Material Magic Wand: Material-Aware Grouping of 3D Parts in Untextured Meshes

Umangi Jain[1*] Vladimir Kim[2] Matheus Gadelha[2] Igor Gilitschenski[1†] Zhiqin Chen[2†]
[1]University of Toronto, [2]Adobe Research

Abstract

We introduce the problem of material-aware part grouping in untextured meshes. Many real-world shapes, such as scales of pinecones or windows of buildings, contain repeated structures that share the same material but exhibit geometric variations. When assigning materials to such meshes, these repeated parts often require piece-by-piece manual identification and selection, which is tedious and time-consuming. To address this, we propose Material Magic Wand, a tool that allows artists to select part groups based on their estimated material properties – when one part is selected, our algorithm automatically retrieves all other parts likely to share the same material. The key component of our approach is a part encoder that generates a material-aware embedding for each 3D part, accounting for both local geometry and global context. We train our model with a supervised contrastive loss that brings embeddings of material-consistent parts closer while separating those of different materials; therefore, part grouping can be achieved by retrieving embeddings that are close to the embedding of the selected part. To benchmark this task, we introduce a curated dataset of 100 shapes with 241 part-level queries. We verify the effectiveness of our method through extensive experiments and demonstrate its practical value in an interactive material assignment application. Project Page: https://umangi-jain.github.io/material-magic-wand.

1. Introduction

Many 3D shapes contain groups of related parts that share a common structural form while varying in their precise geometry. Such families of parts arise naturally in both natural and man-made objects. For instance, a pinecone may contain hundreds of individual scales that share a common form but differ in shape, orientation, and size; similarly, windows in buildings and vehicles are often repeated but non-identical. In a 3D modeling scenario, such repeated parts often share the same material. Yet in current modeling tools, repeated elements must be selected and assigned materials individually. This makes material assignment repetitive and time-consuming, and its cost grows with the number of repeated elements and the mesh complexity.

To address this, we introduce the task of *material-aware part grouping*: it assumes an *untextured* 3D mesh already decomposed into fine-grained parts and a query part as input. The goal is to retrieve other parts within the shape that are likely to share the same material. Despite the importance of material assignment in asset creation, the problem of grouping material-consistent parts under geometric variation remains largely unaddressed.

Existing works tackle related but fundamentally different problems. 3D part segmentation methods [2, 28] aim to partition geometry into semantically meaningful components, whereas our task is higher-level grouping, not initial decomposition. In our setting, the fine-grained mesh parts are already given, because in a typical 3D modeling workflow, an artist-created mesh can be naturally decomposed into parts defined by connected components, which we use as the smallest elements to be grouped. Shape retrieval methods [6, 26] focus on comparing whole shapes across the database, rather than identifying related parts within a single shape. Symmetry detection methods [35] generally rely on exact or near-isometric correspondences, and thus do not capture the breadth of structural similarity that arises in repeated but non-identical elements. Material segmentation methods [10, 24] rely on textures to provide hints on material properties, and achieve material segmentation by adopting foundation image segmentation models, therefore facing challenges when handling small parts due to resolution limits. To our knowledge, no existing method or benchmark evaluates grouping pre-segmented parts based on material consistency.

In this work, we present *Material Magic Wand*. Just like the Magic Wand tool in Photoshop, where a user can easily select pixels of similar colors by clicking on one pixel, Material Magic Wand enables artists to select parts that share the same material by clicking on one part, see Figure 1. And

[*] Work done partially during internship at Adobe
[†] Equal advising

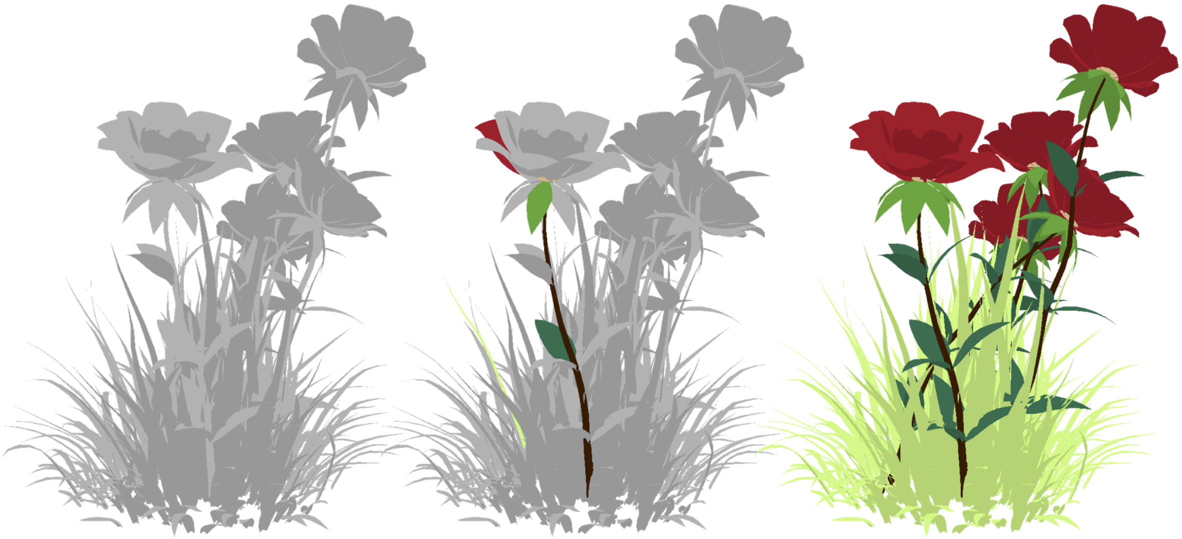

Figure 1. **Material Magic Wand.** Given an untextured 3D mesh (left) with existing part segmentation, which is often obtained by finding connected components of the mesh, a user can apply our tool to select a group of material-consistent parts by clicking on one single representative part. In the middle, we show example selections of petal, base, sepal, stem, leaf, and grass. For each selection, the tool automatically finds all other parts in the shape that are likely to share the same material (right) through geometric and contextual cues, accelerating the material assignment process. Colors may appear darker due to backface shading.

similar to Magic Wand's *Tolerance* parameter to balance between selecting more pixels with less similar colors or fewer pixels with more similar colors, Material Magic Wand can select more parts with less confidence or fewer parts with more confidence by tuning a threshold parameter, enabling fine-grained control and hierarchical selection, see Figure 6. Our tool can significantly speed up the material assignment process, allowing designers to assign materials to hundreds of parts with a single interaction.

Our key insight is to embed each part into an embedding space that encodes material similarity, so that part grouping can be done by retrieving embeddings that are close to the embedding of the query part. Deriving such an embedding space is nontrivial, as both classic geometric descriptors [6] and latest image embedding models such as DINO [34] or SigLIP [44] lack material understanding. We therefore design a part encoder model that learns material-aware embeddings from large-scale collections of 3D shapes. Our encoder generates an embedding code for each 3D part, taking as input multiple images of the part, which are rendered in specific configurations to capture both local geometry and global context. We train the encoder with a supervised contrastive loss to place the embeddings of parts that share the same material close together in the embedding space, while separating the embeddings of parts that differ in their material identity.

To our knowledge, there is no existing 3D dataset for material-aware grouping. However, from the 3D shapes available in Objaverse [8], we are able to curate a dataset of approximately 1.9 million parts across 22,000 meshes with reliable material annotations to supervise our training. Nonetheless, ambiguities naturally arise in material-aware grouping, so that parts can be grouped in various ways under different interpretations of the scene (see Figure 3). To reduce the noise caused by such ambiguities in evaluation, we propose a benchmark of 100 shapes and define 241 part groups for retrieval. Shapes in the benchmark feature repeated but geometrically varied structures, and we manually refine their part-material associations to create clean ground truth for qualitative and quantitative studies. Experiments are conducted on the benchmark, comparing our approach against various baselines, confirming its effectiveness. In addition, we demonstrate Material Magic Wand in an interactive material assignment application, which further exemplifies the appealing capabilities offered by our method.

In summary, our main contributions are:

- We introduce the task of material-aware part grouping on untextured and pre-segmented meshes.
- We curate a sizable 3D dataset for material-aware grouping and a benchmark for evaluation.
- We propose Material Magic Wand to address the material-aware part grouping. Our method enables artists to interactively select material-consistent part groups efficiently.

2. Related Work

We situate our work in relation to prior research in part segmentation, shape retrieval, symmetry and repetition analysis, and material prediction.

Part segmentation aims to decompose a 3D shape into semantically meaningful subparts. The introduction of large-scale part segmentation datasets [32, 52] enabled supervised training of deep neural networks and marked a major step toward scalable part segmentation. More recently, advances in 2D vision foundation models, such as Segment Anything (SAM) [21, 38], CLIP [37], and GLIP [23], have catalyzed the development of open-world 3D segmentation models. Several methods propagate 2D SAM masks to 3D for shape decomposition [12, 18, 43, 53], or train feedforward networks that segment parts using 3D point prompts [22, 28, 58]. Others [1, 2, 19, 27, 29, 45, 51, 56, 57] achieve text-based part segmentation by applying open-world 2D detectors on rendered multi-view images and fusing the predictions. Distinct from part segmentation and component labeling methods [16, 47], our work assumes that parts are already segmented into their finest level, and we focus on grouping the parts based on their potential material assignment.

Shape matching and retrieval aim to retrieve 3D shapes from a database that are similar to a query shape. The query could be sketch-based [46], image-based [49], text-based [39], and 3D shape-based. For 3D shape-based retrieval, global shape descriptors, both classic ones [3, 6, 13] and deep learning-based [9, 26, 48, 50], compare 3D shapes at the object level. These approaches seek inter-shape geometric similarity, while our work targets intra-shape semantic similarity: we retrieve similar parts within a single shape based on their estimated material properties.

Symmetry and structural repetition methods detect symmetry, periodicity, and near-regular structure in 3D geometry. Early work focuses on global or local symmetry detection using geometric hashing or feature correspondences [30, 36]. Subsequent research considers partial and approximate symmetries and near-regular structures [25, 31], as well as repeated elements and their spatial organization [14, 35]. These approaches generally assume repeated elements to be related by rigid or near-isometric transformations, and work best when the intra-group variation is relatively small. In contrast, the parts we aim to group often vary significantly in shape and proportion.

Material segmentation and assignment have been extensively studied in the image domain [11, 41]. In 3D, classical methods [4] define hand-crafted rules to compute surface similarity for matching textured surfaces. Recent approaches [10, 24] perform material-aware segmentation by finetuning SAM and fusing multi-view predictions, but the view resolution in these methods often limits their ability to capture small parts. They also assume textured meshes, where colors provide cues about material properties. For untextured meshes, most approaches [15, 55] generate material maps for whole objects, inheriting the same resolution issues and offering limited editability due to the lack of material segmentation.

3. Method

Given an untextured 3D mesh with existing part segmentation, which is often obtained by finding connected components of the mesh, our method learns a material-aware embedding space for its parts, where each part is encoded by our encoder network through rendered views.

Notations. Assuming a mesh $\mathcal{S}$ is segmented into parts p_i, and associated with a material label y_i. We aim to learn a part encoder network that maps each part p_i into a latent embedding z_i, such that ideally, $z_i = z_j$ if and only if $y_i = y_j$. For part p_i, we define its positive set as $P_i = \{j | j \neq i, y_j = y_i\}$, which contains all other parts in the mesh $\mathcal{S}$ that share the same material with p_i; and its complement set as $A_i = \{j | j \neq i\}$, containing all parts in the mesh except p_i. To encode the parts, each part p_i is represented using three rendered images, defined as $[I_i^{\text{part}}, I_i^{\text{ctx}}, I_i^{\text{full}}]$, corresponding to the *isolated-part* view, *part-with-context* view, and *full-object* view, which will be explained later.

Part Encoder. We adopt a foundation vision model $\mathcal{E}$ to encode the three rendered images individually, initializing the backbone with the DINO-v3 *small* model [42] and fine-tuning the last three transformer blocks. The features from all three images are concatenated to obtain an 1152-d embedding vector x_i, and we use a projection head f to map the features into the contrastive latent space to obtain an 128-d embedding z_i, see Figure 2. The projection head is a two-layer multilayer perceptron (MLP) with ReLU [33] activation. The resulting embedding z_i is ℓ_2-normalized.

$$x_i = \left[\mathcal{E}(I_i^{\text{part}}); \mathcal{E}(I_i^{\text{ctx}}); \mathcal{E}(I_i^{\text{full}})\right]; \; z_i = \frac{f(x_i)}{\|f(x_i)\|_2}.$$

Training objective. We would like for parts with the same material to have nearby embeddings, and for parts with different materials to be far apart. Therefore, we adopt the Supervised Contrastive Loss [17] for learning the embedding space. Our objective is:

$$\mathcal{L} = \mathop{\mathbb{E}}_{\mathcal{S}} \mathop{\mathbb{E}}_{i} \mathop{\mathbb{E}}_{j \in P_i} \left(-\log \frac{\exp(z_i \cdot z_j / \tau)}{\sum\limits_{a \in A_i} \exp(z_i \cdot z_a / \tau)} \right),$$

where τ is the temperature parameter and $\cdot$ is dot product. In practice, we put all parts in a training batch except for p_i itself to A_i to stablize training.

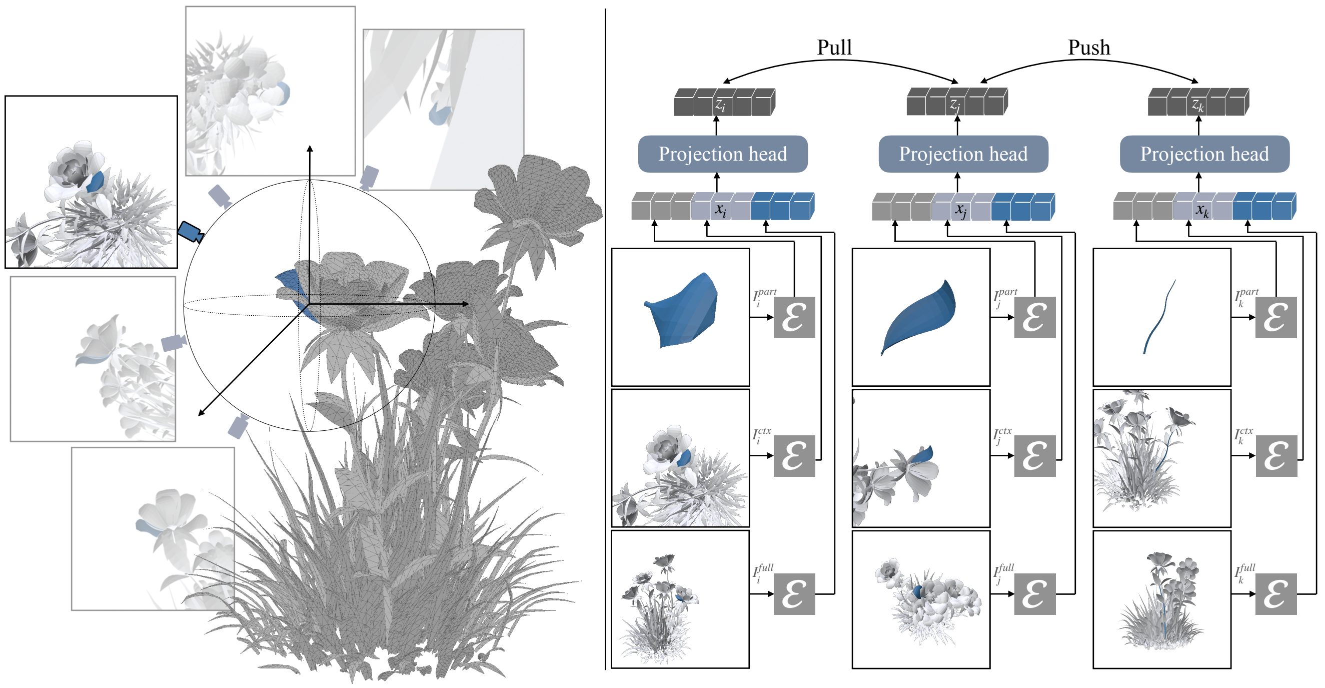

Figure 2. **Method Overview.** *Left:* Our view selection process renders each part with nearby context from multiple viewpoints sampled randomly over a hemisphere. We choose the one with minimal occlusion, I^{ctx}, and use the same viewpoint to render the part in isolation, I^{part}. I^{full} captures the entire mesh. We highlight the part with a different color from the rest of the mesh. *Right:* For each part, its corresponding images are passed through an encoder and their embeddings are concatenated. During training, embeddings of parts with the same material are pulled together, while those with different materials are pushed apart.

Inference. We find that instead of using the compressed embedding z_i, using the original higher-dimensional embedding x_i leads to consistently better performance (see Table 2). Therefore, we compute embedding x_i for each part p_i in the 3D mesh. We define the similarity metric between two parts p_i and p_j as the negative ℓ_1 distance between their embeddings: $s(p_i, p_j) = -\|x_i - x_j\|_1$. Then, given a query part p_i, we can select a group of parts $\{p_j | s(p_i, p_j) \leq \lambda\}$, where a higher threshold λ includes more parts, providing a looser grouping, and a lower λ restricts the selection to the most similar parts.

View selection. To encode part-level representations, we render each part under three contexts: *isolated-part*, *part-with-context*, and *full-object*, see Figure 2. In the isolated part view I_i^{part}, only the part p_i is shown in a highlighted color and other parts are hidden. In the part-with-context view I_i^{ctx}, we render part p_i as well as other parts of the mesh, while only p_i has highlighted color. We adjust the camera distance, so that in the rendered view, p_i occupies around 25% of space in one image dimension. The full-object view I_i^{full} renders the entire mesh, while p_i is also highlighted. For I_i^{ctx}, we render multiple views from different viewpoints, and select the best view following the perceptual preference for maximizing visible surface area [40]. Specifically, we sample 16 candidate camera positions on a hemisphere surrounding the part and choose the one view that maximizes the visible area of the part in the rendered image. If the part is heavily occluded under all candidate views, we reduce the context region by zooming the camera towards the part. For I_i^{part}, we use the same viewpoint as I_i^{ctx} but with camera zoomed-in on the part. For I_i^{full}, we place the camera along the direction from the object center to the part centroid. Our method targets untextured meshes, so we remove all material and texture before rendering.

Part deduplication. Meshes can have duplicated parts - parts that are essentially identical but have undergone transformations resulting in different sizes and orientations. Identifying duplicated parts from rigid transformations is relatively easy with a histogram matching algorithm [3, 5]. Therefore, we run the algorithm to group identical parts in each mesh, and randomly select one exemplar part in each group to compute the embedding for all the parts in the group. This deduplication step reduces computational cost.

Training dataset. Recent datasets such as Material3D [15] and DreamMat [55] provide textured 3D meshes for training material prediction and generation models. However, these datasets assign materials at the surface level, not part level, therefore they cannot be directly used for part-level grouping. Our task requires a large-scale 3D dataset consisting of shapes with per-part material ID, where the same material ID is shared by different parts. Therefore, we curate a subset of 22,000 meshes with material assignment

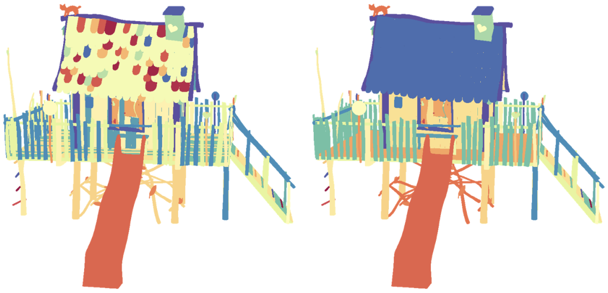

Figure 3. **Ambiguities in raw material IDs.** Using meshes from Objaverse directly for material grouping can introduce ambiguity in the testing benchmark due to artistic intent, noisy labeling, or coarse material assignment. For example, some shingles on the roof exhibit mixed materials, and fence or wall slats alternate between different material labels (left). To reduce such inconsistencies, we manually refine the material annotations to make the material assignment more uniform and fine-grained (right).

from Objaverse [7]. Since Objaverse does not provide fine-grained part segmentation, we extract parts using connected components, after performing vertex merging on the mesh to avoid fragmented components. We assign each part a material ID by taking the majority material label over the faces it contains. Material IDs are defined independently for each mesh. Meshes in Objaverse also present several challenges, especially the imbalanced material distribution, e.g., 99% of the materials are only used once on one part, or one material is used on 99% of the parts. We apply data balancing strategies to mitigate both within and across mesh imbalance, resulting in a more uniform distribution over both meshes and material IDs. See appendix for more details.

Implementation details. We use an OpenGL-based renderer for generating the training data at scale. All images are rendered at 512 × 512 resolution. We train the model using the Adam optimizer [20] with learning rate 1×10^{-5}. The model is trained for 20,000 steps with a batch size of 256. We observe marginal difference when scaling to a larger encoder backbone; see ablation study in Table 2.

4. Experiments

Evaluation Benchmark. Existing datasets do not provide part-level groupings within a single mesh that reflect fine-grained material-level similarity. Therefore, we construct a testing benchmark dataset that features repeated but geometrically varied structures in each shape. The material assignments in Objaverse can exhibit inconsistencies: materials may be shared across parts that differ functionally and in appearance, while in some cases, artistic intent can introduce additional ambiguity by assigning different materials to the same type of parts. For instance, in Figure 3, roof shingles and fence slats have a variety of material types. Such cases introduce ambiguity that would lead to unreliable evaluation. To obtain consistent ground-truth, we manually refine the material assignments for 100 meshes from Objaverse in Blender, thereby resolving ambiguous cases. This results in 241 query parts with well-defined ground-truth retrieval sets, forming our benchmark for both qualitative and quantitative evaluation.

The benchmark spans a wide range of mesh complexity. Across the 100 meshes, the number of parts per mesh varies substantially (median = 265; IQR = 1,144; range = 16–40,086), reflecting large differences in granularity. The 241 ground-truth part groups also exhibit significant variations in size (median 20; IQR = 72; range = 2-32,267). For evaluation, an arbitrarily chosen part is designated as the query for the group, and all other parts in the same group are considered equally relevant positive matches.

Metrics. For each query, we evaluate retrieval as a ranked list over all other parts within the same mesh. We report standard retrieval metrics: area under the precision–recall curve (AUC PR), R-Precision (R-Prec), mean average precision (mAP), and Recall@k. All metrics are reported using macro-averaging (each query weighted equally) to prevent queries with large ground-truth sets from dominating the scores. To assess grouping performance, we report the average F1 score, computed between the predicted and ground-truth groups for a given similarity threshold λ. The threshold λ is selected separately for each method to ensure fairness, using the value that maximizes F1 on a small held-out validation split (5 meshes, 13 queries).

Comparison baselines. We compare against three categories of baselines. (*i*) **Histogram matching** [5]: a non-learning geometry-based baseline that uses statistics of the given mesh as features; we also use this algorithm with low tolerance threshold for part deduplication. (*ii*) **Vision foundation model embeddings**: we render each part (isolated, with context, and full-object) and extract embeddings using DINO-v2 [34], DINO-v3 [42], and SigLIP [54]; results are reported for the best-performing backbone (we find minimal variance across these models; see Appendix). (*iii*) **PartField** [28]: a hierarchical 3D part segmentation model that produces part-level embeddings; we test the released model directly on our benchmark. We apply part deduplication to all baselines for a fair comparison.

Quantitative results. Table 1 compares our method against geometric, vision foundation, and part-feature baselines across retrieval and grouping metrics. Our model achieves the highest performance on all measures, outperforming the strongest baseline (DINO-v3 *small*) by a margin of +8.6% in AUC for retrieval and +16.6% in F1 score for grouping. Figure 4 shows the precision-recall curve, where our method consistently maintains higher precision across the full recall range. For computing the precision–recall curve,

Table 1. Comparison of geometry-based, vision-embedding, and segmentation-based baselines on our material-aware grouping task. All metrics are macro-averaged and reported in percentage. Our method achieves consistent improvements across all evaluation metrics.

Method	AUC PR	R-Prec	mAP	Recall@K				F1
				R@5	R@10	R@20	R@100	
Histogram Matching [5]	26.85	25.45	30.71	4.55	8.89	15.97	45.64	23.84
SigLIP-v2 [44]	62.83	56.02	60.58	22.88	31.20	40.72	68.44	39.44
PartField [28]	75.30	67.74	70.52	26.88	37.12	47.92	71.31	56.57
DINO-v3 *small*† [42]	81.14	78.32	83.49	34.57	45.49	56.63	82.18	59.36
Ours	**89.74**	**88.33**	**91.70**	**37.99**	**50.98**	**62.79**	**83.67**	**75.94**

†We evaluate several DINO variants (v2/v3 at different scales) and report the one achieving the highest AUC.

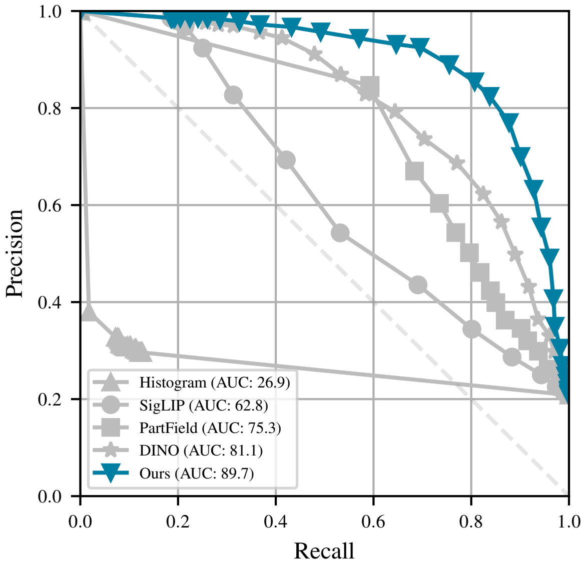

Figure 4. **Precision–Recall curve.** We sweep the similarity threshold to evaluate retrieval performance. Our method consistently maintains higher precision across all recall levels, achieving the highest AUC (89.7), followed by DINO (81.1).

Table 2. **Ablation results.** We examine the contribution of our design choices. Removing any component reduces performance, with the full model achieving the highest scores across all metrics.

Method	AUC	R-Prec	mAP	R@20
w/o *isolated part*	86.90	84.97	88.53	61.11
w/o *part-with-context*	87.30	85.37	88.91	61.09
w/o *full-object*	88.89	87.72	90.75	62.36
Only *isolated part*	86.18	81.88	85.88	59.87
Init from Dino-v2 *L*	86.51	84.58	88.61	61.13
Random initialization	78.54	74.52	79.62	56.25
Finetune last 5 blocks	89.52	88.22	91.35	62.52
Retrieval with z	87.45	87.58	90.90	62.34
w/o data rebalancing	78.41	76.22	80.39	55.83
Ours	**89.74**	**88.33**	**91.70**	**62.79**

we sweep thresholds across the full range of similarity scores between the minimum and maximum values. To ensure uniform coverage under non-linear score distributions, we use quantile-based threshold sampling. The geometric baseline (Histogram Matching) exhibits a steep precision drop as recall increases, indicating that purely shape-based descriptors are brittle and only effective for near-duplicate parts. The weaker performance of PartField can be attributed to its different training objective: hierarchical part segmentation, which does not align with the goal of learning material-consistent embeddings.

Qualitative Results. Figure 5 presents a visual comparison across methods. While DINO and SigLIP retrieve parts with similar visual appearance, they often miss structurally related components, for example, caster wheels in the wheelchair, vertical wire meshes in the shopping cart, or the rear windscreen in the car. The embeddings from PartField, trained for hierarchical shape decomposition, tend to miss geometric correspondences. Our method incorporates contextual cues to retrieve material-consistent parts, such as the jewels on the crown that appear under the same spatial context, while avoiding visually similar but contextually different ones retrieved by DINO and PartField.

Effect of changing threshold. Material Magic Wand offers a grouping tolerance parameter that can be controlled via the similarity threshold λ. Increasing λ progressively expands the retrieved set from strictly matching parts to broader, structurally related regions. As shown in Figure 6, our model exhibits smooth transitions: for instance, in the bed example, retrieval expands from the identical pillow to geometrically similar pillows to all pillows; likewise, in the pot example, retrieval grows from local leaves to the full set of leaves in the shape. In contrast, DINO and PartField start retrieving unrelated components as threshold increases. PartField also lacks fine-grained control, e.g., grouping most parts even at very low thresholds.

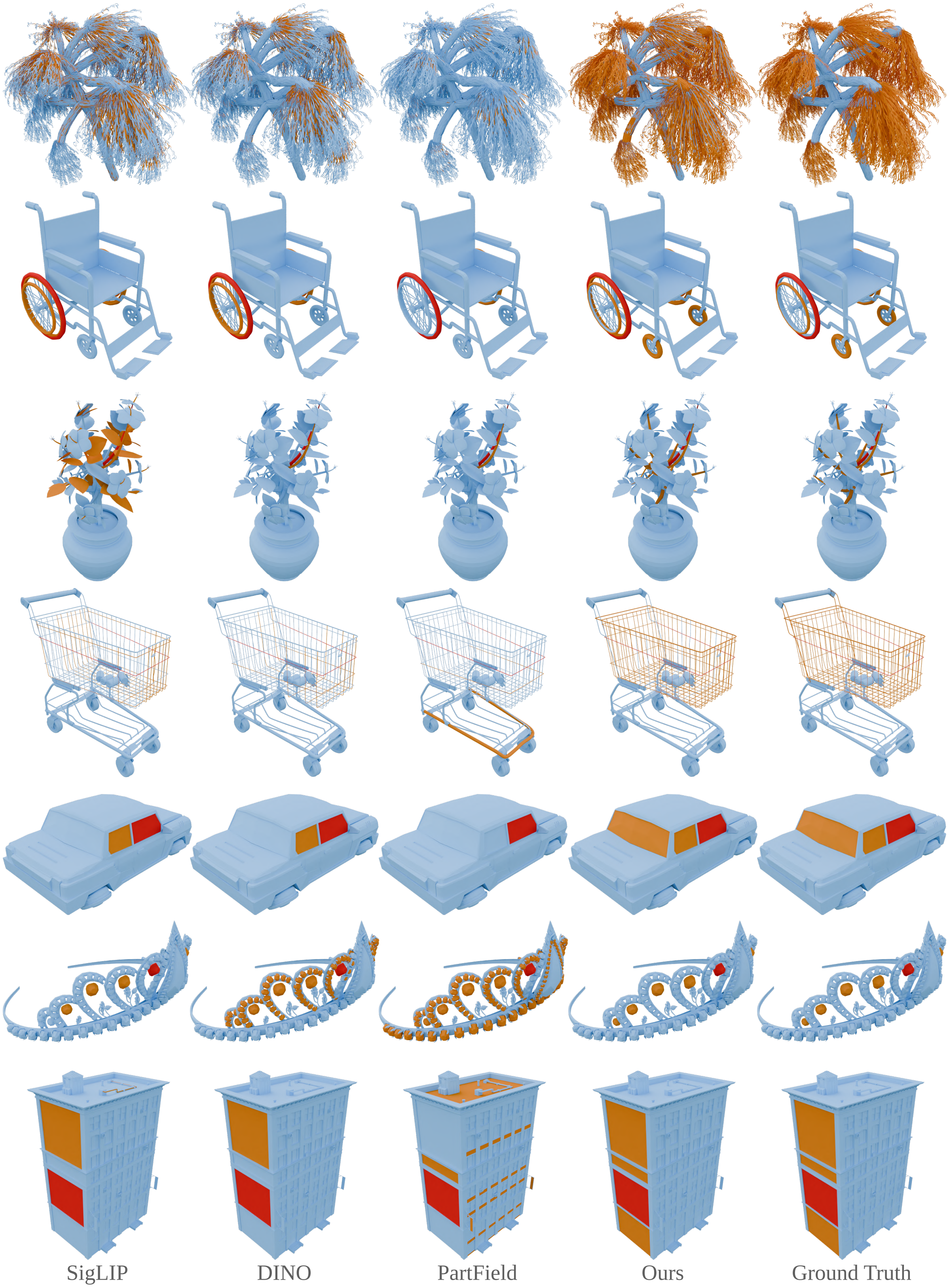

Figure 5. **Qualitative comparison.** For each mesh, the red part denotes the query, and orange parts indicate the retrieved matches. Our method retrieves components that are both geometrically and contextually similar with the query, while baselines often miss structurally related parts or include visually similar but contextually incorrect ones.

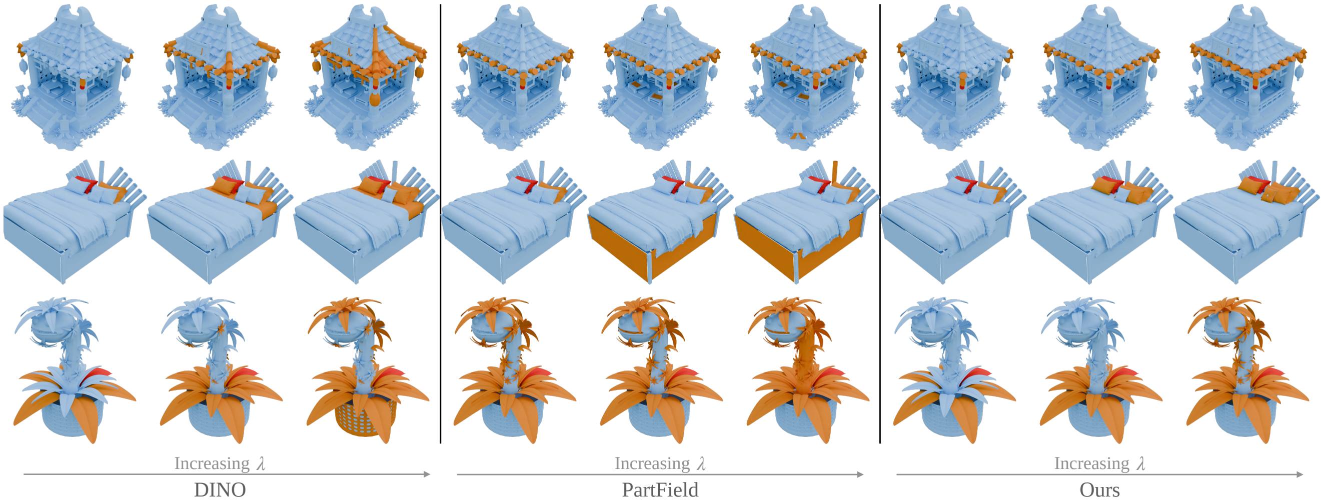

Figure 6. **Effect of changing the grouping tolerance parameter.** The similarity threshold λ controls the cut-off distance in the parts' embedding space for part retrieval. With our method (right), at low values of λ, only highly similar parts are selected; increasing λ gradually expands the selection to include less similar, yet related parts. In contrast, baselines exhibit less stable behavior by either retrieving unrelated components at higher thresholds or lacking fine-grained control at lower thresholds.

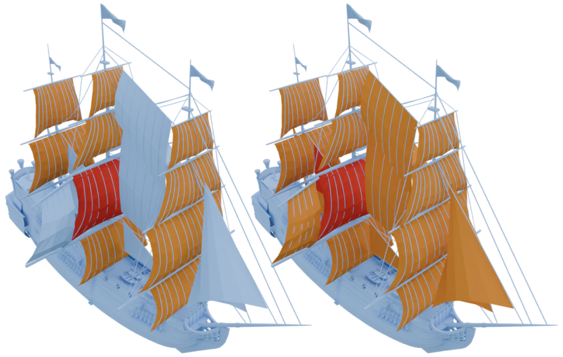

Figure 7. **Effect of multiple queries.** Additional query parts improve retrieval by capturing a more complete set of desired components. The initial query (left) retrieves only a subset of sails, while adding an additional example (right) helps recover the rest.

Ablation study. We analyze the contribution from each design choice in Table 2. Removing any of the three rendered views (*isolated part*, *part-with-context*, or *full-object*) reduces performance, with the largest drop from removing the isolated part. To emulate a shape-retrieval-only setup, we also evaluate using only the isolated part rendering and observe drop in performance, highlighting the importance of contextual cues. We test performance using pre-projection x_i and post-projection embedding z_i and find that the former performs better for retrieval. Increasing the model size (to DINO large) or finetuning additional layers provides negligible gains, likely limited by the noisy supervision in Objaverse. Training from scratch or omitting data rebalancing sharply reduces performance, emphasizing the importance of strong initialization and balanced supervision (all variants are trained for same number of steps for fairness).

Multiple Clicks. While adjusting the tolerance provides control over the degree of similarity, part grouping can remain ambiguous depending on user intent or multiple valid choices. Our method can be applied iteratively, where providing additional query parts refines the retrieval by improving coverage of the desired components. As shown in Figure 7, the initial selection retrieves only geometrically similar sails, whereas adding another example also retrieves the jibs. We show our tool in an interactive material assignment application in the supplementary video.

Limitations. Our method produces a deterministic ranking of parts, but grouping can involve multiple valid choices which we do not explicitly model. In meshes with significant self-occlusion or many highly obstructed parts, our rendering-based view selection can struggle to produce clear contextual cues. See appendix for more details.

5. Conclusion

We introduced Material Magic Wand, a framework for material-aware part grouping that learns to associate geometrically and contextually similar components within a mesh. Through contrastive training on large-scale data, our method enables robust retrieval and grouping, facilitating fast and more consistent material assignment (see Appendix for runtime). We also create a test benchmark for evaluating this task. Our experiments demonstrate significant improvements over geometric and vision-based baselines, and qualitative results highlight its applicability to 3D modeling pipelines.

References

[1] Ahmed Abdelreheem, Abdelrahman Eldesokey, Maks Ovsjanikov, and Peter Wonka. Zero-shot 3d shape correspondence. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3

[2] Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. Satr: Zero-shot semantic segmentation of 3d shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15166–15179, 2023. 1, 3

[3] Mihael Ankerst, Gabi Kastenmüller, Hans-Peter Kriegel, and Thomas Seidl. 3d shape histograms for similarity search and classification in spatial databases. In *International symposium on spatial databases*, pages 207–226. Springer, 1999. 3, 4

[4] Matthäus G Chajdas, Sylvain Lefebvre, and Marc Stamminger. Assisted texture assignment. In *Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games*, pages 173–179, 2010. 3

[5] Siddhartha Chaudhuri. Thea. 4, 5, 6

[6] Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen, and Ming Ouhyoung. On visual similarity based 3d model retrieval. In *Computer graphics forum*, pages 223–232. Wiley Online Library, 2003. 1, 2, 3

[7] 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 *CVPR*, pages 13142–13153, 2023. 5

[8] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2

[9] Yi Fang, Jin Xie, Guoxian Dai, Meng Wang, Fan Zhu, Tiantian Xu, and Edward Wong. 3d deep shape descriptor. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2319–2328, 2015. 3

[10] Michael Fischer, Iliyan Georgiev, Thibault Groueix, Vladimir G Kim, Tobias Ritschel, and Valentin Deschaintre. Sama: Material-aware 3d selection and segmentation. *arXiv preprint arXiv:2411.19322*, 2024. 1, 3

[11] Julia Guerrero-Viu, Michael Fischer, Iliyan Georgiev, Elena Garces, Diego Gutierrez, Belen Masia, and Valentin Deschaintre. Fine-grained spatially varying material selection in images. *arXiv preprint arXiv:2506.09023*, 2025. 3

[12] Haodi He, Colton Stearns, Adam W Harley, and Leonidas J Guibas. View-consistent hierarchical 3d segmentation using ultrametric feature fields. In *European Conference on Computer Vision*, pages 268–286. Springer, 2024. 3

[13] Masaki Hilaga, Yoshihisa Shinagawa, Taku Kohmura, and Tosiyasu L Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212, 2001. 3

[14] Qixing Huang, Leonidas J Guibas, and Niloy J Mitra. Near-regular structure discovery using linear programming. *ACM Transactions on Graphics (TOG)*, 33(3):1–17, 2014. 3

[15] Xin Huang, Tengfei Wang, Ziwei Liu, and Qing Wang. Material anything: Generating materials for any 3d object via diffusion. In *CVPR*, pages 26556–26565, 2025. 3, 4

[16] R Kenny Jones, Aalia Habib, Rana Hanocka, and Daniel Ritchie. The neurally-guided shape parser: Grammar-based labeling of 3d shape regions with approximate inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11614–11623, 2022. 3

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *NeurIPS*, 33:18661–18673, 2020. 3

[18] Chung Min Kim, Mingxuan Wu, Justin Kerr, Ken Goldberg, Matthew Tancik, and Angjoo Kanazawa. Garfield: Group anything with radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21530–21539, 2024. 3

[19] Hyunjin Kim and Minhyuk Sung. Partstad: 2d-to-3d part segmentation task adaptation. In *European Conference on Computer Vision*, pages 422–439. Springer, 2024. 3

[20] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[21] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 3

[22] Itai Lang, Fei Xu, Dale Decatur, Sudarshan Babu, and Rana Hanocka. iseg: Interactive 3d segmentation via interactive attention. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 3

[23] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10975, 2022. 3

[24] Zeyu Li, Ruitong Gan, Chuanchen Luo, Yuxi Wang, Jiaheng Liu, Ziwei Zhu, Qing Li, Xucheng Yin, Man Zhang, Zhaoxiang Zhang, et al. Materialseg3d: Segmenting dense materials from 2d priors for 3d assets. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 370–379, 2024. 1, 3

[25] Yaron Lipman, Xiaobai Chen, Ingrid Daubechies, and Thomas Funkhouser. Symmetry factored embedding and distance. In *ACM SIGGRAPH 2010 papers*, pages 1–12. 2010. 3

[26] 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. *Advances in neural information processing systems*, 36:44860–44879, 2023. 1, 3

[27] Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. Partslip: Low-shot part

segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21736–21746, 2023. 3

[28] Minghua Liu, Mikaela Angelina Uy, Donglai Xiang, Hao Su, Sanja Fidler, Nicholas Sharp, and Jun Gao. Partfield: Learning 3d feature fields for part segmentation and beyond. In *CVPR*, pages 9704–9715, 2025. 1, 3, 5, 6

[29] Ziqi Ma, Yisong Yue, and Georgia Gkioxari. Find any part in 3d. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7818–7827, 2025. 3

[30] Niloy J Mitra, Leonidas J Guibas, and Mark Pauly. Partial and approximate symmetry detection for 3d geometry. *ACM Transactions on Graphics (ToG)*, 25(3):560–568, 2006. 3

[31] Niloy J Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. Symmetry in 3d geometry: Extraction and applications. In *Computer graphics forum*, pages 1–23. Wiley Online Library, 2013. 3

[32] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *CVPR*, pages 909–918, 2019. 3

[33] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010. 3

[34] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 2, 5

[35] Mark Pauly, Niloy J Mitra, Johannes Wallner, Helmut Pottmann, and Leonidas J Guibas. Discovering structural regularity in 3d geometry. In *ACM SIGGRAPH 2008 papers*, pages 1–11. 2008. 1, 3

[36] Joshua Podolak, Philip Shilane, Aleksey Golovinskiy, Szymon Rusinkiewicz, and Thomas Funkhouser. A planar-reflective symmetry transform for 3d shapes. In *ACM SIGGRAPH 2006 Papers*, pages 549–559. 2006. 3

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PmLR, 2021. 3

[38] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025. 3

[39] Yue Ruan, Han-Hung Lee, Yiming Zhang, Ke Zhang, and Angel X Chang. Tricolo: Trimodal contrastive loss for text to shape retrieval. In *WACV*, pages 5815–5825, 2024. 3

[40] Adrian Secord, Jingwan Lu, Adam Finkelstein, Manish Singh, and Andrew Nealen. Perceptual models of viewpoint preference. *ACM Transactions on Graphics (TOG)*, 30(5): 1–12, 2011. 4

[41] Prafull Sharma, Julien Philip, Michaël Gharbi, Bill Freeman, Fredo Durand, and Valentin Deschaintre. Materialistic: Selecting similar materials in images. *ACM Transactions on Graphics*, 42(4), 2023. 3

[42] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 3, 5, 6

[43] George Tang, William Zhao, Logan Ford, David Benhaim, and Paul Zhang. Segment any mesh: Zero-shot mesh part segmentation via lifting segment anything 2 to 3d. *arXiv e-prints*, pages arXiv–2408, 2024. 3

[44] 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. 2, 6

[45] Ardian Umam, Cheng-Kun Yang, Min-Hung Chen, Jen-Hui Chuang, and Yen-Yu Lin. Partdistill: 3d shape part segmentation by vision-language model distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3470–3479, 2024. 3

[46] Fang Wang, Le Kang, and Yi Li. Sketch-based 3d shape retrieval using convolutional neural networks. In *CVPR*, pages 1875–1883, 2015. 3

[47] Xiaogang Wang, Bin Zhou, Haiyue Fang, Xiaowu Chen, Qinping Zhao, and Kai Xu. Learning to group and label fine-grained shape components. *ACM Transactions on Graphics (TOG)*, 37(6):1–14, 2018. 3

[48] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21026–21036, 2025. 3

[49] Qirui Wu, Daniel Ritchie, Manolis Savva, and Angel X Chang. Generalizing single-view 3d shape retrieval to occlusions and unseen objects. In *2024 International Conference on 3D Vision (3DV)*, pages 893–902. IEEE, 2024. 3

[50] Jin Xie, Yi Fang, Fan Zhu, and Edward Wong. Deepshape: Deep learned shape descriptor for 3d shape matching and retrieval. In *CVPR*, pages 1275–1283, 2015. 3

[51] Yuheng Xue, Nenglun Chen, Jun Liu, and Wenyun Sun. Zerops: High-quality cross-modal knowledge transfer for zero-shot 3d part segmentation. In *2025 International Conference on 3D Vision (3DV)*, pages 1328–1339. IEEE, 2025. 3

[52] Li Yi, Vladimir G Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG)*, 35(6):1–12, 2016. 3

[53] Haiyang Ying, Yixuan Yin, Jinzhi Zhang, Fan Wang, Tao Yu, Ruqi Huang, and Lu Fang. Omniseg3d: Omniversal 3d segmentation via hierarchical contrastive learning. In *Pro-*

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20612–20622, 2024. 3

[54] 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*, pages 11975–11986, 2023. 5

[55] Yuqing Zhang, Yuan Liu, Zhiyu Xie, Lei Yang, Zhongyuan Liu, Mengzhou Yang, Runze Zhang, Qilong Kou, Cheng Lin, Wenping Wang, et al. Dreammat: High-quality pbr material generation with geometry-and light-aware diffusion models. *ACM Transactions on Graphics (TOG)*, 43(4):1–18, 2024. 3, 4

[56] Ziming Zhong, Yanyu Xu, Jing Li, Jiale Xu, Zhengxin Li, Chaohui Yu, and Shenghua Gao. Meshsegmenter: Zero-shot mesh semantic segmentation via texture synthesis. In *ECCV*, pages 182–199. Springer, 2024. 3

[57] Yuchen Zhou, Jiayuan Gu, Xuanlin Li, Minghua Liu, Yunhao Fang, and Hao Su. Partslip++: Enhancing low-shot 3d part segmentation via multi-view instance segmentation and maximum likelihood estimation. *arXiv preprint arXiv:2312.03015*, 2023. 3

[58] Yuchen Zhou, Jiayuan Gu, Tung Yen Chiang, Fanbo Xiang, and Hao Su. Point-sam: Promptable 3d segmentation model for point clouds. In *ICLR*, 2025. 3