BREPLIFT: B-REP RECONSTRUCTION WITH SHARED 3D FOUNDATION PRIORS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present BRepLift, a framework for reconstructing boundary representations (B-Reps) from 3D meshes using shared 3D foundation priors. Its core innovation is the adaptation of shared 3D foundation features across all stages of B-Rep reconstruction, enabling more accurate and consistent primitive identification, geometry recovery, and topology prediction. To align pretrained features with B-Rep primitives, we adapt them with a face-level contrastive objective and use cluster-guided aggregation to segment the mesh into individual B-Rep face masks. The predicted face masks then organize the same shared 3D foundation features into intra-face and cross-face contexts for reconstructing surfaces, edges, and their connectivity. Parametric fitting and geometric post-processing assemble these predictions into the final B-Rep. Experiments on three test sets spanning diverse geometries and topologies show that BRepLift improves face, edge, and vertex F1 scores and B-Rep validity over DualBrep on all three sets. In particular, the percentage of valid reconstructions increases from 10.60% to 22.14% on ABC-ParseNet and from 17.34% to 29.46% on our curated test set.

1 INTRODUCTION

We study the problem of reconstructing CAD models in boundary representation (B-Rep) from 3D meshes. B-Rep serves as the de facto standard in commercial CAD systems and forms the representational basis for design, simulation (Schulz et al., 2017; Hafner et al., 2019; Tian et al., 2022), and manufacturing (Uchytil & Storti, 2023; Brown, 1981). Recovering such a representation from 3D observations remains a fundamental challenge in reverse engineering, as it demands synthesizing precise parametric geometry from unstructured input while guaranteeing global topological validity.

Existing approaches broadly follow two paradigms. Program-based methods predict procedural modeling sequences that can be executed by a CAD kernel to generate the target B-Rep (Pyatov et al., 2026; Kolodiazhnyi et al., 2025; Rukhovich et al., 2025b; Kabisov et al., 2026; Qi et al., 2026). Surface-based methods recover its geometric primitives (Liu et al., 2024b; Sharma et al., 2020; Li et al., 2023; Yan et al., 2021; Fang et al., 2025) and their connectivity directly (Guo et al., 2022; Liu et al., 2024a; 2025b; 2026). In this paper, we focus on the latter, where primitive identification, geometry estimation, and topology recovery place complementary demands on the input representation simultaneously. Despite the clear interdependence of these tasks, leading approaches often tackle the problem by dividing it into discrete, task-specific components.

For instance, ComplexGen (Guo et al., 2022) first predicts primitive proposals with connectivity probabilities, then reconstructs the B-Rep through constrained optimization (DiCarlo et al., 2014). Fusing these proposals with subsequent optimization steps is non-trivial and can introduce additional computational overhead. Recent feedforward such as DualBrep (Liu et al., 2026) attempt to bypass explicit optimization by directly predicting implicit fields and reconstructing B-Reps in a single pass. However, they still learn separate reconstruction networks that operate on intermediate predictions, rather than directly reusing the features that produced them. Yet a primitive prediction need not retain all the contextual evidence useful for later reconstruction. A face mask, for example, specifies region membership but does not convey the learned features describing the underlying surface. This motivates us to establish a shared foundation feature, in which geometric priors remain accessible across stages and adapt to the distinct demands of specified steps.

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

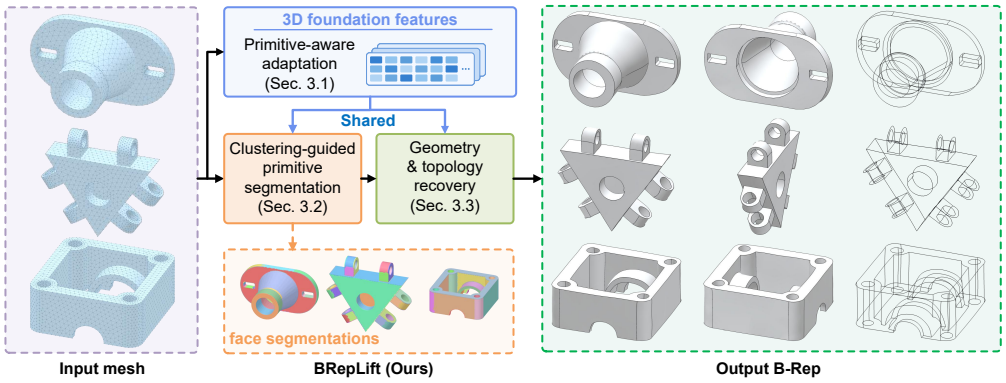

Figure 1: **Overview of BRepLift**. Given a mesh as input, we adapt pretrained 3D features to distinguish B-Rep primitives and share the resulting representation across face segmentation and geometry and topology recovery. The resulting framework with shared primitive-aware features can process meshes with diverse geometries, including those with complex curved surfaces.

Large-scale pre-trained 3D foundation models offer a promising source of such a representation. Models such as PartField (Liu et al., 2025a) and Utonia (Zhang et al., 2026) provide rich geometric priors, which already support tasks such as part segmentation and articulation estimation (Yang et al., 2026; Li et al., 2026a). Their use for B-Rep reconstruction, however, requires resolving a *mismatch in granularity*, as B-Rep faces are finer-grained, like the individual faces of a B-Rep, which can be much smaller and more numerous than the parts typically captured by 3D foundation models.

In this paper, we introduce BRepLift, a framework that reconstructs B-Reps from meshes using shared 3D foundation features, as illustrated in Fig. 1(b). Our *key idea* is to use primitive predictions to organize the shared 3D foundation priors into context for subsequent reconstruction. Specifically, we first adapt pretrained 3D foundation features with a B-Rep face-level *contrastive objective* that encourages feature consistency within a face and separation across faces. Then, we also introduce a clustering-guided strategy to cluster the adapted features into local regions and learn to aggregate these regions into complete face instances, preserving subtle primitive boundaries while enabling primitive-level context grouping. Finally, we propose a shared-prior framework that uses the predicted face masks to group the shared features into intra-face contexts for surface reconstruction and cross-face contexts for edge geometry and existence prediction. In this way, segmentation provides structural guidance while reconstruction retains direct access to the adapted foundation priors.

We evaluate BRepLift on spanning different geometric and topological complexities: the Dual-Brep Liu et al. (2026) and ABC-ParseNet Sharma et al. (2020) splits, and a new test set we curated from Fusion360 Lambourne et al. (2021) and CC3D-Ops Dupont et al. (2022). Compared with DualBrep, our method improves face, edge, and vertex F1 scores and B-Rep validity on all three test sets. On ABC-ParseNet, the percentage of valid reconstructions increases from 10.60% to 22.14% (relative improvement: 108.49%); on the curated test set, it increases from 17.34% to 29.46% (relative improvement: 69.96%). In summary, our key contributions are as follows:

- We introduce BRepLift, which shares adapted 3D foundation features across primitive identification, geometry recovery, and topology reasoning for B-Rep reconstruction.
- We develop stage-specific designs that adapt shared features at intra- and inter-primitive levels, including clustering-guided B-Rep face identification and complementary context-conditioned parametric geometry reconstruction and topology reasoning.
- In an extensive set of experiments, we show that BRepLift consistently outperforms existing approaches in geometric accuracy, topological correctness, and B-Rep validity.

2 Related Works

Surface-based methods recover geometric primitives and their topology from 3D observations. Primitives are localized through proposal prediction (Guo et al., 2022), segmentation (Li et al., 2019; Sharma et al., 2020; Li et al., 2023; Yan et al., 2021; Huang et al., 2021; Fang et al., 2025), or implicit Voronoi representations that additionally encode face adjacency (Liu et al., 2024a; 2026).

Geometry recovery uses analytic fitting (Schnabel et al., 2007; Schneider et al., 2003), neural implicit fitting (Liu et al., 2024b), or learned prediction of control points (Sharma et al., 2020), quadric parameters (Chen et al., 2026), and structured point grids (Liu et al., 2025b; 2026). Topology is recovered through geometric intersections (Liu et al., 2024b; Li et al., 2023) or connectivity optimization under chain-complex constraints (Guo et al., 2022; DiCarlo et al., 2014). Our work adapts pretrained 3D foundation features into a shared B-Rep-aware representation reused across these stages to handle geometric variation and complex topology.

Program-based methods reconstruct B-Rep models through procedural modeling sequences, progressing from sketch-and-extrude generation (Wu et al., 2021; Xu et al., 2022; 2023; Wang et al., 2026) to LLM-based code generation and multimodal reconstruction (Rukhovich et al., 2025a; Xu et al., 2024a; Wang et al., 2025; Li et al., 2026b). Further improvements incorporate structural priors (Dupont et al., 2024), contrastive or masked learning (Ma et al., 2023; 2024), and reinforcement learning (Guan et al., 2026; Kolodiazhnyi et al., 2025; Kabisov et al., 2026). Unlike these approaches, our method directly reconstructs B-Rep geometry and topology.

3D Foundation Models and Applications. Large-scale pretraining provides expressive priors for 3D generation in models such as TRELLIS (Xiang et al., 2025) and Hunyuan3D (Team, 2025), and transferable geometric representations in PartField (Liu et al., 2025a) and Utonia (Zhang et al., 2026). Such priors support downstream tasks including part-level generation (Chen et al., 2025) and kinematics understanding (Li et al., 2026a), etc. Broader coverage is available in recent surveys (Thengane et al., 2025; Hu et al., 2026). We investigate their adaptation to B-Rep reconstruction for improved robustness and generalization.

3 METHOD

Given a mesh $\mathcal{M} = (V, F)$ with vertices V and faces F, our goal is to recover a watertight B-Rep solid representation $\mathcal{B} = (\mathcal{F}, \mathcal{E}, \mathcal{V}, \mathbf{T}_{FE}, \mathbf{T}_{EV})$, where $\mathcal{F}$, $\mathcal{E}$, and $\mathcal{V}$ denote the sets of faces, edges, and vertices, respectively, and $\mathbf{T}_{FE}$ and $\mathbf{T}_{EV}$ encode face-edge and edge-vertex topology.

Our *key idea* is to leverage prior knowledge from large-scale pretrained 3D foundation models for identifying primitives and reconstructing accurate B-Rep geometry and topology. In particulate, we introduce BRepLift, as illustrated in Fig. 1 (b), a shared-prior framework that adapts pretrained 3D representations to B-Rep reconstruction. It first adapts pretrained 3D features into B-Rep primitive-aware features (section 3.1), followed by clustering-guided primitive segmentation (section 3.2). Finally, it constructs intra- and cross-primitive geometric contexts to guide the reconstruction of B-Rep geometry and topology (section 3.3). Next, we discuss these components in detail.

3.1 PRIMITIVE-AWARE FEATURE ADAPTATION

Pretrained representation. We first extract vertex-wise features from the input mesh $\mathcal{M}$ using Utonia Zhang et al. (2026), a 3D foundation model built on PTv3 Wu et al. (2024). Its hierarchical backbone combines local geometric information with broader shape context. We first sample M vertices from $\mathcal{M}$ and feed them into the pretrained backbone to obtain vertex-wise features $\mathbf{F}^{(l)} \in \mathbb{R}^{M \times D}$, where $l = 0, 1, 2, 3$ indexes the four hierarchical blocks of the backbone and D is the feature dimension. We channel-concatenate the block features of the input vertices to obtain $\mathbf{F} = \mathrm{Concat}\left(\{\mathbf{F}^{(l)}\}_{l=0}^{3}\right) \in \mathbb{R}^{M \times D}$. These features provide a useful starting point for B-Rep reconstruction, but their pretraining objective does not explicitly distinguish individual B-Rep primitives. In particular, neighboring faces can have similar features even when they correspond to distinct surface primitives.

Primitive-aware adaptation. We adapt it into a primitive-aware representation with LoRA adapters. Specifically, we freeze the shallow blocks and only train the LoRA adapters in the final block.

$$\mathbf{F}' = \mathrm{Concat}\left(\mathbf{F}^{(0)}, \mathbf{F}^{(1)}, \mathbf{F}^{(2)}, \mathbf{F}^{(3)}_{\mathrm{LoRA}}\right) \in \mathbb{R}^{M \times D}, \tag{1}$$

where $\mathbf{F}^{(3)}_{\mathrm{LoRA}}$ is the final block output after LoRA adaptation, which is optimized with a contrastive objective. It increases the feature similarity among vertices belonging to the same B-Rep face while

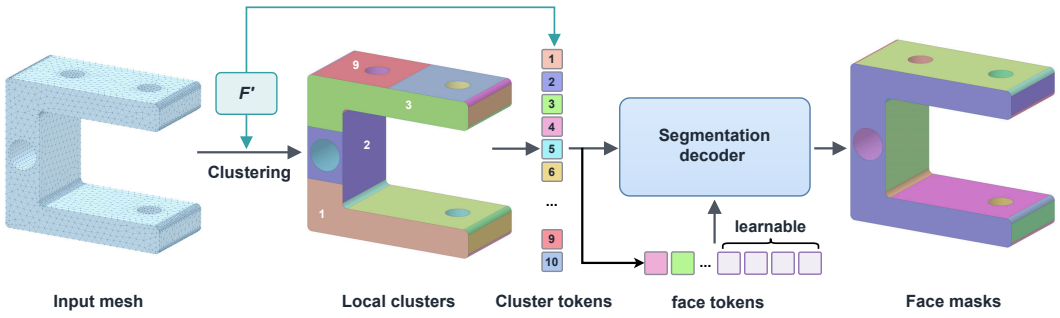

Figure 2: **Overview of the B-Rep primitive segmentation module**. Shared adapted features $\mathbf{F}'$ guide local clustering and cluster-token construction. The segmentation decoder aggregates clusters into face masks using face queries initialized from sampled cluster tokens and learnable embeddings.

reducing the similarity between these on different faces. The loss is formmulated as follows:

$$\mathcal{L}_{\text{adapt}} = -\frac{1}{m}\sum_{j=1}^{m}\frac{1}{|\mathcal{P}(j)|}\sum_{p\in\mathcal{P}(j)}\log\frac{\exp\left(\cos(\mathbf{f}'_j,\mathbf{f}'_p)/\tau\right)}{\sum_{a\in\mathcal{P}(j)\cup\mathcal{N}(j)}\exp\left(\cos(\mathbf{f}'_j,\mathbf{f}'_a)/\tau\right)}, \tag{2}$$

where m is the sampled anchor vertex count, $\mathbf{f}'_*$ is the adapted feature of the $*$-th anchor vertex, where $* \in \{j, p, a\}$ refers to the anchor, positive, and all sampled vertices, respectively. $\mathcal{P}(j)$ and $\mathcal{N}(j)$ contain vertices from the same face and other faces, respectively, and τ is the temperature.

3.2 Cluster-guided Primitive Segmentation

After obtaining the adapted features $\mathbf{F}'$ for the vertices, the primitive segmentation module leverages these features to predict B-Rep face masks over these vertices. We introduce a clustering-guided segmentation module to leverage local feature consistency for more accurate B-Rep face segmentation.

Local cluster representation. As shown in fig. 2, we apply agglomerative clustering (Müllner, 2011) to the adapted vertex features to obtain K compact local regions that largely preserve B-Rep face boundaries. We represent each cluster by a feature token $\mathbf{z}_k$ and a positional embedding (PE) $\mathbf{p}_k$:

$$\mathbf{z}_k = \text{Pool}_{\text{attn}}\left(\{\mathbf{f}'_i\}_{i\in S_k}\right), \qquad \mathbf{p}_k = \text{Pool}_{\text{attn}}\left(\{\text{PE}(\mathbf{v}_i)\}_{i\in S_k}\right), \tag{3}$$

where S_k is the vertex index set of cluster k, $\mathbf{f}'_i$ is the corresponding adapted feature of vertex i, with original features $\mathbf{f}_i \in \mathbf{F}'$, and $\mathbf{v}_i$ is the 3D coordinate of vertex i on the mesh. $\text{PE}(\cdot)$ denotes sinusoidal positional encoding, and $\text{Pool}_{\text{attn}}(\cdot)$ represents attentive pooling, which computes a weighted sum of the features, where the weights are determined by an attention mechanism. In practice, we pool over a fixed number of vertices sampled uniformly from each cluster.

Learned aggregation. A Transformer (Kirillov et al., 2023; Li et al., 2026a) process the cluster tokens $\{\mathbf{z}_k\}_{k=1}^{K}$ with their PE $\{\mathbf{p}_k\}_{k=1}^{K}$ and N_{max} queries $\{\mathbf{q}_n\}_{n=1}^{N_{\text{max}}}$. Since the number of B-Rep faces is unknown, we set N_{max} larger than actual face count in a typical B-Rep. Then, for half of the queries, we initialize them with learnable embeddings and the other half are sampled cluster tokens. Each attention block in the Transformer has alternating query self-attention, query-to-cluster and then cluster-to-query cross-attention, producing updated tokens $\{\widetilde{\mathbf{z}}_k\}_{k=1}^{K}$ and $\{\widetilde{\mathbf{q}}_n\}_{n=1}^{N_{\text{max}}}$. We then prdict a matrix of logits $\widetilde{\mathbf{A}} \in \mathbb{R}^{K\times N_{\text{max}}}$, denoting which clusters belong to which queries using a similarity score, $\widetilde{\mathbf{A}}_{k,n} = \text{sigmoid}(\phi_q(\widetilde{\mathbf{q}}_n)^{\mathrm{T}}\phi_c(\widetilde{\mathbf{z}}_k))$, where $\phi_q(\cdot)$ and $\phi_c(\cdot)$ are learnable projections.

Mesh vertex segmentation. During inference, We assign each cluster to the retained query with the highest logits and lift the assignments to the mesh vertices, $\ell_k = \arg\max_n \widetilde{\mathbf{A}}_{k,n}, \Omega_n = \bigcup_{k:\ell_k=n} S_k$, where ℓ_k denotes the query assigned to cluster k. Then, we obtain the final face masks $\{\Omega_{n_f}\}_{n_f=1}^{N_f}$, where each Ω_{n_f} is a set of mesh vertices assigned to the face n_f, with totally $N_f \le N_{\text{max}}$ faces.

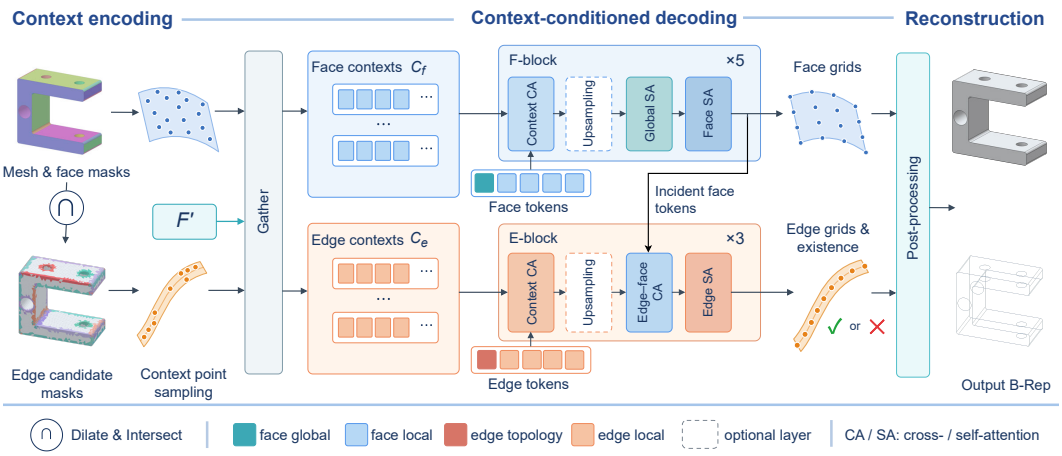

Figure 3: **Overview of the B-Rep reconstruction module**. Features at sampled context points are gathered from the shared adapted representation $\mathbf{F}'$ to form face and edge contexts, which guide geometry and edge-existence prediction. The edge branch additionally attends to incident-face tokens. Post-processing assembles the predictions into the final B-Rep.

3.3 Context-conditioned Geometry & Topology Recovery

This section reconstructs the B-Rep $\mathcal{B}$ from the input mesh $\mathcal{M}$, adapted features $\mathbf{F}'$, and predicted face masks $\{\Omega_{n_f}\}_{n_f=1}^{N_f}$. The *central design* is to condition each primitive on a relevant subset of the *shared features: a face uses features from its segmented region, while an edge uses features near the boundary between its incident faces.* As illustrated in fig. 3, the network predicts face and edge geometry together and determines the validity of edge candidates conditioned on *primitive-specific* contexts. Following DualBrep Liu et al. (2026), we then fit parametric primitives $\mathcal{F}, \mathcal{E}, \mathcal{V}$ and consistent topology $\mathbf{T}_{FE}, \mathbf{T}_{EV}$ to assemble the final B-Rep through optimization-based post-processing.

Primitive Contexts. For face context, we uniformly sample mesh vertices from each face mask Ω_{n_f} and use their adapted features to form the **face context** $\mathbf{C}_{f,n_f} = \{\mathbf{f}'_i\}_{i\in\Omega_{n_f}}$. To construct **edge context**, we dilate each pair of face masks over mesh connectivity and intersect them. In particular, we apply DBSCAN to each nonempty intersection to separate spatially distinct regions. Each region defines an edge candidate, allowing a face pair to share multiple edges. For each edge candidate, let Ω_{n_e} denote the set of mesh vertices in the n_e-th edge candidate. We then uniformly sample mesh vertices from each edge candidate to form the **edge context** $\mathbf{C}_{e,n_e} = \{\mathbf{f}'_i\}_{i\in\Omega_{n_e}}$.

Primitive queries. For each **face**, we each use a 2×2 grid of *local tokens* encoding face geometry and 1 *global token* for information exchange across faces. For each **edge**, we use 2 local tokens and 1 *topology token* for edge-existence prediction. All tokens are just learnable embeddings at initialization and are updated through cross-attention with their corresponding primitive contexts.

Primitive decoding. The face branch and edge branch contain a series of attention blocks. There are five attention blocks in the face branch. Each block first applies cross-attention to its primitive context $\mathbf{C}_{f,n_f}$, followed by global self-attention over the local and global tokens of all faces, and finally face-wise self-attention within each face's tokens. In the last three blocks, the local token grids are bilinearly upsampled to 4×4, 8×8, and 16×16 before attention refinement. The global token is retained throughout decoding.

The edge branch comprises three attention blocks aligned with the final three face blocks. Each block first applies cross-attention to the edge context $\mathbf{C}_{e,j}$, followed by cross-attention to the tokens of its two incident faces, f_{j_1} and f_{j_2} output by the corresponding face block, and finally self-attention within each edge candidate's tokens. The incident-face cross-attention replaces global self-attention in edge branch to prioritize adjacent-face information that directly constrains edge geometry and existence. Across the three blocks, the local tokens are progressively upsampled to lengths 4, 8, and 16, while the topology token is retained throughout decoding.

Table 1: **Overall quantitative results** across different test sets. Validity ratio is not reported for SED-Net+Point2CAD because it outputs polygonal meshes instead of explicit B-Reps.

Testset	Methods	Global Geometry		Primitive Accuracy (F1↑)			Topology Accuracy (F1↑)		Validity Ratio (↑)
		Shape CD ↓	NC ↑	Face	Edge	Vertex	Face-Edge	Edge-Vertex	
DualBrep	SEDNet+Point2CAD	0.0256	0.769	0.479	0.440	0.402	0.403	0.437	–
	DualBrep	0.0154	0.918	0.899	0.846	0.852	**0.895**	**0.909**	60.97
	Ours	**0.0152**	**0.922**	**0.916**	**0.8536**	**0.877**	0.889	0.897	**64.93**
ParseNet	SEDNet+Point2CAD	0.0413	0.703	0.325	0.311	0.304	0.294	0.291	–
	DualBrep	**0.0330**	0.849	0.623	0.499	0.482	0.525	0.541	10.60
	Ours	0.0333	**0.890**	**0.745**	**0.592**	**0.622**	**0.644**	**0.649**	**22.14**
Ours	SEDNet+Point2CAD	0.0405	0.682	0.288	0.273	0.269	0.210	0.232	–
	DualBrep	0.0261	0.843	0.543	0.467	0.459	0.555	0.595	17.34
	Ours	**0.0226**	**0.882**	**0.717**	**0.605**	**0.604**	**0.613**	**0.610**	**29.46**

Geometry and existence prediction. Following prior work Guo et al. (2022); Liu et al. (2024b; 2026), we represent primitive geometry with structured point grids, which support both analytic and free-form surfaces and curves. Geometry heads map the decoded local tokens to $\mathbf{G}_{f,i} \in \mathbb{R}^{N_G \times N_G \times 3}$, $\mathbf{G}_{e,j} \in \mathbb{R}^{N_G \times 3}$, where $N_G = 16$. A binary classification head maps each topology token to an edge-existence confidence, yielding a binary selection vector $\mathbf{b} \in \{0,1\}^{N_e'}$ after thresholding.

B-Rep assembly. We retain the $N_e = \sum_{j=1}^{N_e'} b_j$ selected edge candidates and reindex them as $k = 1, \ldots, N_e$, preserving their associated face pairs (k_1, k_2). These pairs directly determine the face-edge topology: $\mathbf{T}_{FE}[i,k] = \mathbf{1}\left[i \in k_1, k_2\right], \mathbf{T}_{FE} \in 0, 1^{N_f \times N_e}$, where $\mathbf{1}[\cdot]$ denotes the indicator function. Following DualBrep Liu et al. (2026), we fit parametric surfaces and curves to the predicted face grids and retained edge grids, recover vertices $\mathcal{V}$ and edge-vertex topology $\mathbf{T}_{EV}$, and assemble the primitives into the final B-Rep.

4 Experiments

4.1 Experimental Setups

Datasets. We evaluate on three test sets with complementary B-Rep complexity. The DualBrep (Liu et al., 2026) split contains about 4,000 mostly regular CAD shapes with moderate geometric complexity. ABC-ParseNet (Sharma et al., 2020) contains 3,500 shapes, each with at least one B-Spline face, exhibiting the greater geometric complexity but relatively simple topology. We further curate 3,500 shapes from Fusion360 Gallery (Lambourne et al., 2021) and CC3D-Ops (Dupont et al., 2022), providing a more balanced combination of geometric variation and topological complexity. Complexity measures, dataset construction, and detailed statistics are provided in the section A.

Baselines. We compare BRepLift with two state-of-the-art pipelines using their publicly released pretrained models. SED-Net (Li et al., 2023)+Point2CAD (Liu et al., 2024b) represents a fitting-based pipeline that segments primitive regions and reconstructs them through parametric fitting. DualBrep (Liu et al., 2026) represents a learning-based pipeline that predicts primitive geometry and topology from learned geometric contexts.

Evaluation Metrics. For B-Rep face segmentation, predicted and ground-truth faces are matched one-to-one by mask IoU; we report precision, recall, and F1 at IoU thresholds of 0.5 and 0.75, together with mIoU for matched face pairs. For reconstruction, predicted and ground-truth primitives are matched one-to-one based on Chamfer Distance, with a pair considered successfully matched only if its distance is below 0.01. We report face, edge, and vertex precision, recall, and F1, Chamfer Distance over successfully matched primitives, and topology accuracy based on predicted face-edge and edge-vertex incidences. Overall reconstruction is further evaluated using shape-level Chamfer Distance, normal consistency, and B-Rep validity ratio.

4.2 Comparisons

Overall B-Rep Reconstruction. Table 1 reports B-Rep reconstruction on all three test sets. We evaluate a single input orientation without test-time rotation augmentation, which is stricter than the original DualBrep (Liu et al., 2026) protocol that selects successful reconstructions from multiple rotated inputs; consequently, the absolute validity ratios reported here are lower.

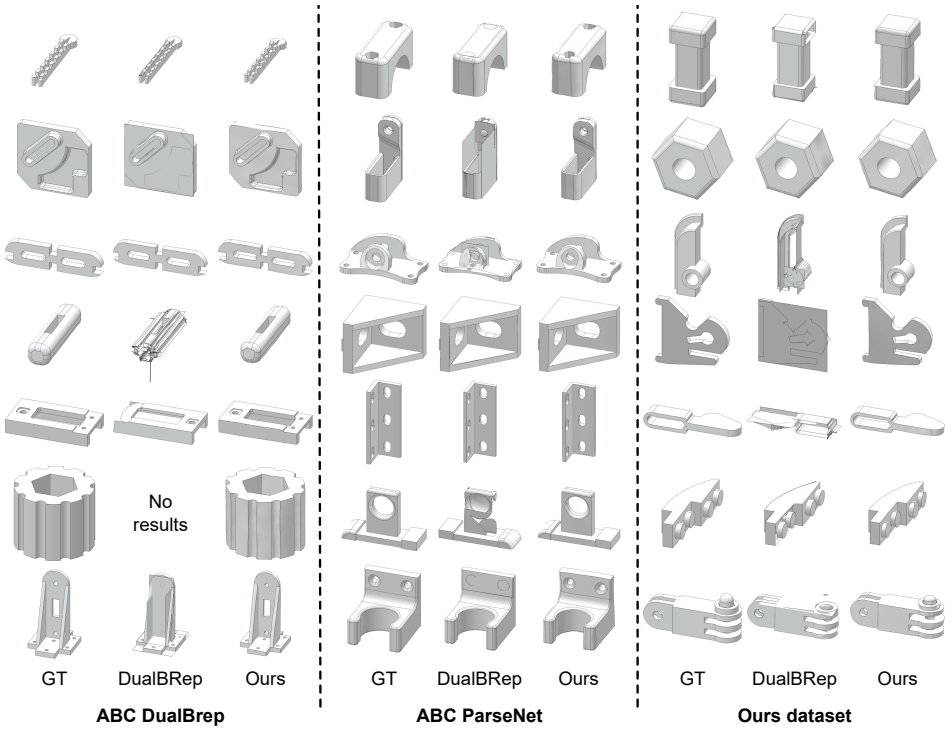

Figure 4: **Qualitative comparison of B-Rep reconstruction** on test sets. BRepLift outperforms the baselines in preserving geometric details and topological correctness.

On ABC-DualBrep, our method achieves the best global geometry, improves primitive recovery, and increases the validity ratio from 60.97% to 64.93%, while maintaining comparable topology accuracy. On the more geometrically challenging ABC-ParseNet split, our method achieves comparable Chamfer Distance to DualBrep (0.0333 vs. 0.0330), but substantially improves normal consistency, primitive and topology recovery, and raises validity from 10.60% to 22.14%. On our curated test set which combines geometric variation with topological complexity, our method reduces Chamfer Distance from 0.0261 to 0.0226 and increases validity from 17.34% to 29.46%. These results demonstrate improved robustness on shapes that present both geometric and topological challenges.

As shown in Fig. 4, our method better preserves curved and free-form surfaces, thin structures, and locally complex configurations, while DualBrep more frequently exhibits distorted geometry, missing structures, or incorrect primitive assembly. Additional qualitative results for SED-Net+Point2CAD are provided in Appendix.

Primitive Segmentation. Table 2 shows that our method consistently outperforms SED-Net across all three test sets, with substantial gains in matched mIoU and instance-level precision, recall, and F1-score at both IoU thresholds. The improvement is particularly clear at IoU$= 0.75$, indicating more accurate and complete face delineation rather than only coarse region localization.

The qualitative results in Fig. 5 further show that our method produces cleaner primitive boundaries and better preserves small faces, while SED-Net more frequently introduces boundary noise or merges minor primitives into larger neighborings. These results demonstrate the advantage of the adapted foundation features and clustering-guided segmentation for fine-grained B-Rep face identification.

4.3 Ablations

Primitive Segmentation. We ablate primitive-aware feature adaptation and clustering-guided segmentation in Table 3. Removing clustering only slightly reduces matched mIoU (81.88→81.63), but lowers F1 from 0.898 to 0.860 at IoU$= 0.50$ and from 0.799 to 0.778 at IoU$= 0.75$, mainly due to reduced recall. This indicates that clustering helps preserve small or ambiguous faces and prevents cross-boundary merging. Removing feature adaptation causes a much larger degradation, with matched mIoU dropping to 71.45 and F1 to 0.796/0.540 at IoU$= 0.50/0.75$, showing its impor-

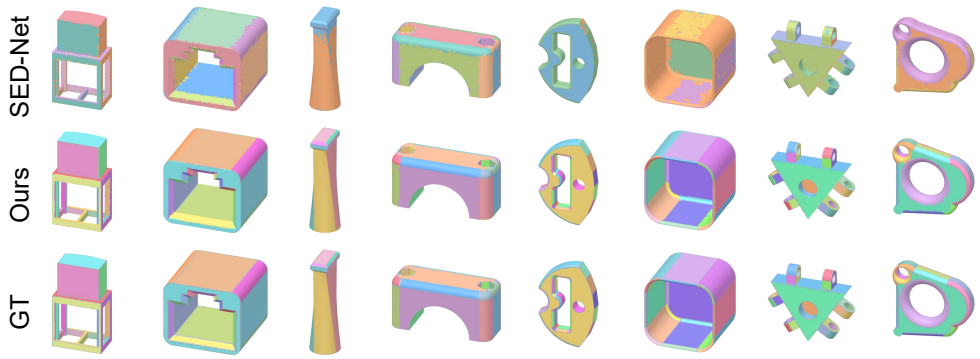

Figure 5: **Primitive segmentation comparisons**, with our predicted faces colored consistently with the ground-truth for easier comparison and SED-Net faces assigned random colors.

Table 2: Quantitative comparison of B-Rep face segmentation on three test sets. Higher is better for all metrics; best results are highlighted in bold.

Testset	Methods	Matched mIoU (↑)	IoU = 0.50			IoU = 0.75		
			P	R	F1	P	R	F1
DualBrep	SEDNet	70.05	0.780	0.474	0.589	0.578	0.351	0.437
	Ours	**87.92**	**0.973**	**0.980**	**0.980**	**0.932**	**0.929**	**0.930**
ParseNet	SEDNet	66.64	0.707	0.301	0.422	0.462	0.197	0.276
	Ours	**81.88**	**0.953**	**0.849**	**0.898**	**0.848**	**0.755**	**0.799**
Ours	SEDNet	62.29	0.653	0.298	0.406	0.434	0.196	0.270
	Ours	**81.44**	**0.934**	**0.923**	**0.928**	**0.820**	**0.809**	**0.814**

tance for fine-grained primitive discrimination and accurate boundary localization. As illustrated in Fig. 6, clustering improves small-face preservation and boundary sharpness, while feature adaptation substantially reduces jagged boundaries and the absorption of small primitives into neighboring faces.

Geometry and Topology Reconstruction. We ablate the reconstruction stage using ground-truth face masks to eliminate errors from primitive segmentation. Replacing the adapted features with the original Utonia features consistently worsens geometric accuracy, increasing the face/edge/vertex Chamfer Distance from 0.0182/0.0238/0.0299 to 0.0192/0.0242/0.0316, while primitive and topology metrics remain comparable. This indicates that feature adaptation mainly improves geometric fidelity, which is also reflected by the cleaner reconstructions in Fig. 7.

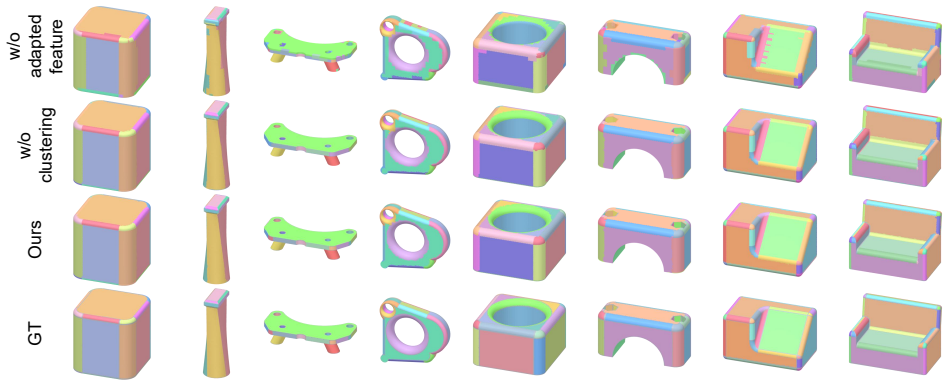

Figure 6: **Qualitative ablation results for primitive segmentation.** 'w/o clustering' indicates the results without clustering-guided segmentation, and 'w/o adaptation' indicates the results without primitive-aware feature adaptation.

Table 3: Ablation study on face segmentation on ABC-ParseNet. Adapted Feature enables primitive-aware adaptation; Clustering enables cluster-level rather than vertex-level mask prediction.

Adapted Feature	Clustering	Matched mIoU (↑)	IoU = 0.50			IoU = 0.75		
			P	R	F1	P	R	F1
-	✓	71.45	0.845	0.752	0.796	0.573	0.511	0.540
✓	-	81.63	0.930	0.799	0.860	0.842	0.724	0.778
✓	✓	**81.88**	**0.953**	**0.849**	**0.898**	**0.848**	**0.755**	**0.799**

Table 4: Ablation study on geometry and topology reconstruction. Adapted Feature denotes primitive-aware feature adaptation; Context-CA denotes cross-attention to primitive contexts. All variants use ground-truth face segmentation masks.

Adapted Feature	Context-CA	Primitive Geometry (CD↓)			Primitive Accuracy (F1↑)			Topology Accuracy (F1↑)	
		Face	Edge	Vertex	Face	Edge	Vertex	Face-Edge	Edge-Vertex
✓	–	0.0677	0.0939	0.1370	0.642	0.401	0.329	0.320	0.304
–	✓	0.0192	0.0242	0.0316	0.927	**0.747**	**0.779**	0.799	**0.812**
✓	✓	**0.0182**	**0.0238**	**0.0299**	**0.938**	0.733	0.777	**0.804**	0.803

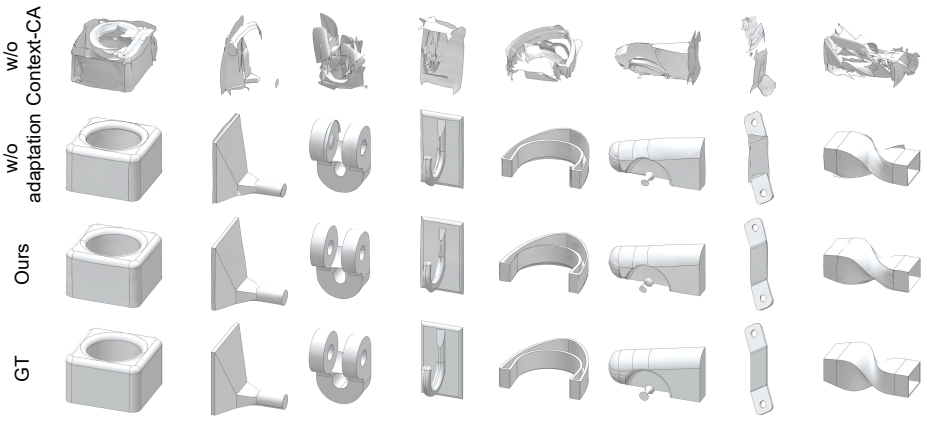

Figure 7: Qualitative ablation results for geometry & topology reconstruction.

The larger drop occurs when removing the repeated context-conditioned cross-attention layers while retaining only the first one in each face and edge branch: face/edge/vertex F1-scores fall to 0.642/0.401/0.329, and face-edge/edge-vertex topology accuracies to 0.320/0.304. As shown in Fig. 7, this variant frequently produces distorted geometry and invalid B-Reps, confirming that persistent primitive-specific context is critical for joint geometry and topology reconstruction.

5 Conclusions

In this work, we present a shared-prior framework for B-Rep reconstruction that adapts pretrained 3D foundation features into B-Rep-aware representations and reuses them across primitive segmentation, geometry recovery, and topology reconstruction. Experiments on datasets with increasing geometric and topological complexity demonstrate the effectiveness of foundation representations for fine-grained B-Rep reasoning and show that primitive-aware adaptation improves both reconstruction robustness and generalization. These results highlight the potential of pretrained 3D features as transferable priors for structured CAD reconstruction. Our current pipeline still relies on multiple separately trained stages and intermediate predictions. Future work will investigate more unified formulations that jointly adapt and exploit foundation representations across B-Rep reconstruction, reducing stage-wise complexity and improving end-to-end consistency.

REFERENCES

Christopher M. Brown. Some mathematical and representational aspects of solid modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3(4):444–453, 1981.

Minghao Chen, Jianyuan Wang, Roman Shapovalov, Tom Monnier, Hyunyoung Jung, Dilin Wang, Rakesh Ranjan, Iro Laina, and Andrea Vedaldi. Autopartgen: Autogressive 3d part generation and discovery. In *NeurIPS*, 2025.

Zhaiyu Chen, Yuqing Wang, and Xiao Xiang Zhu. Unified primitive proxies for structured shape completion. *arXiv preprint arXiv:2601.00759*, 2026.

Antonio DiCarlo, Alberto Paoluzzi, and Vadim Shapiro. Linear algebraic representation for topological structures. *Computer-Aided Design*, 46:269–274, 2014.

Elona Dupont, Kseniya Cherenkova, Anis Kacem, Sk Aziz Ali, Ilya Arzhannikov, Gleb Gusev, and Djamila Aouada. Cadops-net: Jointly learning cad operation types and steps from boundary-representations. In *2022 International Conference on 3D Vision (3DV)*, pp. 114–123. IEEE, 2022.

Elona Dupont, Kseniya Cherenkova, Dimitrios Mallis, Gleb Gusev, Anis Kacem, and Djamila Aouada. Transcad: A hierarchical transformer for cad sequence inference from point clouds. In *European Conference on Computer Vision*, pp. 19–36. Springer, 2024.

Zheng Fang, Chuanqing Zhuang, Zhengda Lu, Yiqun Wang, Lupeng Liu, and Jun Xiao. Bgpseg: Boundary-guided primitive instance segmentation of point clouds. *IEEE Transactions on Image Processing*, 34:1454–1468, 2025.

Yandong Guan, Xilin Wang, Ximing Xing, Jing Zhang, Dong Xu, and Qian Yu. Cad-coder: Text-to-cad generation with chain-of-thought and geometric reward. *Advances in Neural Information Processing Systems*, 38:59765–59789, 2026.

Haoxiang Guo, Shilin Liu, Hao Pan, Yang Liu, Xin Tong, and Baining Guo. Complexgen: Cad reconstruction by b-rep chain complex generation. *ACM Transactions on Graphics (TOG)*, 41(4): 1–18, 2022.

Christian Hafner, Christian Schumacher, Espen Knoop, Thomas Auzinger, Bernd Bickel, and Moritz Bächer. X-cad: optimizing cad models with extended finite elements. *ACM Trans. Graph.*, 38(6), November 2019.

Yuqi Hu, Longguang Wang, Xian Liu, Ling-Hao Chen, Yuwei Guo, Yukai Shi, Ce Liu, Anyi Rao, Zeyu Wang, and Hui Xiong. Simulating the real world: A unified survey of multimodal generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Jingwei Huang, Yanfeng Zhang, and Mingwei Sun. Primitivenet: Primitive instance segmentation with local primitive embedding under adversarial metric. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15323–15333. IEEE, 2021.

Soslan Kabisov, Gennadiy Savrasov, Maksim Elistratov, Antonio Rodriguez, Daniil Ignatiev, Nikita Gavrilov, Rustam Uzdenov, Alexey I Boyko, Igor Pasechnik, Anton Konushin, et al. Cadena: Stepwise cad reverse engineering. *arXiv preprint arXiv:2608.00799*, 2026.

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 *ICCV*, pp. 4015–4026, 2023.

Maksim Kolodiazhnyi, Denis Tarasov, Dmitrii Zhemchuzhnikov, Alexander Nikulin, Ilya Zisman, Anna Vorontsova, Anton Konushin, Vladislav Kurenkov, and Danila Rukhovich. Cadrille: Multi-modal cad reconstruction with reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2025.

Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955.

Joseph G Lambourne, Karl DD Willis, Pradeep Kumar Jayaraman, Aditya Sanghi, Peter Meltzer, and Hooman Shayani. Brepnet: A topological message passing system for solid models. *arXiv preprint arXiv:2104.00706*, 2021.

Lingxiao Li, Minhyuk Sung, Anastasia Dubrovina, Li Yi, and Leonidas J Guibas. Supervised fitting of geometric primitives to 3d point clouds. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 2647–2655. IEEE, 2019.

Ruining Li, Yuxin Yao, Chuanxia Zheng, Christian Rupprecht, Joan Lasenby, Shangzhe Wu, and Andrea Vedaldi. Particulate: Feed-forward 3d object articulation. In *CVPR*, 2026a.

Xueyang Li, Jiahao Li, Yu Song, Yunzhong Lou, and Xiangdong Zhou. Seek-CAD: A self-refined generative modeling for 3d parametric CAD using local inference via deepseek. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Yuanqi Li, Shun Liu, Xinran Yang, Jianwei Guo, Jie Guo, and Yanwen Guo. Surface and edge detection for primitive fitting of point clouds. In *ACM SIGGRAPH 2023 conference proceedings*, pp. 1–10, 2023.

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 *ICCV*, pp. 9704–9715, 2025a.

Yilin Liu, Jiale Chen, Shanshan Pan, Daniel Cohen-Or, Hao Zhang, and Hui Huang. Split-and-fit: Learning b-reps via structure-aware voronoi partitioning. *ACM Transactions on Graphics (TOG)*, 43(4):1–13, 2024a.

Yilin Liu, Duoteng Xu, Xingyao Yu, Xiang Xu, Daniel Cohen-Or, Hao Zhang, and Hui Huang. Hola: B-rep generation using a holistic latent representation. *ACM Transactions on Graphics (TOG)*, 44 (4):1–25, 2025b.

Yilin Liu, Pradeep Jayaraman, Chinthala Reddy, Xiang Xu, and Hooman Shayani. Dualbrep: A dual-field continuous representation for b-rep modelling. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–11, 2026.

Yujia Liu, Anton Obukhov, Jan Dirk Wegner, and Konrad Schindler. Point2cad: Reverse engineering cad models from 3d point clouds. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3763–3772. IEEE, 2024b.

Weijian Ma, Minyang Xu, Xueyang Li, and Xiangdong Zhou. Multicad: Contrastive representation learning for multi-modal 3d computer-aided design models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 1766–1776, 2023.

Weijian Ma, Shuaiqi Chen, Yunzhong Lou, Xueyang Li, and Xiangdong Zhou. Draw step by step: Reconstructing cad construction sequences from point clouds via multimodal diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27154–27163, 2024.

Daniel Müllner. Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378*, 2011.

Vladislav Pyatov, Gleb Bobrovskikh, Saveliy Galochkin, Nikita Boldyrev, Oleg Voynov, Alexander Filippov, Gonzalo Ferrer, Peter Wonka, and Evgeny Burnaev. Cadfs: A big cad program dataset and framework for computer-aided design with large language models. *arXiv preprint arXiv:2605.01925*, 2026.

Dacheng Qi, Chenyu Wang, Jingwei Xu, Tianzhe Chu, Zibo Zhao, Wen Liu, Wenrui Ding, Yi Ma, and Shenghua Gao. Pointer-cad: Unifying b-rep and command sequences via pointer-based edges & faces selection. *arXiv preprint arXiv:2603.04337*, 2026.

Danila Rukhovich, Elona Dupont, Dimitrios Mallis, Kseniya Cherenkova, Anis Kacem, and Djamila Aouada. Cad-recode: Reverse engineering cad code from point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9801–9811, 2025a.

Danila Rukhovich, Elona Dupont, Dimitrios Mallis, Kseniya Cherenkova, Anis Kacem, and Djamila Aouada. Cad-recode: Reverse engineering cad code from point clouds. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9801–9811. IEEE, 2025b.

Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, volume 26, pp. 214–226. Wiley Online Library, 2007.

Philip J Schneider, David H Eberly, et al. *Geometric tools for computer graphics*, volume 11. Morgan Kaufmann San Francisco, 2003.

Adriana Schulz, Jie Xu, Bo Zhu, Changxi Zheng, Eitan Grinspun, and Wojciech Matusik. Interactive design space exploration and optimization for cad models. *ACM Trans. Graph.*, 36(4), July 2017.

Gopal Sharma, Difan Liu, Subhransu Maji, Evangelos Kalogerakis, Siddhartha Chaudhuri, and Radomír Měch. Parsenet: A parametric surface fitting network for 3d point clouds. In *European conference on computer vision*, pp. 261–276. Springer, 2020.

Tencent Hunyuan3D Team. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation, 2025.

Vishal Thengane, Xiatian Zhu, Salim Bouzerdoum, Son Lam Phung, and Yunpeng Li. Foundational models for 3d point clouds: A survey and outlook. *arXiv preprint arXiv:2501.18594*, 2025.

Yunsheng Tian, Jie Xu, Yichen Li, Jieliang Luo, Shinjiro Sueda, Hui Li, Karl D. D. Willis, and Wojciech Matusik. Assemble them all: Physics-based planning for generalizable assembly by disassembly. *ACM Trans. Graph.*, 41(6), November 2022.

Christopher Uchytil and Duane Storti. A function-based approach to interactive high-precision volumetric design and fabrication. *ACM Trans. Graph.*, 43(1), September 2023.

Ruiyu Wang, Shizhao Sun, Weijian Ma, and Jiang Bian. CAD-tokenizer: Towards text-based CAD prototyping via modality-specific tokenization. In *The Fourteenth International Conference on Learning Representations*, 2026.

Siyu Wang, Cailian Chen, Xinyi Le, Qimin Xu, Lei Xu, Yanzhou Zhang, and Jie Yang. Cad-gpt: Synthesising cad construction sequence with spatial reasoning-enhanced multimodal llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7880–7888, 2025.

Rundi Wu, Chang Xiao, and Changxi Zheng. Deepcad: A deep generative network for computer-aided design models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6772–6782, 2021.

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.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *CVPR*, June 2025.

Jingwei Xu, Chenyu Wang, Zibo Zhao, Wen Liu, Yi Ma, and Shenghua Gao. Cad-mllm: Unifying multimodality-conditioned cad generation with mllm. *arXiv preprint arXiv:2411.04954*, 2024a.

Xiang Xu, Karl DD Willis, Joseph G Lambourne, Chin-Yi Cheng, Pradeep Kumar Jayaraman, and Yasutaka Furukawa. Skexgen: Autoregressive generation of cad construction sequences with disentangled codebooks. *arXiv preprint arXiv:2207.04632*, 2022.

Xiang Xu, Pradeep Kumar Jayaraman, Joseph G Lambourne, Karl DD Willis, and Yasutaka Furukawa. Hierarchical neural coding for controllable cad model generation. *arXiv preprint arXiv:2307.00149*, 2023.

Xiang Xu, Joseph Lambourne, Pradeep Jayaraman, Zhengqing Wang, Karl Willis, and Yasutaka Furukawa. Brepgen: A b-rep generative diffusion model with structured latent geometry. *ACM Transactions on Graphics (TOG)*, 43(4):1–14, 2024b.

Siming Yan, Zhenpei Yang, Chongyang Ma, Haibin Huang, Etienne Vouga, and Qixing Huang. Hpnet: Deep primitive segmentation using hybrid representations. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 2733–2742. IEEE, 2021.

Yunhan Yang, Yuanchen Guo, Yukun Huang, Zi-Xin Zou, Zhipeng Yu, Yangguang Li, Yanpei Cao, and Xihui Liu. Holopart: Generative 3d part amodal segmentation. In *International Conference on Learning Representations*, volume 2026, pp. 52657–52678, 2026.

Yujia Zhang, Xiaoyang Wu, Yunhan Yang, Xianzhe Fan, Han Li, Yuechen Zhang, Zehao Huang, Naiyan Wang, and Hengshuang Zhao. Utonia: Toward one encoder for all point clouds. *arXiv preprint arXiv:2603.03283*, 2026.

Appendix Contents

A Dataset Configurations **13**

B Training Details **14**

B.1 Feature Adaptation 15

B.2 Primitive Segementation 15

B.3 Geometry & Topology Recovery 15

C Additional Results **16**

A Dataset Configurations

Dataset Curation. To complement existing benchmarks with a balanced combination of geometric and topological complexity, we curate CAD models from Fusion360 Gallery Lambourne et al. (2021) and CC3D-Ops Dupont et al. (2022) through two filtering steps.

First, *hard filtering* removes invalid, degenerate, or unsuitable shapes. We retain shapes with more than 5 and fewer than 64 B-Rep faces, fewer than 50 incident edges per face, and no edges shorter than 0.005. We also remove duplicate and near-duplicate shapes following the general strategy of BRepGen Xu et al. (2024b). Geometric hashes group potentially similar shapes, which are then compared using B-Rep statistics and geometric similarity. Pairs whose face or edge counts differ by more than 15% are considered distinct. For the remaining pairs, we normalize shapes to $[-1, 1]^3$, uniformly sample point clouds, and compute their symmetric L1 Chamfer distance. Pairs with a distance exceeding 0.02 are considered distinct; only one representative is retained from each duplicate group.

Second, *relative complexity filtering* reduces the proportion of geometrically trivial shapes. We rank the remaining shapes by the geometric complexity score defined below, discard the lowest 30%, and randomly select 3,500 shapes from the remaining pool. This excludes overly simple cases without imposing an absolute complexity threshold.

Complexity Evaluation. We characterize each shape by its *geometric complexity*, based on surface-normal variation along face parametric directions (UV), and *topological complexity*, based on primitive connectivity.

For each B-Rep face, let $\mathbf{p}_{u,v}$ and $\mathbf{n}_{u,v}$ denote the position and unit normal sampled on a UV grid. Along the U direction, we define the unit tangent and signed normal variation as

$$\mathbf{t}^U_{u,v} = \frac{\mathbf{p}_{u+1,v} - \mathbf{p}_{u,v}}{||\mathbf{p}_{u+1,v} - \mathbf{p}_{u,v}||_2}, \qquad q^U_{u,v} = (\mathbf{n}_{u+1,v} - \mathbf{n}_{u,v})^\top \mathbf{t}^U_{u,v}. \tag{4}$$

For each fixed v, we compute

$$R^U_v = \mathbf{1}\left[\mathrm{mean}_u\, |q^U_{u,v}| > \epsilon\right] \left(1 + \frac{\mathrm{std}_u(q^U_{u,v})}{\mathrm{mean}_u\, |q^U_{u,v}| + \epsilon}\right), \tag{5}$$

Table 5: Dataset complexity statistics. Geometric complexity, mean face degree, and mean vertex degree are reported as mean and standard deviation (Std) across shapes; multi-edge face pairs are reported as the mean per-shape proportion.

Dataset	Geometric complexity		Mean face degree		Mean vertex degree		Multi-edge face pairs
	Mean	Std	Mean	Std	Mean	Std	Proportion (%)
ABC-DualBrep	1.124	0.909	4.578	0.804	**2.685**	0.556	0.0
ABC-ParseNet	**2.735**	**1.683**	3.513	**1.346**	2.296	**0.931**	1.7
Ours	2.007	1.116	4.153	1.118	2.601	0.765	**4.2**

where $\mathbf{1}[\cdot]$ is the indicator function. The V-directional statistic $R^V u$ is computed analogously by exchanging u and v.

The indicator suppresses negligible normal variation, while the normalized standard deviation measures face non-uniformity. Planar faces get zero scores, whereas curved faces get higher scores as their directional normal variation becomes less uniform.

We average the directional statistics to obtain the score for face f,

$$R_f = \text{mean}_v \, R_v^U + \text{mean}_u \, R_u^V. \tag{6}$$

The shape-level geometric complexity is

$$Q_{\text{geo}} = \frac{\log(1 + N_f)}{N_f} \sum R_f, \tag{7}$$

where N_f is the number of B-Rep faces. The logarithmic factor incorporates primitive count while moderating its contribution.

For *topological complexity*, we report the mean face degree and mean vertex degree, defined as the average number of edges incident to each face and vertex, respectively. We additionally measure the proportion of adjacent face pairs sharing multiple edges which arise naturally in practical CAD models when trimming produces disconnected shared boundaries, as illustrated by E1 and E2 in the right inset. These separated boundaries cannot be merged into a single edge without changing the geometry or face decomposition.

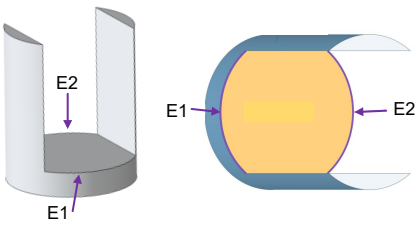

Table 5 summarizes the complementary complexity profiles of the three test sets. ABC-DualBrep has the lowest geometric complexity but the highest mean face and vertex degrees, with no face pairs sharing multiple edges. ABC-ParseNet exhibits the highest geometric complexity and variability, but the lowest mean face and vertex degrees. Our curated dataset lies between the two in geometric complexity and mean connectivity degrees, while exhibiting the highest proportion of multi-edge face pairs (4.2%). It therefore provides a balanced combination of geometric and topological complexity, with more frequent multiple-edge connections between adjacent faces.

B Training Details

We train all three stages of our pipeline separately on the ABC-ParseNet Sharma et al. (2020) training set, containing approximately 22,000 shapes. All stages use an initial learning rate of 5×10^{-4}. The segmentation and reconstruction networks are trained for up to 300 epochs with a batch size of 32. Their learning rate remains fixed for the first 100 epochs and then decreases to 6×10^{-5} through cosine annealing over the remaining 200 epochs.

All models are trained on two NVIDIA RTX PRO 6000 GPUs with 192,GB of total GPU memory. Feature adaptation takes approximately 12 hours, while face segmentation and primitive reconstruction each require approximately two days.

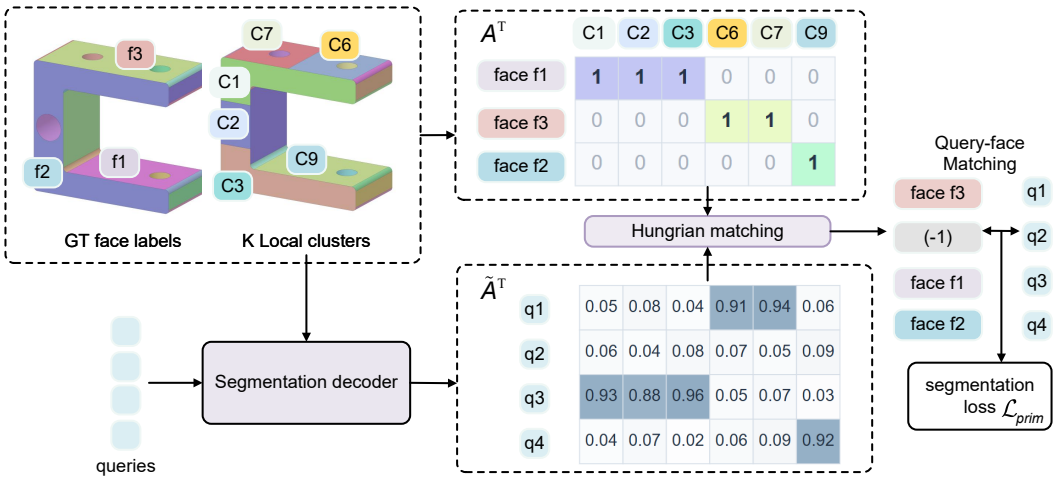

Figure 8: Illustrations for the supervision of cluster-guided primtive segmentation.

B.1 Feature Adaptation

We train the feature adaptation module for 75 epochs and update the sampling weights every 25 epochs. To emphasize ambiguous regions while maintaining sample diversity, we combine hard and randomly sampled examples. Hard positives have low within-face feature similarity, whereas hard negatives have high cross-face similarity. Mining initially uses pretrained features $\mathbf{F}$ and is periodically refreshed using adapted features $\mathbf{F}'$. Each anchor uses 32 positives and 64 negatives, with a hard-to-random ratio of 7:3 in each set and a contrastive temperature of $\tau = 0.1$. The adapted features subsequently guide local clustering for segmentation and provide geometric contexts for reconstruction.

B.2 Primitive Segementation

we assign each cluster to a ground-truth B-Rep face by majority voting over its mesh vertices. This yields a binary target matrix $\mathbf{A} \in \{0,1\}^{K \times N_f^{\text{gt}}}$, where N_f^{gt} is the number of ground-truth faces and $A_{k,i} = 1$ indicates that cluster k is assigned to face i. As illustrated in Fig. 8, each row in $\mathbf{A}^T$ defines a face-instance mask over the K clusters and inherits the primitive-type label of its corresponding B-Rep face.

We use Hungarian matching (Kuhn, 1955) based on mask IoU between $\mathbf{A}$ and $\widetilde{\mathbf{A}}$ to associate predicted query masks with these target masks. Matched queries receive the corresponding face-type labels, while unmatched queries receive the no-object label -1. Mask supervision applies only to matched queries using binary cross-entropy and Dice losses, whereas face-type classification is supervised for all queries using cross-entropy. The overall objective is

$$\mathcal{L}_{\text{prim}} = \lambda_{\text{bce}}\mathcal{L}_{\text{BCE}} + \lambda_{\text{dice}}\mathcal{L}_{\text{Dice}} + \lambda_{\text{type}}\mathcal{L}_{\text{type}}. \tag{8}$$

B.3 Geometry & Topology Recovery

We canonicalize face and edge point grids following DualBrep Liu et al. (2026) and apply random rotation augmentation. To bound training cost, we sample at most 64 face contexts and 48 candidate edge contexts per shape, with up to 512 points per context; contexts containing fewer than 32 points are discarded. Training combines direct primitive supervision, geometric fitting constraints, and edge-existence supervision. The primitive supervision comprises an ℓ_2 point grid reconstruction loss $\mathcal{L}\text{grid}$ on face and edge coordinates and a cosine-based normal consistency loss $\mathcal{L}\text{norm}$, comparing normals computed from predicted face grids with ground-truth normals. For geometric fitting, we triangulate face grids into surface meshes and define $\mathcal{L}\text{fit}$ as the sum of three point-to-surface distance terms: sampled context points to predicted faces, ground-truth boundary edge grids to predicted

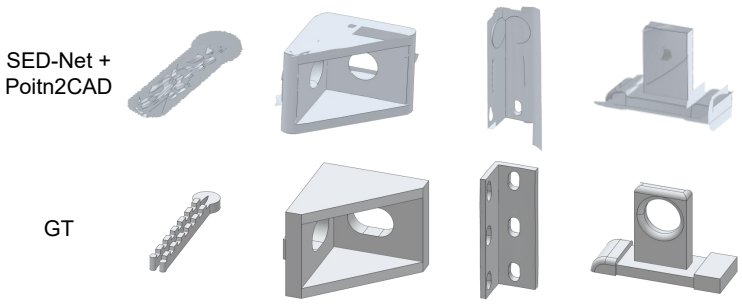

Figure 9: Qualitative results of SED-Net + Point2CAD on examples presented in the main paper.

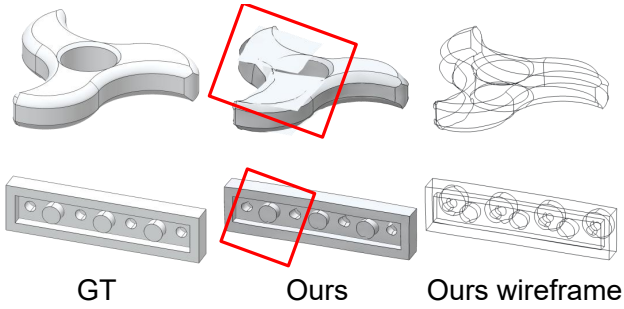

Figure 10: Failure cases involving incorrect surface trimming (top) and insufficient geometric smoothness (bottom). Red boxes highlight the affected regions.

faces, and predicted edge grids to their ground-truth incident faces. Edge existence is supervised using binary cross-entropy $\mathcal{L}\text{exist}$ over positive and negative candidates. The overall objective is $\mathcal{L}_{\text{recon}} = \lambda_{\text{grid}}\mathcal{L}_{\text{grid}} + \lambda_{\text{norm}}\mathcal{L}_{\text{norm}} + \lambda_{\text{fit}}\mathcal{L}_{\text{fit}} + \lambda_{\text{exist}}\mathcal{L}_{\text{exist}}$, where we empirically set $\lambda_{\text{grid}} = 2.0$, $\lambda_{\text{mask}} = \lambda_{\text{exist}} = 0.001$, $\lambda_{\text{norm}} = 1.0$, $\lambda_{\text{fit}} = 0.5$. Note that edge coordinate and fitting losses are applied only to positive edge candidates with ground-truth correspondences.

C Additional Results

SED-Net + Point2CAD results. As shown in Fig. 9, SED-Net Li et al. (2023) + Point2CAD Liu et al. (2024b) roughly recovers the overall shape but fails to preserve local geometry and structural details. The reconstructions exhibit fragmented or distorted surfaces, poorly aligned boundaries, and missing or incorrectly reconstructed holes. These errors are particularly apparent around repeated small features and openings, resulting in substantial deviations from the ground truth.

Failure cases. Fig. 10 illustrates two limitations of our method. In the first row, post-processing fails to recover the correct trimming boundaries, leaving a planar surface untrimmed and covering regions that should remain open. In the second row, small geometric details are reconstructed with insufficient accuracy, producing angular artifacts along boundaries that should be smooth.

For more qualitative results, please refer to the demo provided in the supplementary materials.

Statement

ChatGPT was used to assist with manuscript writing. CodeX was used to assist with coding, particularly for debugging and generating visualization results. The authors have carefully reviewed and verified all AI-assisted content and take full responsibility for scientific integrity of this work.

We provide the implementation detials needed to reproduce results in appendix. The code will be publicly released upon acceptance.