B-rep Guided Representation Learning for CAD Segmentation

Youngseok Hwang[1] Geonwoo Lee[2] Dayeon Kang[1] Jisung Son[3]
[1]Graduate School of Data Science, Seoul National University
[2]Department of Geography Education, Seoul National University
[3]Interdisciplinary Program in Artificial Intelligence, Seoul National University
{yshwang35, dayeon36}@snu.ac.kr dlzus42@snu.ac.kr jisung9973@snu.ac.kr

Abstract

Boundary representation (B-rep) is the most effective input modality for CAD segmentation, owing to its explicit face–edge–vertex topology. However, B-rep, which preserves geometric and topological cues for recovering design intent, is rarely available outside CAD authoring environments. Practical scenarios such as reverse engineering often provide only meshes or point clouds, making B-rep-based methods inapplicable. We propose a ***B-rep guided relational distillation framework*** *that transfers the structural knowledge of a pre-trained B-rep encoder into mesh and point-cloud student encoders. We perform Relational Knowledge Distillation (RKD), an objective that matches pairwise inter-surface relations rather than individual face features. Experiments on the Fusion 360 Gallery Segmentation dataset demonstrate consistent improvements over mesh-only and point-cloud-only baselines, confirming that B-rep topology serves as a meaningful supervisory signal for CAD segmentation, even when unavailable at inference.*

1. Introduction

Computer-Aided Design (CAD) underpins modern manufacturing and engineering, and recent deep learning advances have driven progress on CAD tasks including part classification, segmentation, reconstruction, and generative design [7, 9, 10, 22]. Among 3D representations, Boundary Representation (B-rep) stands out as the native format of CAD modeling software, encoding faces, edges, and vertices with explicit adjacency relations that naturally reflect design intent. B-rep-based methods [1, 7, 9] have consistently proven most effective on CAD benchmarks precisely because this structured encoding aligns with the semantic information the network is asked to recover.

However, B-rep is often unavailable in practice: reverse engineering of scanned parts, recovery of legacy designs whose source files are lost, and post-machining inspection

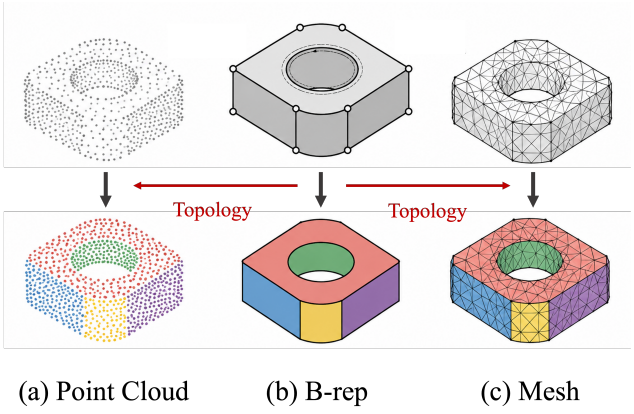

Figure 1. CAD segmentation on point cloud (left), B-rep (center), and mesh (right). Although B-rep provides the richest topology, our framework enables accurate segmentation on mesh and point cloud by transferring B-rep topology as supervisory signal during training.

all yield only meshes or point clouds. Neither representation retains the inter-face topology that B-rep preserves; mesh facets and sampled points are mere surface discretizations carrying no information about face adjacency or the semantic units they belong to. Existing CAD-specific methods [4, 14, 15] provide no mechanism to recover this lost structure. Point2CAD [10] attempts to bridge this gap via reconstruction, but its multi-stage pipeline makes topology recovery dependent on the quality of preceding segmentation and surface-fitting steps. Critically, such geometry-based segmentation cannot resolve faces that are geometrically identical yet produced by different operations, whose distinction resides in topological context rather than local shape.

Rather than reconstructing B-rep, we transfer its topological and relational structure directly into mesh and point-cloud encoders via knowledge distillation. The expressive power of B-rep networks lies not in per-face features but in the relations among faces that collectively encode design in-

tent; mimicking features alone therefore overlooks the relational structure that matters most. We thus adopt Relational Knowledge Distillation (RKD) [13], which aligns pairwise inter-surface relations rather than individual features. As shown in Figure 1, this enables accurate CAD segmentation on mesh and point cloud without B-rep at inference time. Overall, our contributions can be summarized as follows:

- We analyze why B-rep embeddings are an effective teacher, showing that they encode operation-level information that is provably inaccessible from per-face geometry alone.
- We propose a B-rep-guided relational distillation framework that transfers this topological knowledge from a B-rep teacher into mesh and point-cloud encoders for CAD segmentation.
- We demonstrate consistent improvements over mesh and point-cloud baselines on the Fusion 360 Segmentation dataset, without requiring B-rep at inference time.

2. Related Work

2.1. B-rep Representations Learning

B-rep encodes CAD solids as faces, edges, and vertices with explicit topological adjacency. Each B-rep face corresponds to a semantically meaningful surface patch reflecting design intent, making face-to-face adjacency a natural boundary cue for part segmentation. Early works established how to bring B-rep into neural pipelines: UV-Net [7] samples faces onto UV grids for CNN processing, while BRepNet [9] performs message passing over face-edge-coedge graphs to exploit topology explicitly. AAGNet [19] further combines geometric and topological cues via attribute adjacency graphs for machining feature recognition. FoV-Net [1] achieves state-of-the-art B-rep face segmentation through rotation-invariant surface encoding combined with graph attention, producing high-quality face-level embeddings. The structural richness of B-rep has even motivated reverse engineering approaches such as Point2CAD [10], which reconstructs B-rep directly from point clouds. However, B-rep remains unavailable outside CAD authoring environments, motivating the use of more accessible modalities as inference-time inputs.

2.2. Segmentation on Mesh and Point Cloud

Mesh-based methods exploit discrete surface connectivity for feature learning. MeshCNN [4] operates directly on mesh edges with convolution and edge-collapse pooling. PDMeshNet [12] extends this line by constructing coupled primal and dual graphs over a triangle mesh, where attention-based message passing captures interactions between face and edge structures and task-driven pooling progressively clusters mesh faces. DiffusionNet [17] learns diffusion processes over LBO eigenbases, achieving robustness to varying mesh resolution and triangulation.

Point-cloud based methods should infer structure from unordered point sets without explicit topology. PointNet [14, 15] established per-point and hierarchical local feature learning, followed by convolution-based methods such as KPConv [18]. The Point Transformer [20, 25] introduced attention-based aggregation, with PTv3 [21] achieving state-of-the-art performance via serialization-based hierarchical attention. In CAD segmentation, both modalities share a fundamental limitation. Neither preserves the face-level topology that defines semantically meaningful boundaries in B-rep. This loss of structural boundary information directly motivates our distillation framework.

2.3. Cross-Modal Knowledge Distillation

Knowledge distillation (KD) was originally introduced for model compression via soft label transfer [5], with extensions to intermediate features [16] and attention maps [24]. These approaches assume compatible feature layouts between teacher and student, limiting their applicability when inputs are heterogeneous. Cross-modal KD transfers supervision across modality boundaries [3], but modality gaps and misaligned supervision signals can limit direct feature imitation [6]. Relational KD [13] circumvents this by transferring pairwise structural relationships rather than absolute feature values, a principle that has proven effective in dense prediction [23] and cross-modal representation learning [11]. However, our setting introduces a further challenge: teacher and student differ not only in modality but also in granularity. We bridge this via area-weighted scatter aggregation before applying RKD, enabling students to inherit inter-surface relational structure from the B-rep teacher without requiring B-rep at inference time.

3. Preliminaries

We consider three 3D shape representations commonly used in CAD-related tasks: B-rep, Mesh, and Point Cloud. Table 1 summarizes their key differences in terms of structure, geometric fidelity, and practical accessibility. Notably, B-rep provides the richest structural information with explicit topology and analytic geometry, yet has the lowest accessibility, while Point Cloud is easily obtainable but carries no topological structure.

Table 1. Comparison of 3D shape representations in CAD domain.

	B-rep	**Mesh**	**Point Cloud**
Basic Unit	Face / Edge / Vertex	Triangle / Vertex	Point
Topology	Explicit	Implicit	None
Geometry	Analytic	Discrete	Estimated
CAD Fidelity	Complete	Partial	None
Face-level Seg.	✓	△	×
Accessibility	Low	Medium	High

Boundary Representation (B-rep). B-rep encodes a CAD model as a collection of faces, edges, and vertices with explicit topological adjacency. Each face corresponds to a semantically meaningful surface patch defined by analytic geometry, making B-rep the most information-rich representation for CAD segmentation.

Mesh. A triangle mesh approximates a 3D surface as a set of triangles sharing vertices and edges. While topology is implicitly encoded through traingle connectivity, the original B-rep face boundaries are blurred during tessellation, resulting in partial loss of semantic structure.

Point Cloud. A point cloud represents a 3D shape as an unordered set of points with spatial coordinates and estimated surface normals. It carries no explicit topology, making it the most accessible yet least structured representation for CAD tasks.

4. Why B-rep as a Teacher?

A face in a B-rep model is not fully characterized by its local surface geometry alone: the same geometric primitive can correspond to different modeling operations depending on its topological role in the shape. For instance, a planar face may be an extrusion cap, a cut cap, or a revolve cap, even though its isolated surface descriptor is nearly identical. Distinguishing such faces therefore requires information beyond per-face geometry. We verify, through qualitative, global, local, and supervised analyses, that the B-rep representation learned by FoV-Net indeed captures such information. This observation motivates the distillation framework presented in Section 5.

Geometry baseline. Throughout these analyses, we compare FoV-Net embeddings against a geometry baseline defined directly on the per-face descriptors that serve as input to FoV-Net: a 7-dimensional hand-crafted descriptor consisting of surface-type indicators (plane, cylinder, cone, sphere, torus), normalized area, and a rational-surface flag. Since this descriptor is computed independently per face, it captures local surface geometry but contains no face adjacency, loop structure, or other B-rep topological context. Because it is exactly the per-face input of FoV-Net, any information present in the FoV-Net embedding but absent from this baseline must originate from topological context.

Qualitative overview. As a first look, we project the FoV-Net face embeddings into 2D with t-SNE (Fig. 2). Faces of the same operation class form visible clusters, hinting that the embedding encodes operation-level structure. Since t-SNE distorts distances and is sensitive to its hyperparameters, we treat this only as a qualitative cue and quantify it in the analyses that follow.

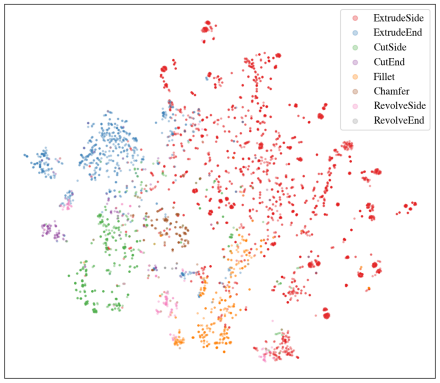

Figure 2. t-SNE visualization of FoV-Net face embeddings, colored by operation class.

Global structure. To probe global organization, we compute a per-class centroid by averaging its face embeddings and measure pairwise cosine similarity between centroids; lower off-diagonal similarity means more distinct classes (Fig. 3). FoV-Net yields moderate off-diagonal similarity, whereas the raw geometry baseline is highly similar for many class pairs, especially pairs that share a surface primitive (e.g., the side/end variants of the same operation). One might attribute this to anisotropy in unnormalized features, so we additionally apply ZCA whitening (Appendix A). Even after whitening, pairs that share a primitive but differ in operation remain highly similar, confirming that their overlap is structural, rooted in indistinguishable per-face geometry, rather than an artifact of feature scaling. This is precisely the ambiguity that face adjacency resolves and that we later distill.

Local structure. To test whether operation classes are locally separable, we perform parameter-free 1-nearest-neighbor classification, assigning each query face the label of its nearest face in the embedding space while excluding faces from the same shape (leave-one-shape-out) to prevent shape-level leakage. As shown in Fig. 4, FoV-Net forms substantially more discriminative neighborhoods, achieving higher accuracy than the geometry baseline across most classes. As 1-NN is non-parametric and nonlinear, this shows the baseline lacks the information needed to distinguish these operation classes in any form, not only in a linearly separable one.

Supervised separability. We further train a class-balanced linear probe on each representation. FoV-Net reaches a macro-F1 of 68.1%, versus 23.9% for the geome-

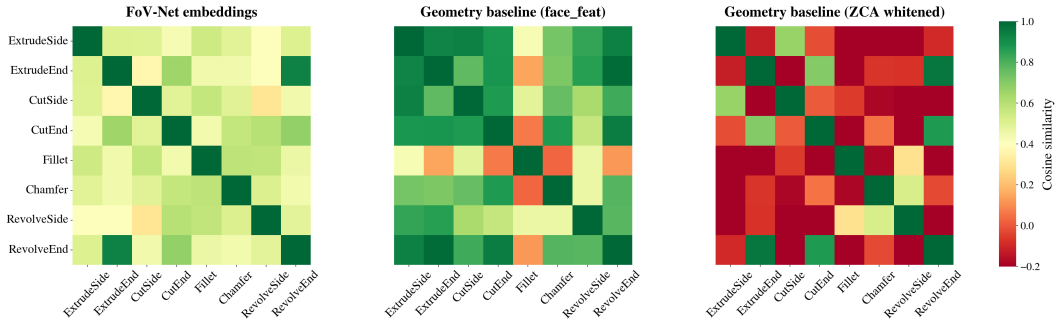

Figure 3. Inter-class centroid cosine similarity. **Left**: FoV-Net embeddings. **Center**: raw geometry baseline. **Right**: geometry baseline after ZCA whitening. Lower off-diagonal similarity indicates more distinct operation classes.

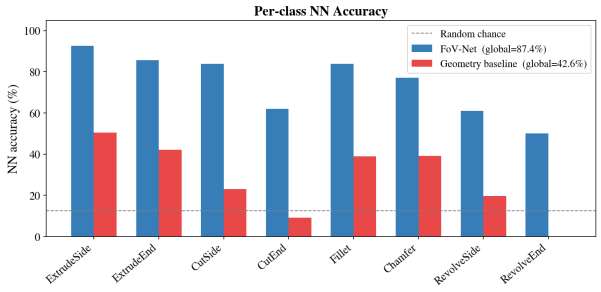

Figure 4. Leave-one-shape-out nearest-neighbor accuracy. FoV-Net substantially outperforms the geometry baseline across most operation classes.

try baseline. As Fig. 5 shows, the geometry baseline never predicts CutSide at all, leaving its column entirely empty: a cut side wall and an extruded side wall share the same per-face descriptor, collapsing into a single, indistinguishable input that no classifier can separate. FoV-Net instead resolves them from topological context.

Figure 5. Linear-probe confusion matrices. **Left**: FoV-Net embeddings. **Right**: geometry baseline. Cell colors indicate row-normalized recall; numbers denote raw counts.

Summary. Across all four analyses; qualitative (t-SNE), global (centroid similarity), local (nearest-neighbor), and supervised (linear-probe) operation-level information is consistently absent from per-face geometry yet present in the FoV-Net embedding. Together these results justify FoV-Net as the teacher whose relational structure we transfer.

5. Methodology

5.1. Problem Statement

Given a 3D CAD shape represented as either a mesh or a point cloud, our goal is to assign a semantic label $y \in \{0, \ldots, C-1\}$ to each primitive element—triangle or point—corresponding to one of C manufacturing operation classes. During training, a pre-trained B-rep encoder is additionally available as a teacher to provide structural supervision via knowledge distillation. At inference, only the mesh or point cloud input is required.

5.2. Overall Framework

Our framework trains two independent segmentation models, one for meshes and one for point clouds, each guided by a shared B-rep teacher through RKD. As illustrated in Figure 6, the B-rep encoder processes the same CAD model to produce face-level embeddings, which are transferred to the student encoders via B-rep Topology Transfer during training.

B-rep Encoder. FoV-Net [1] pre-trained on B-rep face segmentation serves as the teacher, producing per-face embeddings of shape $(F, 128)$. Teacher weights are frozen during student training.

Mesh Encoder. A triangle mesh is processed by DiffusionNet [17], which operates on learned diffusion over the LBO eigenbasis to produce per-triangle embeddings of shape $(T, 128)$.

Point Cloud Encoder. A point cloud of N points with XYZ coordinates and surface normals is processed by

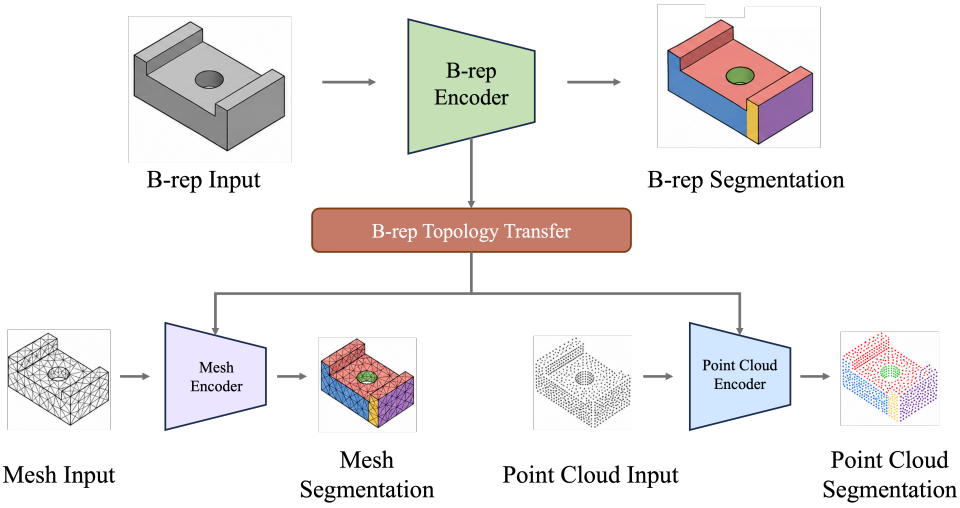

Figure 6. Overview of our framework. The B-rep encoder (top) processes the CAD model and transfers topology-aware relational structure to the mesh and point cloud student encoders (bottom) via relational knowledge distillation. At inference, only the student encoders are used for prediction.

PTv3 [21], producing per-point embeddings of shape $(N, 128)$ via serialization-based hierarchical attention.

Granularity Aligned Relational Distillation. Since student embeddings are defined at triangle or point level while teacher embeddings are at B-rep face level, we apply area-weighted scatter aggregation to map student embeddings to face granularity before computing RKD [13] loss over pairwise distances and angles.

5.3. Training Objective

We train each student encoder with a combination of cross-entropy segmentation loss and relational knowledge distillation loss from the frozen FoV-Net teacher:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda\, \mathcal{L}_{\text{RKD}} \tag{1}$$

$\mathcal{L}_{\text{CE}}$ is standard cross-entropy over per-element logits. For $\mathcal{L}_{\text{RKD}}$, student embeddings are first aggregated to B-rep face level via area-weighted scatter, then distance-wise and angle-wise relational losses [13] are applied between student and teacher face embeddings. λ is set to 1.0. At inference, the B-rep teacher is discarded and only the student encoder with a segmentation head is used.

$$\mathcal{L}_{\text{RKD}} = \lambda_{\text{D}}\mathcal{L}_{\text{D}} + \lambda_{\text{A}}\mathcal{L}_{\text{A}} \tag{2}$$

The distance-wise loss $\mathcal{L}_D$ penalizes differences in pairwise Euclidean distances between face embeddings, normalized by their mean:

$$\mathcal{L}_D = \sum_{(i,j)} \ell_\delta\left(\frac{\|\mathbf{z}_i^S - \mathbf{z}_j^S\|_2}{\mu^S}, \frac{\|\mathbf{z}_i^T - \mathbf{z}_j^T\|_2}{\mu^T} \right) \tag{3}$$

where μ^S, μ^T are the mean pairwise distances of the student and teacher, respectively, and ℓ_δ denotes the Huber loss. The angle-wise loss $\mathcal{L}_A$ penalizes differences in the cosine of angles formed by face embedding triplets:

$$\mathcal{L}_A = \sum_{(i,j,k)} \ell_\delta\big(\cos\theta_{ijk}^S,\ \cos\theta_{ijk}^T\big) \tag{4}$$

where $\cos\theta_{ijk} = \langle \hat{e}_{ij}, \hat{e}_{ik} \rangle$, $\hat{e}_{ij} = (\mathbf{z}_i - \mathbf{z}_j) \,/\, \|\mathbf{z}_i - \mathbf{z}_j\|$, and the sum is over all distinct triplets (i, j, k).

6. Experiments

We evaluate our framework on the Fusion 360 Gallery Segmentation Dataset [9], which uniquely provides B-rep, mesh, and point-cloud representations of the same CAD models, enabling aligned teacher–student training across all three modalities. Crucially, the mesh and point-cloud representations are tessellations and surface samplings of the same B-rep solids, where every triangle and sampled point retains its originating B-rep face index, yielding an explicit per-face correspondence across all three representations.

6.1. Dataset

The Fusion 360 Gallery Segmentation Dataset comprises 35,680 CAD models annotated with per-face labels for eight modeling operation classes. These classes cover the most common CAD operations (extrude, fillet, chamfer, revolve) with side/end distinctions for directional operations. As summarized in Table 2, the dataset is split into train/val/test sets at a 70:15:15 ratio, covering a total of 526,069 B-rep faces. As shown in Table 3, the dataset exhibits significant class imbalance, with ExtrudeSide accounting for 50.4%

of all faces while RevolveEnd comprises only 0.1%. This skew reflects the natural frequency of modeling operations in real-world CAD design.

Table 2. Fusion 360 Gallery Segmentation Dataset statistics.

Split	Objects	B-rep Faces
Train	24,976	368,713
Val	5,352	79,796
Test	5,352	77,560
Total	35,680	526,069

Table 3. Class distribution of Fusion 360 Gallery Segmentation Dataset (B-rep face level).

Class	Train	Val	Test	Total	Train %
ExtrudeSide	185,736	41,268	39,943	266,947	50.4%
ExtrudeEnd	57,740	12,801	12,396	82,937	15.7%
CutSide	48,289	10,033	9,232	67,554	13.1%
Fillet	38,125	6,993	7,631	52,749	10.3%
RevolveSide	19,385	4,007	4,262	27,654	5.3%
Chamfer	10,456	2,746	2,276	15,478	2.8%
CutEnd	8,674	1,873	1,768	12,315	2.4%
RevolveEnd	308	75	52	435	0.1%

6.2. Metric

We evaluate using three complementary metrics: per-face accuracy, frequency-weighted IoU (fwIoU), and mean Intersection over Union (mIoU). Given the significant class imbalance in our dataset, we report all three to capture different aspects of performance. *Accuracy* reflects overall correctness but is dominated by the majority classes and can mask poor performance on rare ones. *fwIoU* weights each class by its prevalence, providing a frequency-aware summary that credits performance on common classes while still penalizing systematic failures. *mIoU* weights every class equally regardless of its frequency, and therefore exposes performance on rare classes. Reporting all three guards against any single metric giving a misleading impression.

For each class c, IoU is computed as

$$\mathrm{IoU}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c}, \tag{5}$$

where TP_c, FP_c, and FN_c denote the per-class true positives, false positives, and false negatives.

Per-face accuracy is the fraction of correctly classified faces over all N faces:

$$\mathrm{Accuracy} = \frac{\sum_{c=0}^{C-1} \mathrm{TP}_c}{N}. \tag{6}$$

The frequency-weighted IoU weights each class by the number of faces N_c belonging to it (equivalently, $N_c = \mathrm{TP}_c + \mathrm{FN}_c$):

$$\mathrm{fwIoU} = \frac{1}{N} \sum_{c=0}^{C-1} N_c \, \mathrm{IoU}_c, \qquad N = \sum_{c=0}^{C-1} N_c. \tag{7}$$

The mean IoU averages the per-class IoU with equal weight:

$$\mathrm{mIoU} = \frac{1}{C} \sum_{c=0}^{C-1} \mathrm{IoU}_c. \tag{8}$$

6.3. Experimental Setup

All experiments are implemented in PyTorch and trained for up to 100 epochs with AdamW (learning rate 1×10^{-3}, weight decay 1×10^{-4}, decayed by 0.5 every 25 epochs), selecting the best epoch by validation mIoU. Unlike conventional RKD [13], we form batches at the object level, so the Granularity Aligned Relational Distillation operates within this dynamically sized batch. We set the RKD loss weight $\lambda = 1$, with $\lambda_{\mathrm{D}} = 1$ and $\lambda_{\mathrm{A}} = 2$. All experiments are evaluated on the Fusion 360 Gallery test split using per-face accuracy, fwIoU, and mIoU, on NVIDIA RTX 3090 and H100 GPUs.

6.4. Results

B-rep Segmentation. As shown in Table 4, FoV-Net achieves 89.29% accuracy, 81.07% fwIoU, and 63.47% mIoU on the Fusion 360 Gallery test split, serving as the B-rep teacher in our distillation framework. Its per-class IoU ranges from 87.42% on ExtrudeSide down to 3.85% on RevolveEnd, the rarest class at only 0.1% of faces, where the teacher itself fails due to extreme scarcity, upper-bounding what distillation can transfer for this class.

Table 4. Per-class results for the FoV-Net B-rep teacher on the Fusion 360 Gallery test split.

Class	Acc (%)	IoU (%)
ExtrudeSide	93.82	87.42
ExtrudeEnd	87.94	82.09
CutSide	86.73	74.51
CutEnd	66.40	56.71
Fillet	81.60	71.63
Chamfer	86.78	59.29
RevolveSide	81.89	72.29
RevolveEnd	3.85	3.85
Overall (Acc / mIoU)	89.29	63.47

Main Results. Table 5 summarizes per-face performance for both student modalities. RKD improves all three metrics for both the mesh (DiffusionNet) and point-cloud (PTv3) students, though the gain profile differs by modality. For

Method	Per-face Acc (%)	Per-face fwIoU (%)	Per-face mIoU (%)
DiffusionNet	64.94	46.15	24.40
DiffusionNet + RKD (ours)	**68.21**	**51.17**	**29.93**
PTv3	65.46	45.88	23.82
PTv3 + RKD (ours)	**65.76**	**49.80**	**33.05**

Table 5. Per-face segmentation performance on the Fusion 360 Gallery dataset, comparing DiffusionNet (mesh) and PTv3 (point cloud) students with and without RKD.

the mesh student, the improvement is broad and balanced: accuracy, fwIoU, and mIoU all rise together, with the two IoU-based metrics gaining the most. For the point-cloud student, accuracy is already near its baseline and changes little, while fwIoU and especially mIoU improve substantially, the latter showing the largest single gain across both modalities. In both cases the improvement concentrates on mIoU, indicating that the transferred signal primarily benefits the rare, topology-dependent classes whose operation cannot be inferred from local geometry. We break these results down per class below.

Mesh Segmentation. Table 6 breaks down the mesh gains per class, showing consistent improvements on most operation classes. The largest IoU gains appear on classes whose operation cannot be inferred from local shape alone. In particular, IoU of Fillet increases from 30.62% to 40.32%, Chamfer from 0.23% to 8.27%, and RevolveSide from 29.34% to 37.09%. Fillet and Chamfer are transition faces that border two primary faces rather than forming standalone primitives, so their identity depends on neighborhood context; the gains indicate that distillation supplies exactly this topological context. The only exception is RevolveEnd, the rarest class (0.1% of faces), which remains unresolved in both settings due to its scarcity.

Table 6. Per-class results for mesh segmentation at B-rep face level (Diffusion distilled without (w/o) and with (w/) B-rep knowledge distillation).

		Acc (%)		IoU (%)	
Class	# of Meshes	w/o	w/	w/o	w/
ExtrudeSide	4,835,396	**89.53**	87.83	64.27	**68.34**
ExtrudeEnd	3,922,157	75.02	**76.49**	45.24	**48.16**
RevolveSide	967,660	37.60	**57.52**	29.34	**37.09**
Fillet	441,384	36.50	**52.21**	30.62	**40.32**
CutSide	434,793	11.27	**18.64**	10.55	**16.62**
CutEnd	134,569	10.50	**18.07**	9.45	**15.20**
Chamfer	91,334	0.23	**8.39**	0.23	**8.27**
RevolveEnd	17,713	5.48	5.48	**5.48**	5.41
	10,845,006	72.48	**75.13**	24.40	**29.93**

Point Cloud Segmentation. Table 7 shows a similar trend for PTv3 baseline. B-rep knowledge distillation substantially improves per-face mIoU from 23.82% to 33.05%, while leaving the overall accuracy nearly unchanged. In particular, IoU of RevolveSide increases from 0.56% to 28.03%, CutSide from 2.20% to 26.01%, and CutEnd from 0.00% to 13.54%. These gains indicate that B-rep teacher provides relational and topological cues that help PTv3 recover rotational and cut-induced attributes, rather than merely acting as a generic accuracy booster. Consistent with mesh results, RevolveEnd remains the only exception, likely because its severe class scarcity limits effective supervision in both settings.

Table 7. Per-class results for point cloud segmentation at B-rep surface level (PTv3 distilled without (w/o) and with (w/) B-rep knowledge distillation).

		Acc (%)		IoU (%)	
Class	# of Points	w/o	w/	w/o	w/
ExtrudeSide	4,970,053	**90.80**	80.20	**63.08**	61.21
ExtrudeEnd	4,045,668	**73.17**	65.46	**49.83**	48.72
RevolveSide	1,038,988	0.56	**38.93**	0.56	**28.03**
CutSide	401,366	2.30	**44.14**	2.20	**26.01**
Fillet	279,841	**55.60**	47.46	42.57	**43.41**
CutEnd	136,781	0.00	**18.95**	0.00	**13.54**
Chamfer	76,093	41.74	**51.10**	32.33	**43.49**
RevolveEnd	12,106	0.00	0.00	0.00	0.00
	10,960,896	65.46	**65.76**	23.82	**33.05**

7. Representational Transfer

We further examine whether RKD transfers the teacher's relational representation, rather than only improving the student's output predictions. Since RKD is designed to align the relative structure of hidden embeddings, output-level metrics alone do not reveal whether the student actually inherits the teacher's representation geometry. We therefore measure the similarity between the student and teacher embedding spaces using Centered Kernel Alignment (CKA) [8]. CKA compares the pairwise relational structure of representations through centered Gram matrices, making it suitable for analyzing whether two networks

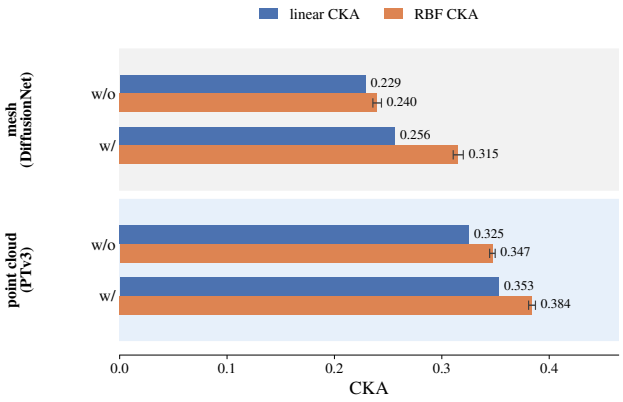

Figure 7. Linear and RBF CKA to FoV-Net, over 77,560 test B-rep faces. RKD increases representational similarity to the teacher under both kernels for both mesh and point cloud student.

organize the same samples similarly even when their architectures or input modalities differ.

For each student modality, we compute CKA against the B-rep teacher, FoV-Net. The teacher operates on B-rep surfaces and produces face-level embeddings. Since the mesh and point-cloud students operate on non-B-rep inputs, we map their features back to B-rep face level for a sample-aligned comparison. This yields one 128-dimensional student embedding per B-rep face, aligned with the FoV-Net teacher embedding. We compute CKA over all $N = 77{,}560$ test faces. We report linear CKA on the full set of faces and RBF-kernel CKA averaged over five random subsets of $5{,}000$ faces. For RBF CKA, we set the bandwidth using the median heuristic [2], which provides a scale-adaptive kernel bandwidth from the empirical pairwise distances without tuning on labels.

7.1. Mesh

As shown in Fig. 7, RKD increases the representational similarity between the mesh student and the B-rep teacher under both kernels. Linear CKA improves from 0.229 to 0.256, while RBF CKA improves from 0.240 to 0.315. Because the teacher and student use different architectures and different input modalities, the absolute CKA values are not expected to be close to one. Therefore, the important observation is the consistent relative increase after RKD, which isolates the effect of distillation on the student's hidden representation.

The larger improvement under the RBF kernel further suggests that RKD transfers more than a linearly aligned subspace. Instead, it improves the nonlinear geometry of the student's embedding space with respect to the FoV-Net teacher. This provides representation-level evidence that RKD pulls the mesh student's hidden features toward the B-rep teacher's relational structure, improving the face-level downstream segmentation tasks.

7.2. Point Cloud

We observe the same trend for the point-cloud student in Fig. 7. RKD increases linear CKA from 0.325 to 0.353 and RBF CKA from 0.347 to 0.384. This shows that the effect of RKD is not specific to the mesh backbone. Although a point-cloud architecture receives a different discretization of the same shape, distillation moves the learned representation closer to the FoV-Net teacher.

Compared with the mesh student, PTv3 starts from a higher CKA to the teacher, suggesting that its baseline representation is already more aligned with the B-rep teacher at the face-aggregated level. Nevertheless, RKD still yields a clear improvement under both linear and nonlinear kernels. The gain is again larger for RBF CKA than for linear CKA, indicating that RKD improves not only global linear alignment but also the nonlinear relational geometry of the point-cloud embedding space. These supports the conclusion that RKD transfers B-rep induced representational structure across multiple student modalities rather than merely reshaping their output distributions.

8. Conclusion

We presented a B-rep guided relational distillation framework for CAD segmentation with mesh and point-cloud inputs. Our method transfers inter-surface relational knowledge from a pre-trained B-rep teacher to non-B-rep student encoders through RKD, enabling topology-aware representation learning without requiring B-rep at inference. Experiments on the Fusion 360 Gallery Segmentation Dataset show that our approach improves mIoU over DiffusionNet and PTv3 baselines, with especially notable gains on minority classes. This indicates that B-rep relational structure provides supervisory cues difficult to recover from surface discretizations alone, demonstrating the effectiveness of distilling B-rep topology into mesh and point-cloud encoders for B-rep-free CAD segmentation.

9. Limitations and Future Work

Our study has several limitations. First, all experiments use a single dataset, as datasets providing all three modalities are scarce; transfer to other CAD datasets remains to be verified. Second, rare and geometrically ambiguous classes such as RevolveEnd (308 training samples) are unsolved by both teacher and students; since information absent from the teacher cannot be distilled, these classes likely require a stronger teacher, a refined taxonomy, or an LLM-based post-hoc refinement over predicted labels and adjacency. Third, our loss entangles the geometric and topological components of the transferred knowledge, which future work could disentangle through separate KD terms.

References

[1] Matteo Ballegeer and Dries F Benoit. Fov-net: Rotation-invariant cad b-rep learning via field-of-view ray casting. *arXiv preprint arXiv:2602.24084*, 2026. 1, 2, 4

[2] Damien Garreau, Wittawat Jitkrittum, and Motonobu Kanagawa. Large sample analysis of the median heuristic, 2018. 8

[3] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2827–2836, 2016. 2

[4] Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. Meshcnn: a network with an edge. *ACM Transactions on Graphics (ToG)*, 38(4):1–12, 2019. 1, 2

[5] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[6] Fushuo Huo, Wenchao Xu, Jingcai Guo, Haozhao Wang, and Song Guo. C2kd: Bridging the modality gap for cross-modal knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16006–16015, 2024. 2

[7] Pradeep Kumar Jayaraman, Aditya Sanghi, Joseph G Lambourne, Karl DD Willis, Thomas Davies, Hooman Shayani, and Nigel Morris. Uv-net: Learning from boundary representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11703–11712, 2021. 1, 2

[8] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMlR, 2019. 7

[9] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12773–12782, 2021. 1, 2, 5

[10] Yujia Liu, Anton Obukhov, Jan Dirk Wegner, and Konrad Schindler. Point2cad: Reverse engineering cad models from 3d point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3763–3772, 2024. 1, 2

[11] Yunyao Mao, Wengang Zhou, Zhenbo Lu, Jiajun Deng, and Houqiang Li. Cmd: Self-supervised 3d action representation learning with cross-modal mutual distillation. In *European Conference on Computer Vision*, pages 734–752. Springer, 2022. 2

[12] Francesco Milano, Antonio Loquercio, Antoni Rosinol, Davide Scaramuzza, and Luca Carlone. Primal-dual mesh convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:952–963, 2020. 2

[13] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976, 2019. 2, 5, 6

[14] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1, 2

[15] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 1, 2

[16] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015. 2

[17] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (ToG)*, 41(3): 1–16, 2022. 2, 4

[18] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 2

[19] Hongjin Wu, Ruoshan Lei, Yibing Peng, and Liang Gao. Aagnet: A graph neural network towards multi-task machining feature recognition. *Robotics and Computer-Integrated Manufacturing*, 86:102661, 2024. 2

[20] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. *Advances in Neural Information Processing Systems*, 35:33330–33342, 2022. 2

[21] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4840–4851, 2024. 2, 5

[22] 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, 2024. 1

[23] Chuanguang Yang, Helong Zhou, Zhulin An, Xue Jiang, Yongjun Xu, and Qian Zhang. Cross-image relational knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12319–12328, 2022. 2

[24] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017. 2

[25] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 2

A. Additional Analyses

A.1. Embedding Analyses

Hard-pair nearest-neighbor accuracy. Figure 8 reports binary nearest-neighbor accuracy on pairs of operations that share a surface primitive but differ in modeling operation. The geometry baseline drops to near binary chance on these pairs, most severely on ExtrudeEnd vs. RevolveEnd, whereas FoV-Net stays well above it, isolating the cases where per-face geometry is fundamentally uninformative.

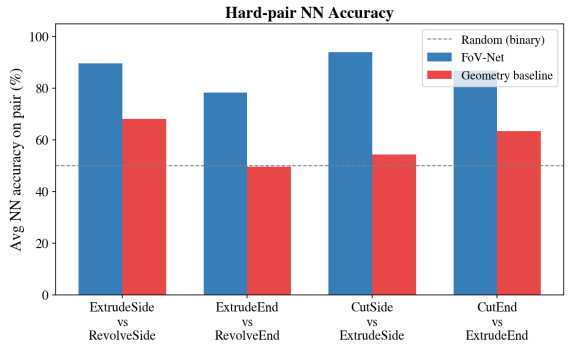

Figure 8. Hard-pair nearest-neighbor accuracy on operation pairs sharing a surface primitive. The dashed line marks binary chance.

ZCA whitening. To verify that the high centroid similarity of the geometry baseline is not merely an anisotropy artifact, we whiten the features before computing cosine similarity. Given centered features with empirical covariance Σ, ZCA whitening applies the linear transform

$$W = (\Sigma + \epsilon I)^{-1/2}, \qquad \tilde{x} = Wx, \tag{9}$$

with a small regularization term $\epsilon = 10^{-5}$ for numerical stability. In practice we compute W from the singular value decomposition $\Sigma = USV^\top$ as $W = U\,(S + \epsilon I)^{-1/2}V^\top$. This yields (approximately) identity covariance, $\mathrm{Cov}(\tilde{x}) \approx I$, so the variance is equalized across all directions. Unlike PCA whitening, ZCA uses the symmetric inverse square root and therefore preserves the original feature axes rather than rotating them, keeping each dimension interpretable. Applying this transform removes any scaling-induced inflation of cosine similarity; the residual similarity between operation pairs that share a surface primitive is thus attributable to genuine geometric indistinguishability rather than feature anisotropy.

A.2. Qualitative Segmentation Results

We present qualitative results on five randomly sampled test shapes (Figs. 9–13), each comparing ground truth, the baseline (w/o RKD), and our distilled model (w/ RKD) for both students. Across these examples, RKD corrects large mislabeled regions rather than scattered errors, and the corrections concentrate on operation pairs that share a surface primitive and are thus geometrically ambiguous. The two modalities behave differently: the mesh student, operating on connected faces, tends to recover coherent surface regions and adjacency-defined classes such as Fillet, whereas the point-cloud student, lacking explicit connectivity, benefits most on classes that local geometry cannot resolve, such as RevolveSide and CutSide, where the relational signal compensates for its missing topology. We detail each shape below.

The first shape (Fig. 9) is a flat link with planar top and side faces. The mesh baseline confuses ExtrudeSide and ExtrudeEnd, the two most frequent classes (50.4%, 15.7%) that differ only in extrusion role, and RKD cleanly separates them. The baseline also misses the thin Chamfer along the edges, which borders two primary faces rather than forming a standalone primitive and thus depends on neighborhood context; RKD recovers it. The second shape (Fig. 10) has a thin Fillet band that the mesh baseline absorbs into the neighboring ExtrudeEnd. As a fillet is a transition surface defined by its neighbors, RKD recovers this adjacency-dependent class (10.3%). The third shape (Fig. 11) is a hemispherical bowl. The point-cloud baseline confuses ExtrudeSide and CutSide, distinguished only by whether material was removed, and recovers CutSide after RKD; the mesh student likewise delineates the Fillet ring more faithfully. The fourth shape (Fig. 12) is a bracket whose cylindrical CutSide (13.1%) is surface-identical to an extruded cylinder. Both students initially mislabel it as ExtrudeSide and correctly assign CutSide after RKD, showing the signal benefits both modalities on the same class. The fifth shape (Fig. 13) is a stepped revolve with thin Chamfer rings. After RKD the mesh student recovers the RevolveSide body but misses the rare Chamfer rings (2.8%), whereas the point-cloud student recovers both, closely matching ground truth, showing the two modalities distribute the recovered signal differently across rare classes.

Overall, these examples confirm that RKD's gains come from correcting structurally coherent regions whose operation is ambiguous from local geometry alone, rather than from scattered, low-level label noise. This is consistent with the per-class quantitative results, where the largest improvements appear on geometrically ambiguous classes such as CutSide, RevolveSide, and Fillet.

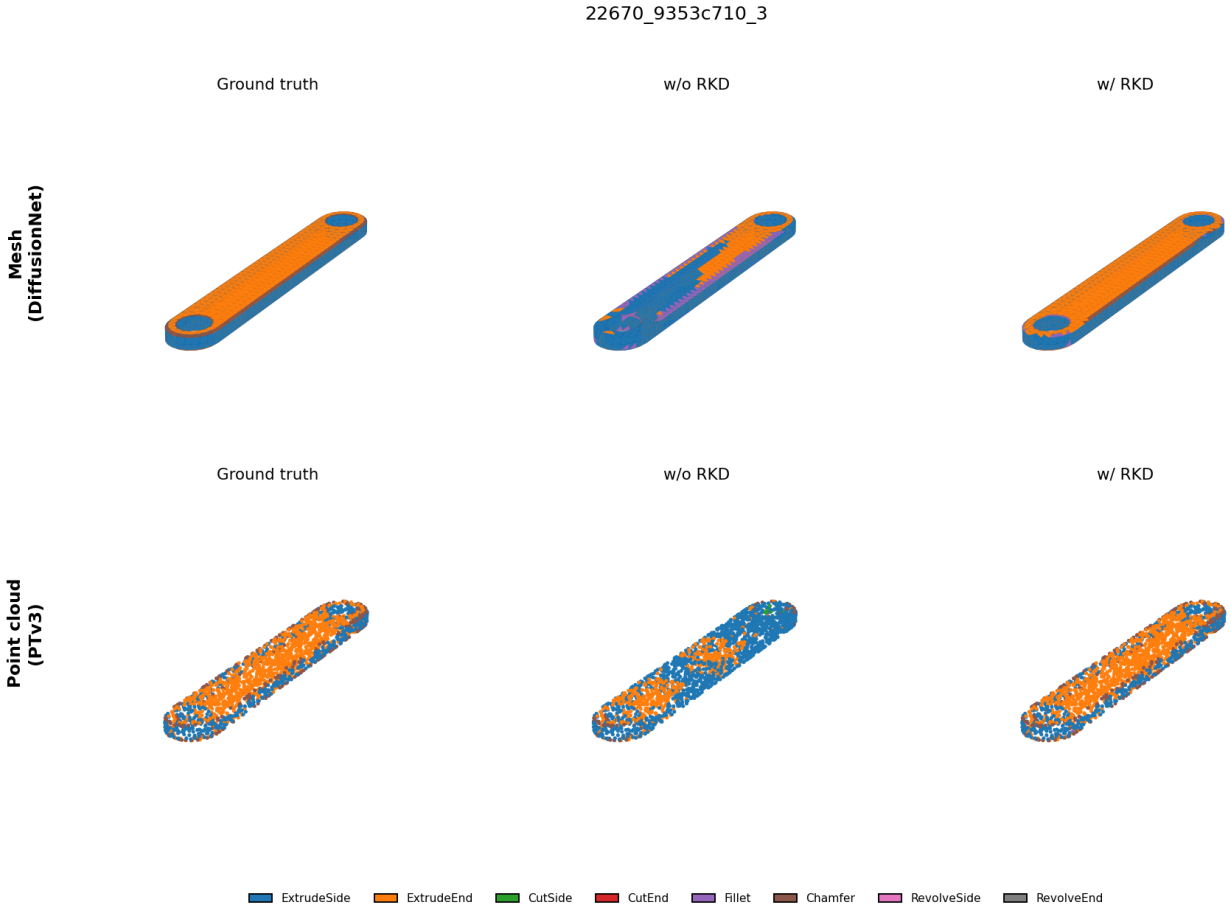

Figure 9. Qualitative segmentation on shape 22670: ground truth, w/o RKD, and w/ RKD, for the mesh (top) and point-cloud (bottom) students.

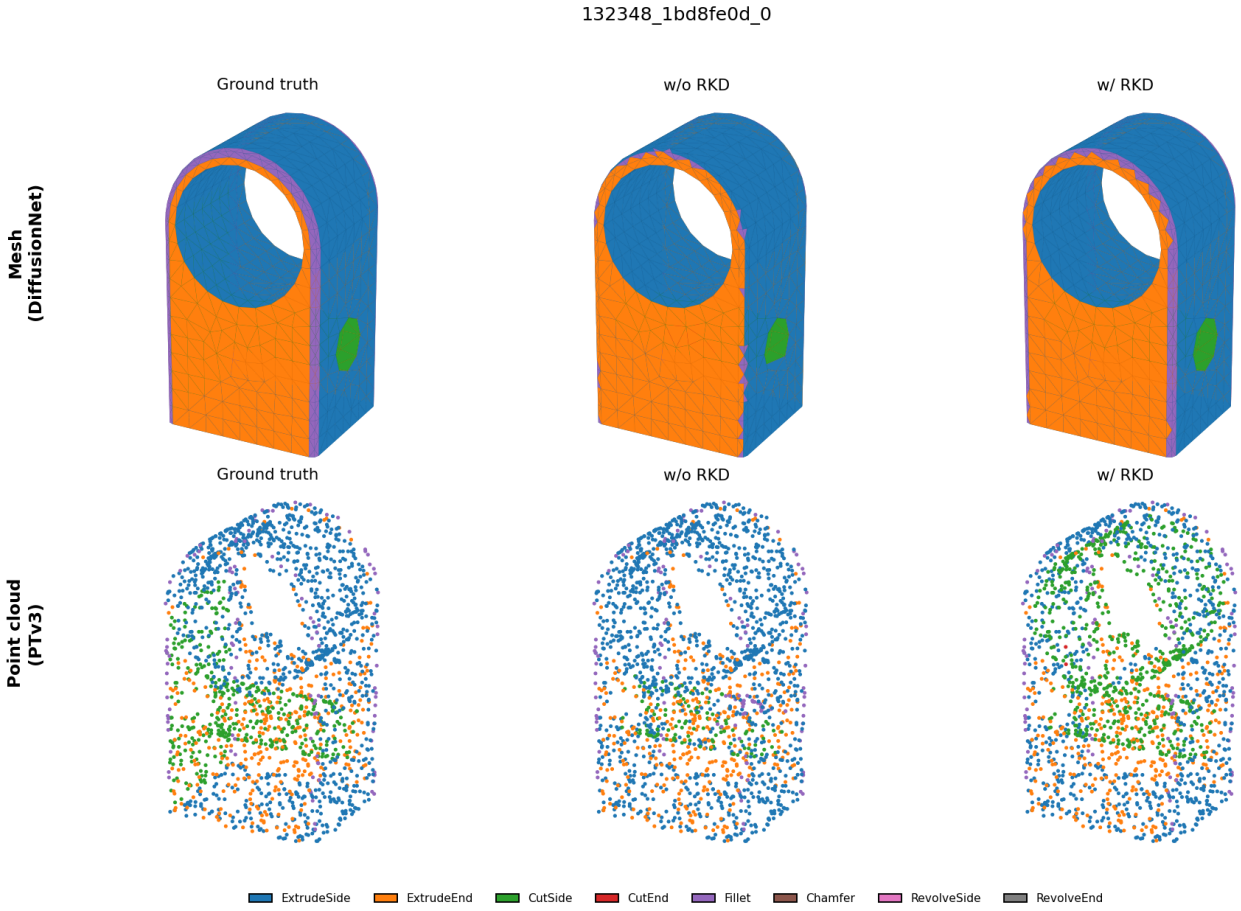

Figure 10. Qualitative segmentation on shape 132348: ground truth, w/o RKD, and w/ RKD, for the mesh (top) and point-cloud (bottom) students.

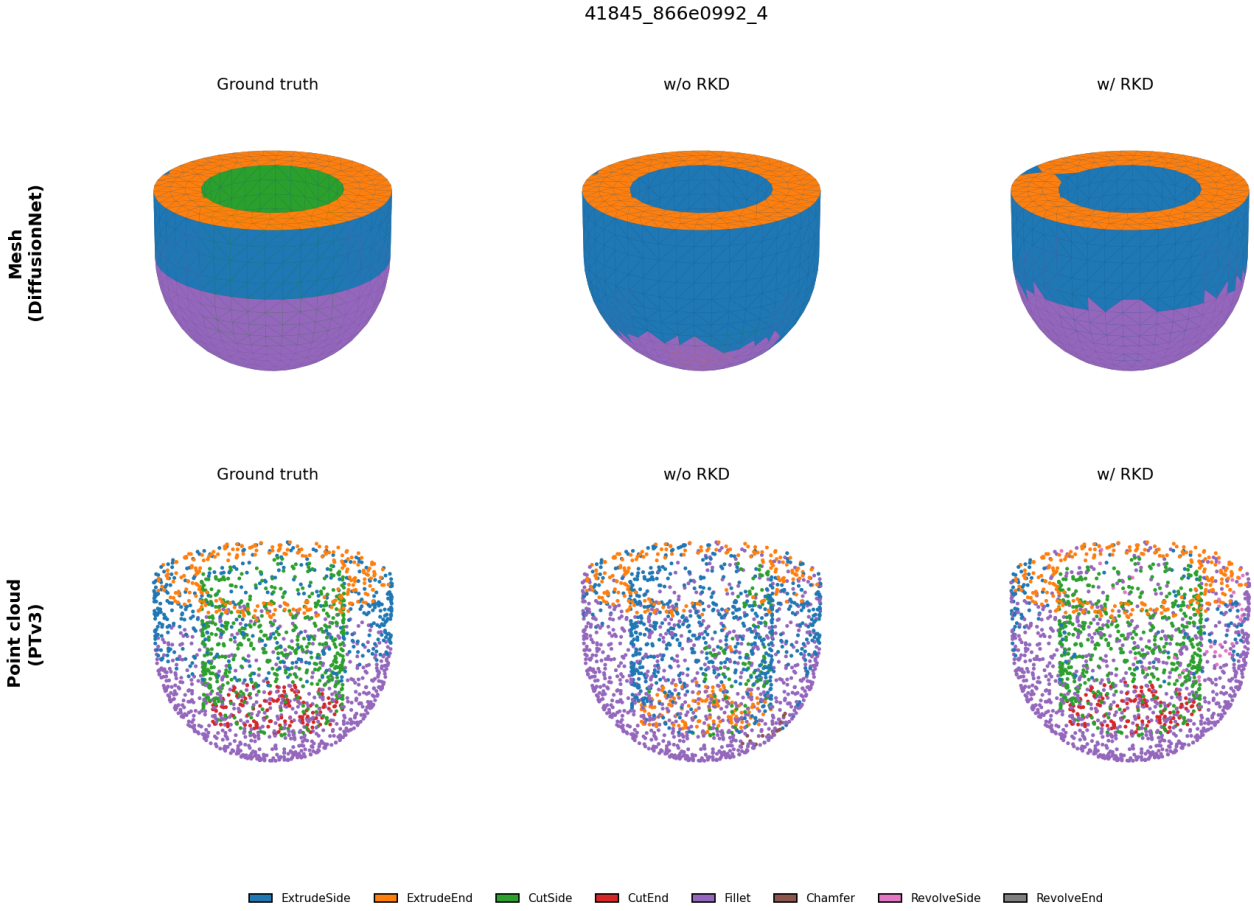

Figure 11. Qualitative segmentation on shape 41845: ground truth, w/o RKD, and w/ RKD, for the mesh (top) and point-cloud (bottom) students.

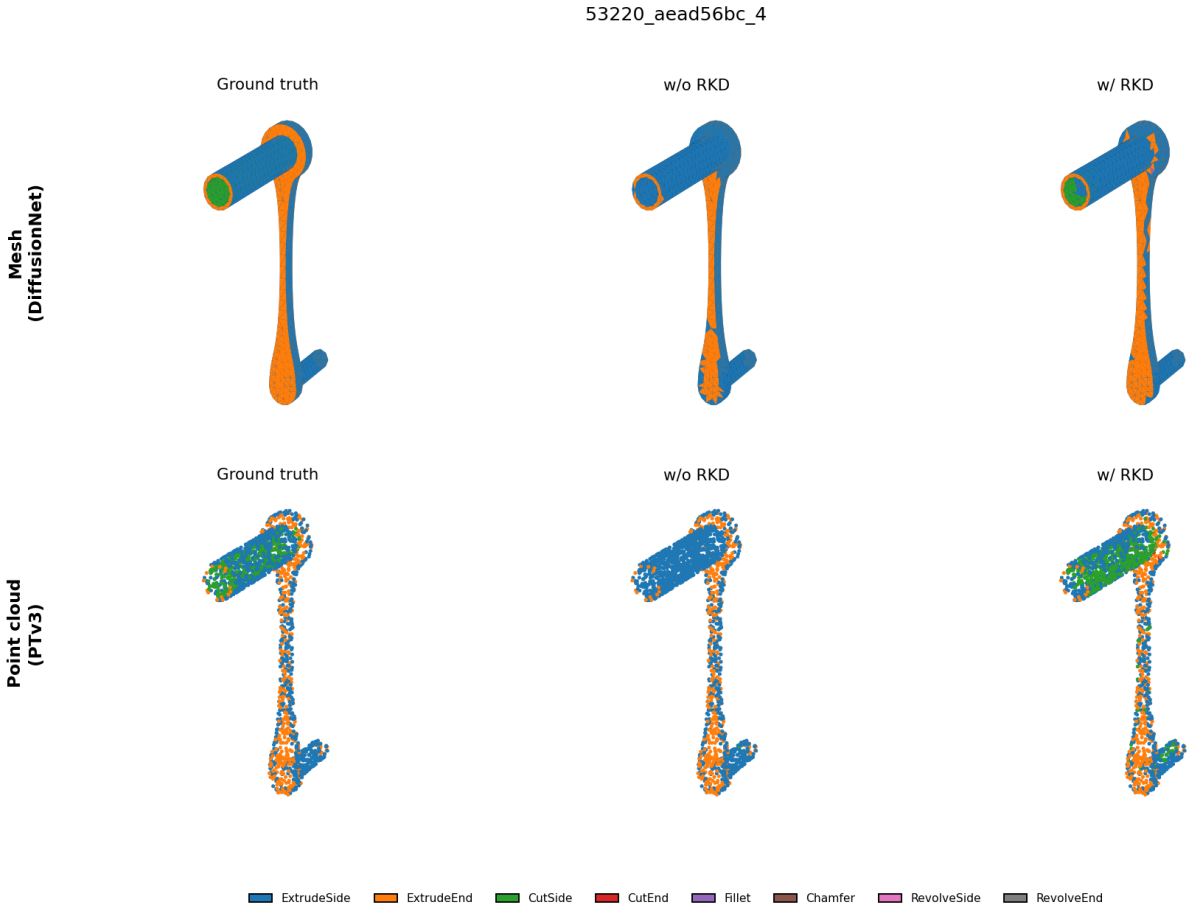

Figure 12. Qualitative segmentation on shape 53220: ground truth, w/o RKD, and w/ RKD, for the mesh (top) and point-cloud (bottom) students.

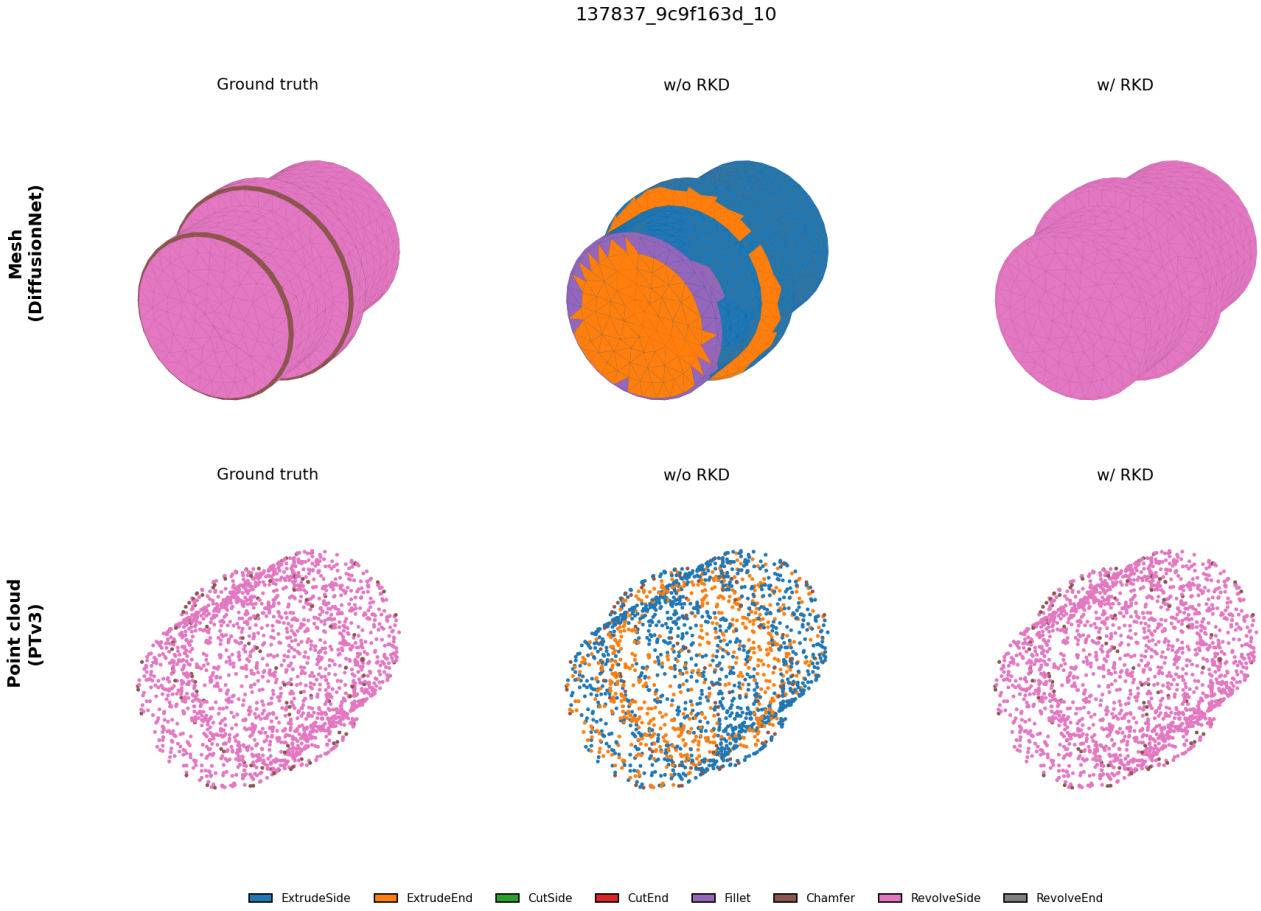

Figure 13. Qualitative segmentation on shape 137837: ground truth, w/o RKD, and w/ RKD, for the mesh (top) and point-cloud (bottom) students.