wang, Chenjie Cao, Junta Wu, Wenhan Luo, Weiming Hu, Jin Gao, and Chunchao Guo. Omnix: Any-view and any-time 4d reconstruction via feed-forward trajectory fields. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari, editors, *Computer Vision - ECCV 2026 - 19th European Conference, Malmö, Sweden, September 8-12, 2026, Proceedings, Part LII*, volume 17052 of *Lecture Notes in Computer Science*, pages 313–331. Springer, 2026.

[16] Huaizhi Qu, Hossein Nourkhiz Mahjoub, Vaishnav Tadiparthi, Kwonjoon Lee, and Tianlong Chen. Monovlm: Monocular 3d visual grounding with vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30986–30996, 2026.

[17] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11975–11986, October 2023.

[18] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[19] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 33330–33342. Curran Associates, Inc., 2022.

[20] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *CVPR*, 2024.

[21] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[22] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[23] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[24] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19313–19322, June 2022.

[25] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 604–621. Springer, 2022.

[26] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson W.H. Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22157–22167, October 2023.

[27] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1179–1189, June 2023.

[28] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27091–27101, June 2024.

[29] 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 A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 44860–44879. Curran Associates, Inc., 2023.

[30] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Learning Representations*, volume 2024, pages 46766–46782, 2024.

[31] Huiqun Wang, Yiping Bao, Panwang Pan, Zeming Li, Xiao Liu, Ruijie Yang, and Di Huang. Multi-modal relation distillation for unified 3d representation learning. In *European Conference on Computer Vision*, pages 364–381. Springer, 2024.

[32] Yipeng Gao, Zeyu Wang, Wei-Shi Zheng, Cihang Xie, and Yuyin Zhou. Sculpting holistic 3d representation in contrastive language-image-3d pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22998–23008, June 2024.

[33] Ziyang Zhou, Pinghui Wang, Zi Liang, Haitao Bai, and Ruofei Zhang. Cross-modal 3d representation with multi-view images and point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3728–3739, June 2025.

[34] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[35] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.

[36] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[37] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 368–381. Springer, 2010.

[38] Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE, 2013.

[39] Shaohui Liu, Yidan Gao, Tianyi Zhang, Rémi Pautrat, Johannes L Schönberger, Viktor Larsson, and Marc Pollefeys. Robust incremental structure-from-motion with hybrid features. In *European Conference on Computer Vision*, pages 249–269. Springer, 2024.

[40] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017.

[41] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5038–5047, 2017.

[42] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018.

[43] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. Deepsfm: Structure from motion via deep bundle adjustment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 230–247. Springer, 2020.

[44] Jianyuan Wang, Yiran Zhong, Yuchao Dai, Stan Birchfield, Kaihao Zhang, Nikolai Smolyanskiy, and Hongdong Li. Deep two-view structure-from-motion revisited. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 8953–8962, 2021.

[45] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. *arXiv preprint arXiv:1812.04605*, 2018.

[46] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, 34:16558–16569, 2021.

[47] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. In *ECCV*, 2024.

[48] Cameron Smith, David Charatan, Ayush Tewari, and Vincent Sitzmann. FlowMap: high-quality camera poses, intrinsics, and depth via gradient descent. 2404.15259, 2024.

[49] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. VGGSfM: visual geometry grounded deep structure from motion. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.

[50] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024.

[51] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024.

[52] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21924–21935, June 2025.

[53] Dario Tenore, Daniel Barath, Marc Pollefeys, and Qunjie Zhou. Sem-mast3r: Semantically guided feature matching with mast3r. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 130–139, October 2025.

[54] Jiahao Chang, Chongjie Ye, Yushuang Wu, Yuantao Chen, Yidan Zhang, Zhongjin Luo, Chenghong Li, Yihao Zhi, and Xiaoguang Han. Reconviagen: Towards accurate multi-view 3d object reconstruction via generation. In *The Fourteenth International Conference on Learning Representations*, 2026.

[55] Team Hunyuan3D, Shuhui Yang, Mingxin Yang, Yifei Feng, Xin Huang, Sheng Zhang, Zebin He, Di Luo, Haolin Liu, Yunfei Zhao, et al. Hunyuan3d 2.1: From images to high-fidelity 3d assets with production-ready pbr material. *arXiv preprint arXiv:2506.15442*, 2025.

[56] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. Featured Certification.

[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[58] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 13142–13153, 2023.

[59] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, et al. Objaverse-xl: A universe of 10m+ 3d objects. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 35799–35813, 2023.

[60] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015.

[61] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025.

Table 5: **Per-class object counts for the evaluation benchmark** We detail the distribution of the 3,055 textured objects across 40 ModelNet40 categories.

Category	#	Category	#	Category	#	Category	#
airplane	85	cup	74	laptop	83	sofa	66
bathtub	97	curtain	89	mantel	75	stairs	45
bed	29	desk	59	monitor	97	stool	80
bench	75	door	40	night_stand	98	table	57
bookshelf	70	dresser	83	person	86	tent	78
bottle	70	flower_pot	69	piano	98	toilet	57
bowl	79	glass_box	54	plant	74	tv_stand	97
car	75	guitar	86	radio	93	vase	86
chair	60	keyboard	65	range_hood	84	wardrobe	84
cone	85	lamp	95	sink	93	xbox	85

A Training and evaluation details

A.1 Evaluation benchmark statistics

Table 5 reports the per-class object counts.

Table 6: **Per-stage training configurations.** Detailed hyperparameter evolution across the three progressive training stages.

	Stage I	Stage II	Stage III
Branch / freezing schedule			
VGGT backbone (frozen)	✓	✓	✓
3D VAE encoder trainable	✓	✓	✓
Shared encoders trainable	✓	✓ (low LR)	✓ (low LR)
Local heads trainable	✓	✓ (low LR)	✓
Enable global branch	✗	✓	✓
Global heads trainable	✗	✓	✓
Architecture toggles (global branch)			
DINO cls fusion (2D)	✗	✓	✓
2D input resolution			
VGGT input size	518×518	518×518	1022×1022
Patch grid	37×37	37×37	73×73
Loss weights			
λ_{local}	1.0	0.25	0.50
λ_{global}	0.0	1.0	0.80
λ_{sub}	0.0	0.05	0.05
λ_{sd}	n/a	0.10	0.20
Local loss details			
Loss mode	soft-only	+hard-neg	+hard-neg
Hard-negative top-k	n/a	64	96
Hard-negative weight	n/a	0.25	0.50
Spatial negative radius δ	0.020	0.020	0.015
Distillation temperature			
τ_d	n/a	0.07	0.05
Optimization			
Base learning rate	1×10^{-4}	6×10^{-5}	3×10^{-5}
Shared LR scale	1.0	0.10	0.05
Local LR scale	1.0	0.30	0.50
Global LR scale	n/a	1.0	1.0
Per-GPU batch size	30	25	7
Training length (epochs)	5	3	3

A.2 Per-stage training configurations

Table 6 lists the per-stage hyperparameters. Entries left unchanged across all three stages (MLP widths, layer norms, $[1, 4]$ view sampling, N=20,480 surface points, 512 FPS query budget, 2D Gaussian-blur augmentation, DeepSpeed ZeRO-2 with bf16, AdamW with weight decay 0.01 and gradient clipping at 1.0) are omitted for brevity.

Table 7: **Per-stage pixel-to-point local correspondence.** LocAcc improves steadily across stages, with Stage III benefiting from higher resolution and stronger hard-negative mining.

Stage	Input views	LocAcc@1	LocAcc@2	LocAcc@3	LocAcc@5	LocAcc@10
Stage I	S=1 (random)	76.40	79.55	81.37	83.55	86.32
	S=4 (random4)	76.47	79.62	81.46	83.69	86.42
	S=4 (ortho4)	75.58	78.90	80.86	83.23	86.14
Stage II	S=1 (random)	76.75	79.74	81.60	83.95	87.01
	S=4 (random4)	76.99	80.00	81.88	84.23	87.35
	S=4 (ortho4)	76.01	79.19	81.17	83.64	86.88
Stage III	S=1 (random)	77.43	80.24	82.02	84.23	87.14
	S=4 (random4)	77.37	80.26	82.07	84.37	87.38
	S=4 (ortho4)	76.37	79.49	81.42	83.85	87.00

Table 8: **Per-stage image-to-shape retrieval.** Retrieval performance following the activation of the global branch in Stage II. The subsequent joint optimization in Stage III further enhances Recall and MRR.

Stage	Input views	R@1	R@2	R@3	R@5	R@10	MRR
Stage II	S=1 (random)	33.76	46.22	53.20	60.19	71.25	45.44
	S=4 (ortho4)	35.27	46.92	54.13	59.37	67.05	45.77
	S=4 (random4)	36.79	48.54	55.88	63.33	71.94	47.91
Stage III	S=1 (random)	35.04	45.17	52.50	62.28	71.83	47.17
	S=4 (ortho4)	37.60	49.94	56.23	63.10	71.59	49.67
	S=4 (random4)	39.12	50.64	55.88	63.21	74.16	50.69

B Additional experiments

B.1 Per-stage pixel-to-point local correspondence

Table 7 reports LocAcc at each training stage under all view configurations.

B.2 Per-stage image-to-shape retrieval

Table 8 reports retrieval metrics at each training stage under all view configurations.

Table 9: **Local component ablations at Stage I.** LocAcc@1 on the supplementary local subset, using random views.

Local branch	S=1	S=4
Full	77.2	77.2
w/o bidirectional loss	77.0	76.9
w/o Gaussian weighting	76.9	76.9

Table 10: **Global component ablations at Stage II.** Single- and four-random-view retrieval, with the Full row evaluated at the same stage as the variants.

	S=1		S=4	
Global branch	R@1	MRR	R@1	MRR
Full	33.8	45.4	36.8	47.9
w/o DINO gated fusion	27.9	39.2	32.6	43.9
w/o subset consistency	28.3	39.4	32.6	43.4
w/o semantic distillation	26.8	38.7	30.2	41.7
CLIP distillation teacher	25.0	37.4	29.1	40.6

B.3 Component ablations

Tables 9 and 10 report local variants trained in Stage I and global variants trained in Stage II at 518×518, each compared with its own Full reference. The local evaluation uses a subset of 1,450 objects. The global variants share the Stage I initialization and use category-level retrieval with the query object's own shape excluded from the gallery. These reference rows are specific to their stage and evaluation set.

The local variants remove the reverse-direction objective or replace Gaussian query weights with uniform weights. The global variants disable DINO class-token fusion, view-subset consistency, or semantic relation distillation. The teacher experiment replaces only the distillation teacher with CLIP, retaining DINO fusion and the remaining architecture.

Table 11: **Component cost.** Parameters for overlapping alignment paths and latencies at the timing boundaries described in Appendix B.4.

Component / alignment path	Parameters (M)	Latency (ms)
2D global	10.9	1.2
2D local	9.7	0.7
3D global	11.6	1.1
3D local	7.4	0.8
VGGT backbone	1256.5	224.7
VAE tokenizer	327.7	47.7

Table 12: **3D global encoder cost.** Single H20 GPU, batch size one, 20,480 points, and 30 timed passes. Numerical precision and peak allocated process memory are reported explicitly.

Encoder	Precision	Parameters (M)	Latency (ms)	Peak memory (MB)
Tango3D	bf16	339.3	48.7	6920
OpenShape	fp32	32.3	21.6	573
Uni3D-g	fp32	1017.4	84.0	4391

B.4 Computational cost

Table 11 reports component parameters and latency for batch size one, four 518×518 views, and 20,480 points. Latencies are means over 30 timed passes after five warm-up passes, with GPU synchronization. Peak allocated GPU memory for the full model is approximately 8.6 GB. The alignment-path parameter counts overlap because they include shared encoders and applicable adapters. The 2D global readout is timed on precomputed shared tokens; the other alignment paths include their shared encoders and use cached backbone outputs. Neither parameter counts nor component latencies should be summed as model totals.

Table 12 compares 3D global encoders on an NVIDIA H20 with 20,480 points and batch size one. Tango3D includes its VAE and global alignment path and uses bf16; OpenShape and Uni3D-g use fp32. Peak memory includes resident modules in the benchmark process rather than isolated activation memory. These measurements do not describe Stage III image processing at 1022×1022.

C 2D-to-3D part transfer

Given an input image, we first apply SAM [61] in automatic mode to produce a set of non-overlapping part masks, and select the target part as the smallest mask containing a user-specified click point. The selected pixel-level mask is then mapped onto the VGGT patch grid by marking a patch as active whenever the mask covers a sufficient fraction of its area, and the local 2D descriptors at all active patches are collected. Each of these 2D descriptors is matched against the full set of 3D token descriptors via cosine similarity, keeping only the top-1 3D token per patch and deduplicating tokens matched by multiple patches by retaining the highest-similarity assignment. To ensure spatial coherence, we discard matches with low similarity scores and apply DBSCAN clustering on the remaining 3D coordinates to keep only the dominant spatial cluster. Finally, the filtered 3D token centers are projected onto the mesh surface to identify nearby seed faces, from which a breadth-first-search flood fill along the face-adjacency graph propagates the region label to topologically connected faces; only the largest connected component is retained, yielding the final 3D part region. The entire pipeline operates at inference time and requires no part-level training or annotations.

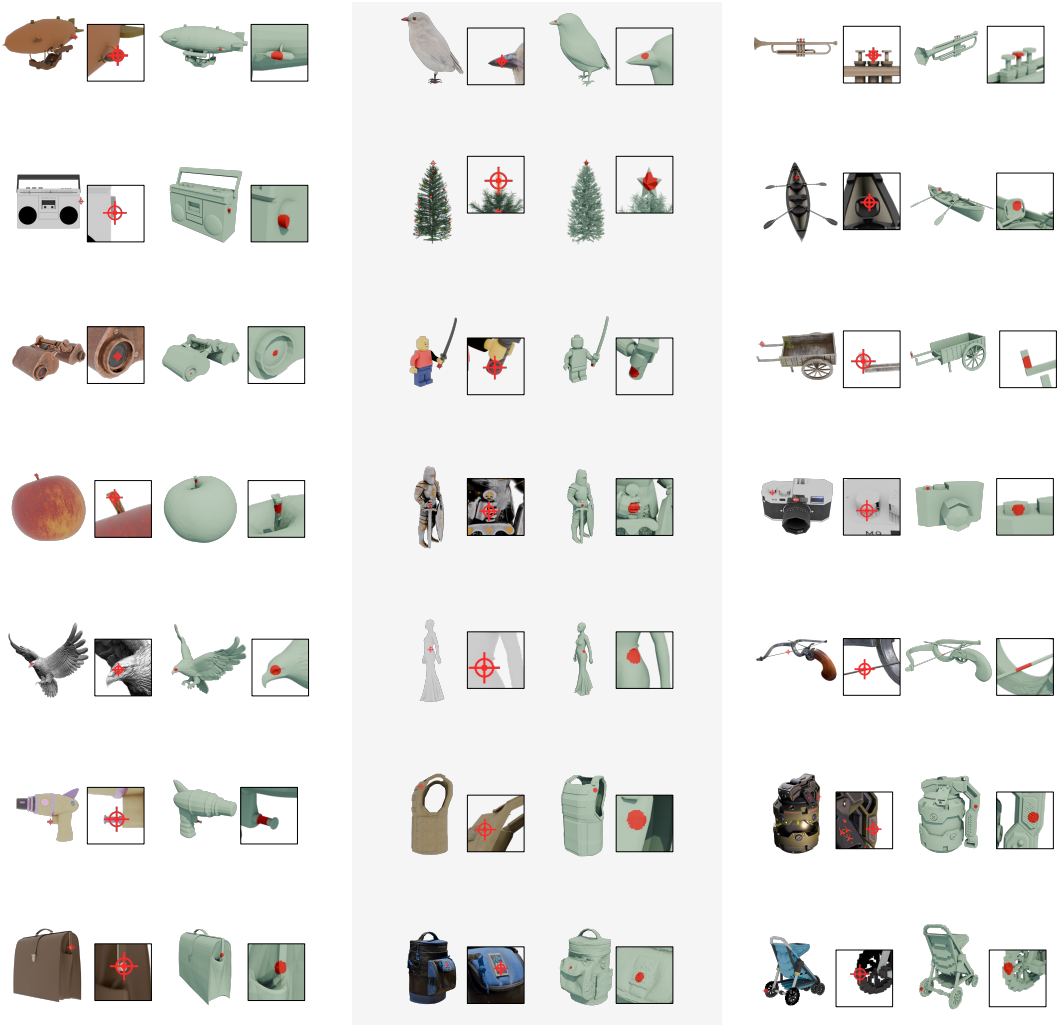

Figure 7: **Additional pixel-to-point correspondence visualizations.** More examples of pixel-to-point matching, complementing Figure 3.

D Additional visualizations

D.1 Additional pixel-to-point correspondence visualizations

Figure 7 shows additional examples complementing Figure 3.

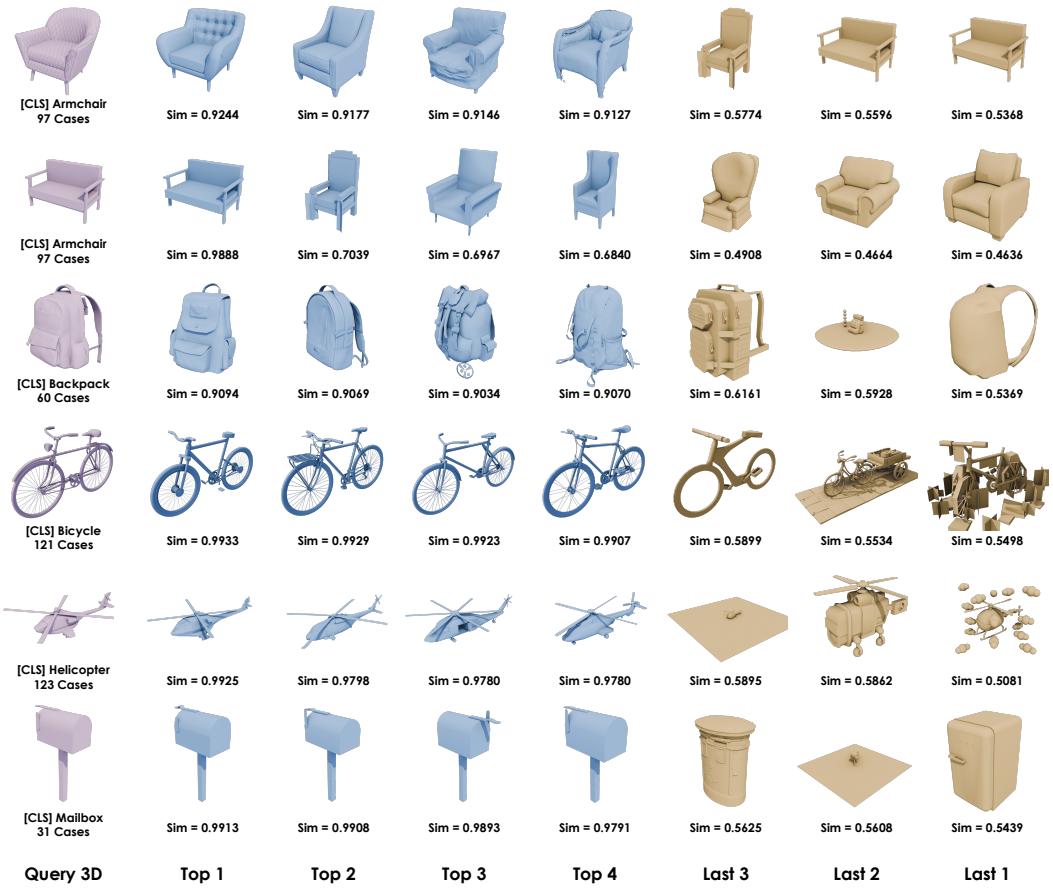

Figure 8: **Additional 3D-to-3D shape retrieval visualizations.** More query-retrieval pairs showing that the 3D global descriptor captures fine-grained geometric similarity.

D.2 Additional shape-to-shape retrieval visualizations

Figure 8 shows additional examples complementing Figure 6.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the scope and contributions, which are directly supported by the experimental results in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: This paper does not include theoretical results. The contributions are architectural and empirical.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide architecture details in Section 3, complete per-stage hyperparameters in Appendix A.2, and training data specifications in Section 4. The model architecture and training procedure are fully described.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code and pretrained model weights will be released upon obtaining institutional approval. The paper provides sufficient architectural and training details for independent reproduction.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details are provided in Section 4 and Appendix A.2. Evaluation metrics and protocols are defined in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The three-stage training pipeline is computationally expensive, making multiple full runs impractical. For evaluation, the local correspondence metric averages over all test objects with sufficient pixel samples per view, reducing evaluation variance.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The compute resources are specified in Appendix A.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm the research conforms with it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is foundational research on 2D-3D representation alignment. We do not foresee direct negative societal impacts. The technology could benefit applications in robotics and 3D content creation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The model proposed does not directly generate content that poses misuse risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All pretrained models (VGGT, Hunyuan3D 2.1, SAM 2) and datasets (Objaverse, Objaverse-XL) are properly cited. Objaverse is released under ODC-By 1.0. All pretrained model weights are used under their respective research-permissive licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We introduce a custom evaluation benchmark comprising 3,055 textured objects across 40 ModelNet40 categories. Per-class statistics and construction details are documented in Section 4 and Appendix A.1. The benchmark will be released alongside the code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not used as a component of the core method in this research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.