FEW-SHOT B-REP LOCAL STRUCTURE RETRIEVAL WITH STRUCTURE-AWARE PRETRAINING AND HARD-EXAMPLE FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing B-rep retrieval methods primarily model whole-part similarity, whereas engineering reuse often requires finding complete models that contain a desired local structure. This setting is challenging because target structures can vary substantially in geometry and topology, especially when a few query examples are available. We present LocBrep, a few-shot local-structure retrieval framework for B-rep models. It couples structure-aware pretraining and hard-example fine-tuning with a coarse-to-fine retriever to learn transferable local representations, accurately retrieve specific parts, and support retrieval for CAD assemble. LocBrep first employs BrepDenoise pretraining, which combines multi-scale neighborhood consistency with topological positional regularization to obtain robust face-level representations. It then performs query-guided hard-example fine-tuning to adapt the encoder to a target structure from limited examples. For scalable retrieval, hyperedge-level coarse search recalls candidate parts, and a query-aware cross-graph reranker refines their ranking according to local structural correspondence. We further introduce LocBrep-Bench, comprising 4500 B-rep models from 50 local-structure categories with face-level annotations. LocBrep outperforms existing methods on FabWave and LocBrep-Bench under both zero-shot and one-shot settings, with up to a 15.71 percentage-point improvement in MAP. Experiments on Fusion 360 further validate its applicability to retrieval in CAD assemblies.

1 INTRODUCTION

Reusing design knowledge from existing Computer-Aided Design (CAD) repositories is central to efficient design process. In practice, engineers often seek not only an entire part, but also local structure, such as a hole, slot, boss, cavity and so on, that carries the geometric and manufacturing semantics relevant to a new task Bai et al. (2010); Ji et al. (2023). A useful retrieved model therefore need not be globally similar to the target design, but must contain a structurally compatible local region. Enabling such local-structure retrieval can thus reduce repetitive modeling and support feature reuse, process planning, and local design reference.

Existing CAD retrieval methods largely follow two lines. Recent learning-based methods, such as UVStyle-Net, FuS-GCN,and VGNetMeltzer et al. (2021); Hou et al. (2023); Qin et al. (2025a), encode complete CAD models into global representations, which cannot adequately capture whether a model contains a target local structure. In contrast, other methods query B-rep local structure through explicit decompositions into segmented local features or engineered multiscale interfacesTao et al. (2013); Li et al. (2015). However, their reliance on predefined representations and matching rules limits their ability to handle complex local structures with diverse geometry and topology. A learning-based framework is still needed for retrieving complex local structures from complete CAD models, especially when only a few query examples are available.

In this paper, we present LocBrep, the first learning-based framework for few-shot B-rep partial retrieval. Given a local B-rep query, it retrieves complete CAD models containing a geometrically and topologically compatible region. The framework consists of three stages: self-supervised B-rep pretraining, few-shot encoder adaptation, and a separate coarse-to-fine retriever. Pretraining learns structural priors from unlabeled CAD models, while adaptation specializes these priors using only a

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.

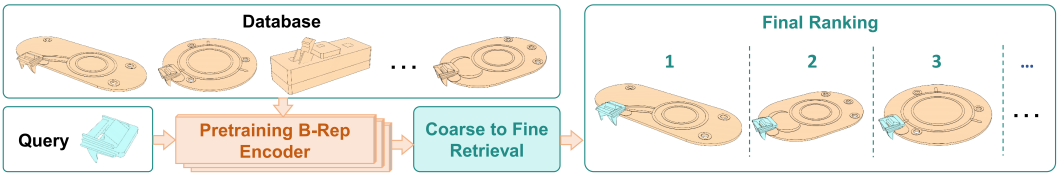


Figure 1: Given a query local structure and a CAD database, LocBrep performs coarse-to-fine retrieval to identify CAD models containing the target structure.

handful of query examples. This framework reflects practical engineering repositories, where CAD data are abundant but annotations for a newly required structure are scarce. Keeping the retriever independent of the encoder further enables flexible local matching and ranking, rather than tightly coupling retrieval behavior to a fixed embedding objective.

Specifically, we introduce BrepDenoise and hard mining for pretraining and fine-tuning respectively. BrepDenoise is a novel diffusion framework that denoises face-level features using multi-scale neighborhood consistency and topological positional regularization objectives. It thus learns transferable local B-rep representations, preserves stable neighborhood layouts and topological positions while deemphasizing model-specific face details, producing features that transfer across geometric variations. For few-shot fine-tuning, we designed label-free hard mining to select informative candidates while reducing false negatives, enabling label-free use of the downstream candidate database. For the retriever, we design it as a coarse-to-fine framework. The coarse retrieval stage represents each CAD part as a set of hyperedge-level local region vectors and retrieves candidate parts through hyperedge-level similarity search. The fine retrieval stage further re-ranks these candidates through cross-graph interaction matching. The reranker is trained on graph pairs constructed from pretraining data, allowing it to learn general local structure matching ability for downstream few-shot retrieval. To support systematic evaluation, we construct LocBrep-Bench, a compositional benchmark for few-shot B-rep CAD partial retrieval. It contains 4500 B-rep CAD models spanning 50 categories of local structures, with face-level annotations for target structure instances. In summary, the main contributions of this paper are as follows:

- We introduce LocBrep, a few-shot B-rep partial-retrieval framework that combines BrepDenoise pretraining with affinity- and uncertainty-guided fine-tuning to learn transferable local representations.
- We develop a coarse-to-fine retriever that couples hyperedge-level candidate recall with cross-graph reranking for efficient and accurate local-structure matching.
- We present LocBrep-Bench, a face-level annotated benchmark for few-shot CAD partial retrieval, and demonstrate state-of-the-art zero-shot and one-shot results on LocBrep-Bench and FabWave.

2 Related Work

2.1 CAD Retrieval

Recent CAD retrieval methods increasingly adopt CAD-native representations that preserve exact geometry and topology. UVStyle-Net, FuS-GCN, VGNet, and CADGCL learn whole-model representations through few-shot metric learning, B-rep graph aggregation, multimodal fusion, and graph contrastive learning, respectively Meltzer et al. (2021); Hou et al. (2023); Qin et al. (2025a;b). Ji et al.Ji et al. (2023) retrieve complete CAD models by using a bag-of-words model to match handcrafted shape descriptors.Unlike these methods based on explicit segmentation, descriptors, or interface representations, our method learns geometric-topological representations for few-shot B-rep local-structure queries.

2.2 CAD Graph Learning

B-rep models provide analytic geometry together with face, edge, and incidence topology, making graph learning a natural choice for CAD representation. BRepNet and UV-Net learn B-rep rep-

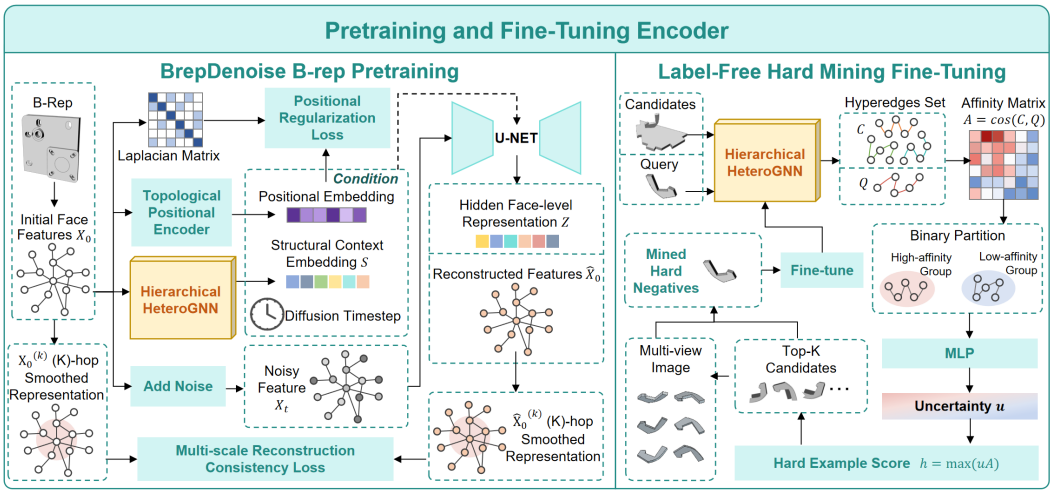


Figure 2: Overview of LocBrep encoder pretraining and hard-example fine-tuning. BrepDenoise pretrains the Hierarchical HeteroGNN backbone, and the mined hard examples are subsequently used to further update the same backbone encoder.

resentations through topological message passing and the fusion of UV-parametric geometry with adjacency relations, while CADOps-Net and BRep-BERT further explore CAD operation understanding and self-supervised pretrainingLambourne et al. (2021); Jayaraman et al. (2021); Dupont et al. (2022); Lou et al. (2023). Graph-based CAD learning has also been widely applied to machining feature understanding. Early methods used 3D convolutions, explicit B-rep graph representations, or STEP information for machining feature recognition Zhang et al. (2018); Cao et al. (2020); Malleswari & Pragvamsa (2023). Hierarchical CADNet, AAGNet, BrepMFR, and recent CAD graph models further learn face-level representations for recognition and segmentation in manufacturing contexts Colligan et al. (2022); Wu et al. (2024); Zhang et al. (2024); Li et al. (2025); Xia et al. (2024); Lee et al. (2023); Guo et al. (2025); Yao et al. (2026). These studies provide the foundation for encoding CAD faces and local feature structures in our retrieval framework.

2.3 CAD Datasets

Large-scale B-rep and CAD-native datasets are essential for learning and evaluating CAD representations. ABC provides precise CAD geometry at scale, SketchGraphs models relational sketch constraints, and DeepCAD and Fusion 360 Gallery provide construction sequences or operation-level annotations for programmatic CAD modeling Koch et al. (2019); Seff et al. (2020); Wu et al. (2021); Willis et al. (2021a); Zhou et al. (2023); Li et al. (2022). BRT introduces TMCAD, a STEP-based mechanical CAD dataset with more than 10,000 real-world B-rep models across ten part classes Zou & Zhu (2025). For machining and local feature analysis, MFCAD++, MFInstSeg, and MFTRCAD provide B-rep models with face-level or machining-feature annotations, while BRepFormer introduces CBF Dataset for complex geometric feature recognition Queen's University Belfast (2022); Wu et al. (2024); Xia et al. (2024); Dai et al. (2025). Collectively, these datasets support pretraining, local feature indexing, and retrieval evaluation for CAD representation learning.

3 Pretraining and Fine-Tuning Encoder

LocBrep adopts an encoder-retriever architecture. An overview of the encoder pretraining and fine-tuning framework is shown in Fig.2. In this section, we introduce our B-rep encoder based on a pretraining and fine-tuning framework. Specifically, we design a BrepDenoise-based B-rep pretraining model, which is specifically tailored to extract local features from 3D B-rep models, aligning with the objective of this study. We then propose a Label-Free Hard Mining Fine-Tuning strategy, which adapts the encoder to achieve strong retrieval performance using only a minimal set of unlabeled hard examples.

3.1 BREPDENOISE B-REP PRETRAINING

Focusing on the local features of 3D B-rep models, our pretraining adopts BrepDenoise, a smooth diffusion model based on local geometric faces, which smooths the feature of a given face with those of its adjacent faces, thereby better capturing local face-to-face relationships.

Specifically, we adopt the hybrid B-rep graph representation, i.e., Hierarchical HeteroGNN Hao et al. (2026), as the backbone encoder. A B-rep CAD model is represented as a hybrid graph G, where regional hyperedges are introduced beyond face nodes and adjacency edges to capture local structural semantics. A face feature encoder extracts geometric features to obtain the initial face feature matrix X_0. During pretraining, noises were introduced into the normalized face-level feature space. The perturbed representation at timestep t is denoted as X_t. To leverage the structural information of CAD graphs to guide the denoising process, we construct three types of condition. First, the diffusion timestep is mapped to a time embedding T_t, which represents the current noise intensity. Second, the clean face features X_0 and the hybrid B-rep graph G are fed into the backbone to obtain the structural context embedding S. This embedding integrates face adjacency relationships and regional hyperedge information, providing stable local structural conditions for the denoising network. Finally, random-walk arrival probabilities are computed on the face adjacency graph and passed through a topological positional encoder to obtain the positional embedding P, which describes the relative position of each face node in the B-rep topological graph. The noisy face features and the above conditions are then jointly fed into the denoising U-Net:

$$\hat{X}_0 = \mathrm{UNet}(X_t, T_t, S, P), \tag{1}$$

where $\hat{X}_0$ denotes the reconstructed clean face features. Detailed network architectures and pathway analyses of the three conditioning modules are provided in Appendix A.1. BrepDenoise adapts the low-frequency conditional diffusion formulation studied in SDMG Zhu et al. (2025) to face-level B-rep representation-learning tasks.

Unlike methods that enforce point-wise recovery of all face-level details, this paper adopts a neighborhood-structure consistency objective to emphasize stable local CAD structures. Repeated mean aggregation on the face adjacency graph yields the (k)-hop smoothed representations X_0^k and $\hat{X}_0^k$. For each scale (k), the similarity s^k is defined as:

$$s^k = \left(\frac{1 + \cos(X_0^k, \hat{X}_0^k)}{2} \right)^{\alpha}. \tag{2}$$

where α is the similarity sharpening coefficient. Finally, the multi-scale reconstruction consistency loss is defined as:

$$L_{rec} = 1 - \frac{1}{N} \sum_{i=1}^{N} \prod_{k \in \mathcal{K}} \left(s_i^k \right)^{\omega_k}, \tag{3}$$

where ω_k denotes the weight of the (k)-hop scale. In addition, a topological positional regularization term was introduced to constrain the learned positional embeddings. Let M denote the normalized Laplacian matrix computed from the face adjacency graph, and let P denote the normalized positional embedding. The positional regularization loss and final pretraining objective are defined as:

$$L_{\text{pos}} = \frac{1}{N^2} \left\| M - PP^\top \right\|_F^2, \quad L = L_{\text{rec}} + \lambda_{\text{pos}} L_{\text{pos}}. \tag{4}$$

where λ_{pos} controls the weight of the positional regularization term.

3.2 LABEL-FREE HARD MINING FINE-TUNING

In the fine-tuning stage, we design a label-free hard example mining strategy that leverages complex samples to improve the efficiency and quality of fine-tuning. Based on hyperedge-level local region representations from the current encoder, the strategy automatically selects candidates with high similarity and high uncertainty for encoder fine-tuning without additional manual annotations.

Given a query structure, we encode and aggregate its target hyperedges to obtain query representations q_i. For candidate parts returned by coarse retrieval, the hyperedge representations are denoted as c_j. This yields the query hyperedge set $Q = \{q_i\}_{i=1}^{M}$, and the candidate hyperedge set

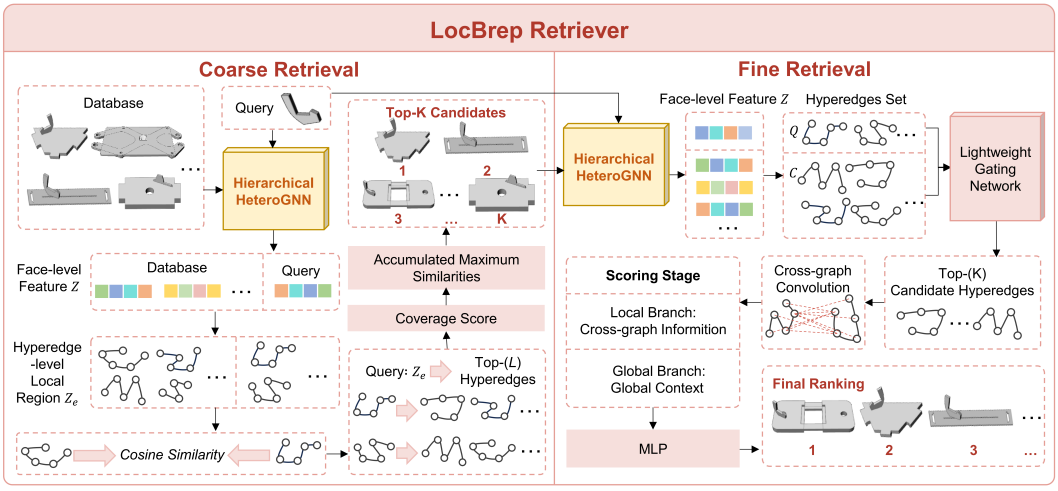


Figure 3: Overview of the LocBrep retrieval framework. The left panel shows hyperedge-level coarse retrieval for efficient candidate recall, and the right panel shows cross-graph matching-based fine retrieval for candidate reranking.

$C = \{c_j\}_{j=1}^{N}$. The affinity A_{ij} between query hyperedge q_i and candidate hyperedge c_j is measured by their cosine similarity.

For candidate hardness estimation, we adapt the affinity-partition-based uncertainty modeling and reservation-based estimation objective used in AUGCL Niu et al. (2024) to query–candidate hyperedge pairs. For each query hyperedge q_i, candidate hyperedges are divided into high- and low-affinity groups using their mean affinity as the threshold, yielding pseudo-labels $y_{ij} \in \{0, 1\}$. The high-affinity group represents candidate local regions that are closer to the query structure, while the low-affinity group represents candidate local regions that are relatively farther from the query structure. We then estimate affinity uncertainty for each query-candidate hyperedge pair using a lightweight MLP. For each query–candidate hyperedge pair, the concatenated representations $[q_i, c_j]$ are fed into a lightweight MLP, which outputs affinity probabilities and reservation uncertainty u_{ij}. The uncertainty objective is defined as:

$$L_{unc} = -\frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \log \left(p_{ij}^{(y_{ij})} + \frac{u_{ij}}{\rho} \right), \tag{5}$$

where $p_{ij}^{(y_{ij})}$ denotes the predicted probability of pseudo-label y_{ij}, u_{ij} denotes the reservation uncertainty, and ρ is the reward coefficient. After estimating uncertainty, we combine it with affinity to prioritize candidate hyperedges that are similar to the query but with high uncertainty, and select the Top-(K) candidates as hard examples. To reduce potential false negatives, pretrained DINOv2 Oquab et al. (2023) features extracted from multi-view CAD renderings are used to filter candidates with high visual similarity to the query.

We then construct a contrastive group using the query representation q and the mined hard negative set $N_q = \{n_j\}$. Using the query self-similarity $\cos(q, q) = 1$ as a fixed reference, we fine-tune the encoder with the following weighted hard-negative separation loss:

$$L = \log \left[1 + \sum_{j=1}^{K} w_j \exp \left(\frac{\cos(q, n_j) - 1}{\tau} \right) \right]. \tag{6}$$

where τ is the temperature coefficient, and w_j denotes the weight of the (j)-th hard negative sample. Details of the affinity uncertainty estimator, relative affinity partitioning, and DINOv2-based false-negative filtering are provided in Appendix B.1, B.2, and B.3, respectively.

4 LOCBREP RETRIEVER

For retrieval, we design a coarse-to-fine retriever structure.An overview of the LocBrep retrieval framework is shown in Fig.3. The retriever includes hyperedge-level coarse retrieval and cross-graph matching-based fine retrieval. The coarse retriever searches candidate parts via global similarity. The fine retriever stage further re-ranks coarse candidates using a graph-to-graph matching module.

4.1 COARSE RETRIEVAL

For coarse retrieval, CAD solid is represented as a set of hyperedge-level local region vectors. We use the pretrained encoder to extract face-level representations for each CAD solid. For a regional hyperedge e containing the face-node set V_e, its local region representation Z_e is defined as:

$$Z_e = \mathrm{Norm}\left(\mathrm{Mean}_{i\in V_e}(Z_i) + \mathrm{Max}_{i\in V_e}(Z_i)\right), \tag{7}$$

where Z_i denotes the representation of the i-th face and $\mathrm{Norm}(\cdot)$ denotes L_2 normalization. The resulting hyperedge representations are indexed.

Both query and database hyperedges are represented using Eq. 7. For each query hyperedge, we retrieve the Top-L indexed database hyperedges with the highest cosine similarities and aggregate the retrieved hyperedges according to their parent parts. Candidate parts are ranked primarily by the number of their retrieved hyperedges. When this number is equal, the accumulated maximum similarities between the query hyperedges and each candidate part are used for tie-breaking. Through the above hyperedge-level indexing and multi-query instance aggregation, the coarse retrieval module can obtain a high-recall candidate part set from the database while maintaining low computational cost. This stage is mainly used to quickly narrow down the retrieval space and provide candidate inputs for the subsequent cross-graph matching-based fine reranking module.

4.2 FINE RETRIEVAL

Although coarse retrieval achieves high recall, independent hyperedge similarity may overlook fine-grained structural correspondences. We therefore introduce a cross-graph matching module to re-rank the coarse candidates. Given a query and a coarse candidate, their face-level representations are aggregated into hyperedge representations using Eq. 7. The target-related query hyperedges and all regional hyperedges of the candidate form sets Q and C, respectively, for cross-graph interaction. To reduce the cost of cross-graph matching, we apply query-aware candidate pruning. The query prototype $\bar{q}$ is obtained by averaging the query hyperedge representations. For each candidate hyperedge c_j, we construct a query-aware candidate feature r_j:

$$r_j = \left[c_j,\, \bar{q},\, |c_j - \bar{q}|,\, c_j \odot \bar{q}\right]. \tag{8}$$

A lightweight gating network takes r_j as input to score candidate hyperedges, retaining the Top-K candidates for cross-graph matching.

In the cross-graph interaction stage, we construct a fully connected bipartite graph between the query hyperedge set Q and the pruned candidate set C' , and apply message-passing-based cross-graph convolution to model their correspondences. On the bipartite graph, the cross-graph convolution computes normalized message weights using the positive cosine similarity between query and candidate hyperedges:

$$\alpha_{ij}^{q\leftarrow c} = \frac{\mathrm{ReLU}\left(\cos(q_i, c_j)\right)}{\sum_{c_{j'}\in C'} \mathrm{ReLU}\left(\cos(q_i, c_{j'})\right)}. \tag{9}$$

$\alpha_{ij}^{q\leftarrow c}$ denotes the normalized message weight from candidate hyperedge c_j to query hyperedge q_i.

The reverse message passing is performed analogously to enable bidirectional interaction between query and candidate hyperedges. We employ two cross-graph convolution layers with a lightweight residual refinement module between them. In the scoring stage, the global branch constructs a part-level global matching feature from the face-level representations of the query and candidate, while the local branch constructs fine-grained local matching features from the outputs of the two cross-graph convolution layers. These features are fused with a statistical matching feature and fed into an MLP to predict the query–candidate relevance score $s_{q,c}$. During inference, the fine retriever computes $s_{q,c}$ for each coarse candidate and re-ranks the candidate parts.

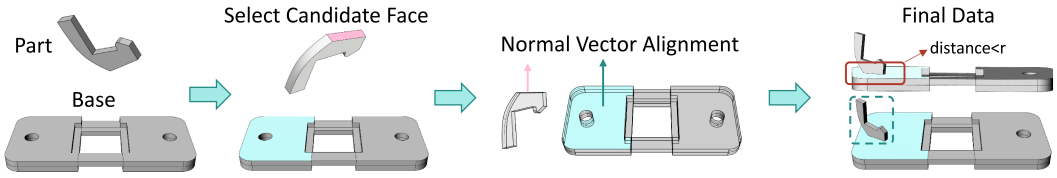


Figure 4: Construction pipeline of LocBrep-Bench. A local component is assembled onto a base model via candidate face selection, normal alignment and geometrically constrained placement.

Table 1: Zero-shot and one-shot retrieval results on FabWave and LocBrep-Bench.

Dataset	Method	Zero-shot			One-shot		
		P@100(%)	R@100(%)	MAP(%)	P@100(%)	R@100(%)	MAP(%)
FabWave	UVStyle-Net	31.83	34.57	40.95	37.25	40.53	55.68
	UV-Net	40.06	43.29	57.17	48.13	52.94	62.08
	BRT	56.90	57.76	66.09	61.95	62.81	68.62
	PP-Brep	55.06	53.41	62.81	57.38	60.30	69.36
	Ours	**64.79**	**68.33**	**74.98**	**69.33**	**75.48**	**79.55**
LocBrep-Bench	UVStyle-Net	19.02	22.40	21.73	23.33	26.73	33.15
	UV-Net	35.98	39.14	41.04	41.15	45.82	50.01
	BRT	44.11	47.42	49.01	61.38	65.51	72.63
	PP-Brep	40.19	43.54	48.77	53.21	59.49	65.11
	Ours	**55.25**	**60.26**	**64.72**	**71.10**	**81.13**	**80.32**

The fine retriever is trained once on an instance-annotated source dataset and directly transferred to downstream retrieval tasks. During training, the pretrained B-rep encoder is frozen, and only the fine retrieval module is optimized using binary cross-entropy loss:

$$L_{\text{rank}} = \text{BCEWithLogits}\left(s_{q,c}, y_{q,c}\right), \tag{10}$$

where $y_{q,c} \in \{0, 1\}$ denotes the matching label of the query-candidate graph pair. In this way, the fine retrieval module learns general local structure matching ability from pretraining data and can be directly used for candidate re-ranking in downstream few-shot local structure retrieval. Detailed feature construction and scoring for global and local branches are provided in Appendix C.

5 Experiment

5.1 Dataset

We build LocBrep-Bench for few-shot B-rep partial retrieval. It includes 50 categories of face-annotated local structures to support query construction, model training and retrieval evaluation. As shown in Fig.4, we first extract local structure components from CAD models with predefined planar contact faces and sample base models from DeepCAD. We then traverse candidate planar faces to find contact faces meeting size and geometric fitting criteria. For composition, each component is aligned to a chosen base face via contact-face normals, randomly positioned under geometric constraints for valid non-penetrating contact. Faces are auto-labeled by source: base-model faces as background, and component faces with their structure category. Additional dataset construction, evaluation protocol and experimental implementation details are provided in Appendix D.

5.2 Experiment Setup

We use the MFInstSeg datasetWu et al. (2024) for representation pretraining and reranker training, where annotations are only leveraged for reranking optimization. The trained reranker is directly transferred to downstream tasks without extra candidate-level labels. We evaluate local structure

retrieval on FabWave Angrish et al. (2019) and our proposed LocBrep-Bench. Since the original FabWave lacks face-level annotations for local structures, we re-label it with face-level instance tags. We adopt P@100, R@100, and MAP as evaluation metrics.

5.3 0-SHOT RETRIEVAL RESULTS

Table 1 presents the zero-shot retrieval results of different methods on FabWave and LocBrep-Bench. Existing CAD representation methods are mostly optimized for global shape retrieval and struggle to capture fine-grained local geometric structures required for partial retrieval tasks. Without any downstream parameter updates, our method achieves the best performance on P@100, R@100, and MAP across both datasets. On FabWave, our method outperforms the strongest baseline by 7.89, 10.57, and 8.89 percentage points on the three metrics, respectively. On LocBrep-Bench, the corresponding improvements are 11.14, 12.84, and 15.71 percentage points. The larger performance gap on LocBrep-Bench indicates that our method gains greater advantages on the dataset with diverse local structure categories. Compared with competitors, our pre-trained hyperedge-level local region embeddings better encode geometric features of partial B-rep components, instead of only learning global shape descriptors. These results demonstrate that our method learns transferable local B-rep representations and effectively retrieves CAD models containing the target local structures. Implementation details of all compared baseline models are provided in Appendix E.

5.4 1-SHOT-FINETUNE RETRIEVAL RESULTS

Table 1 presents the one-shot retrieval results of different methods on FabWave and LocBrep-Bench. Under this setting, our method uses the given query sample to mine hard examples and fine-tune the encoder. Our method achieves the best performance on P@100, R@100, and MAP across both datasets. On FabWave, it outperforms the strongest baselines by 7.38, 12.67, and 10.19 percentage points on the three metrics, respectively.On LocBrep-Bench, the corresponding improvements are 9.72, 15.62, and 7.69 percentage points. Compared with the zero-shot results, our method improves P@100, R@100, and MAP by 4.54, 7.15, and 4.57 percentage points on FabWave, and by 15.85, 20.87, and 15.60 percentage points on LocBrep-Bench, respectively. These results demonstrate that the proposed hard example fine-tuning strategy effectively enhances the model's adaptability to target local structures.Figure 5 visualizes representative one-shot retrieval results on LocBrep-Bench.

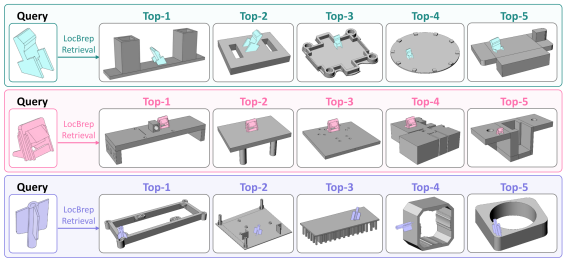


Figure 5: Visualization of one-shot partial retrieval on LocBrep-Bench.

5.5 ABLATION STUDIES

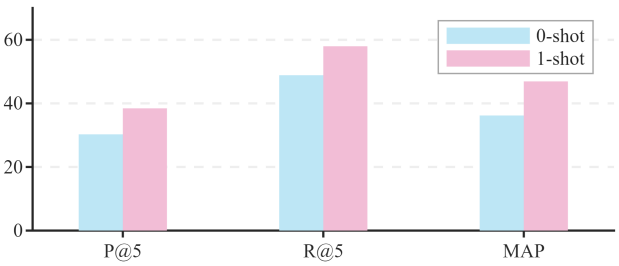
All ablation experiments use models pretrained on MFInstSeg and are evaluated on LocBrep-Bench. Pretraining and retrieval ablations use the zero-shot setting. Hard example mining fine-tuning ablations use the one-shot setting.

Pretraining Strategy. We first compare the proposed BrepDenoise pretraining method with GraphMAEHou et al. (2022). We then construct two ablated variants from the full model: *w/o LF*, which removes the low-frequency smoothing constraint, and *w/o PE*, which removes the topological positional encoding. As shown in Table 2, all alternative and ablated variants perform substantially worse than the full model on P@100, R@100, and MAP, demonstrating the effectiveness of the proposed pretraining strategy and its key components.

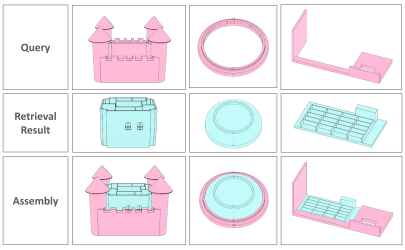
Table 2: Pretraining strategy ablation on LocBrep-Bench.

Method	P@100(%↑)	R@100(%↑)	MAP(%↑)
GraphMAE	29.08	33.05	27.88
w/o LF	28.06	31.92	24.52
w/o PE	39.59	43.79	33.40
Ours	**55.25**	**60.26**	**64.72**

Spectral analysis of the complementary effect of LF and PE is provided in Appendix A.2.



(a) Zero-shot and one-shot partial retrieval results in CAD assemblies on the Fusion 360 Assembly dataset.



(b) Visualization of one-shot partial retrieval in CAD assemblies on Fusion 360 Assembly dataset.

Figure 6: Zero-shot / one-shot partial retrieval visualization on Fusion 360 CAD assembly dataset. Left: retrieval results. Right: one-shot partial retrieval visualization.

Retriever. We ablate the coarse-to-fine retrieval framework, with the results shown in Table 3. All variants share the fixed coarse candidate pool. Coarse uses coarse retrieval only, Coarse+Cos reranks these candidates via cosine similarity, and our full model adopts the cross-graph interaction reranker. All methods use the identical top-100 subset, so P@100 and R@100 stay unchanged. Compared with Coarse and Coarse+Cos, the full model improves MAP by 19.54% and 8.39%. This shows cross-graph matching refines the ordering of relevant candidates inside the top-100 list.

Hard Example Fine-Tuning. We ablate the one-shot fine-tuning strategy in Table 4. Graph Prompt Sun et al. (2023) performs label-guided adaptation, and w/o 2D Filtering removes the multi-view filtering module. Full Fine-tuning underperforms the zero-shot baseline, indicating overfitting under the one-shot setting. Although Graph Prompt improves performance, both it and Full Fine-tuning require downstream labels. In contrast, our method requires no additional candidate-level annotations and achieves the best retrieval metrics.

Table 3: Ablation study of the retriever on LocBrep-Bench.

Method	P@100(%↑)	R@100(%↑)	MAP(%↑)
Coarse	55.25	60.26	45.18
Coarse+Cos	55.25	60.26	56.33
Ours	55.25	60.26	**64.72**

Table 4: Fine-tuning strategy ablation on LocBrep-Bench.

Method	P@100(%↑)	R@100(%↑)	MAP(%↑)
No Fine-tuning	55.25	60.26	64.72
Full Fine-tuning	50.08	53.11	62.83
w/o 2D Filtering	64.89	70.63	71.53
Graph Prompt	65.69	71.58	72.59
Ours	**71.10**	**81.13**	**80.32**

5.6 Partial retrieval in CAD assemblies

To further evaluate the applicability of LocBrep to local-structure-based part retrieval in CAD assemblies, we conduct experiments on the Fusion 360 Gallery Assembly dataset Willis et al. (2021b). Given a target local structure as the query, the model retrieves parts from the candidate library whose local regions geometrically and topologically match the query. We evaluate LocBrep under both zero-shot and one-shot settings. The zero-shot setting directly uses the pretrained model, whereas the one-shot setting adapts the encoder through query-guided hard-example mining and fine-tuning. Figure 6a reports the quantitative results under both settings, while Figure 6b visualizes representative retrieval examples.

6 Conclusion

We presented LocBrep for few-shot B-rep CAD partial retrieval. It learns local structure representations through BrepDenoise pretraining and hard-example fine-tuning, and combines hyperedge-level coarse retrieval with cross-graph reranking. Experiments on FabWave, LocBrep-Bench, and Fusion 360 Gallery Assembly demonstrate its effectiveness under both zero-shot and one-shot settings.Overall, LocBrep provides an effective solution for retrieving target local structures from large CAD repositories under limited annotation settings.

AI USE STATEMENT

In this work, we used generative AI tools to obtain feedback on research methodology and experimental presentation, assist with literature search and reference checking, and improve the readability of the manuscript. All AI-assisted suggestions were manually reviewed and verified by the authors. No experimental data or reported numerical results were generated by generative AI. The authors take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

All experimental settings, model architectures, and evaluation protocols are elaborated in the main paper and supplementary materials. Dataset preprocessing pipelines, training configurations, and ablation setups are comprehensively documented in the appendix to guarantee reproducibility. We will release the anonymous source code, model checkpoints, and processed data splits upon paper acceptance. All quantitative results presented in this work can be faithfully reproduced following our provided implementation and guidelines.

REFERENCES

Atin Angrish, Benjamin Craver, and Binil Starly. Fab-Search: A 3d CAD model-based search engine for sourcing manufacturing services. *Journal of Computing and Information Science in Engineering*, 19(4):041006, 2019. doi: 10.1115/1.4043248.

Jing Bai, Shuming Gao, Weihua Tang, Yusheng Liu, and Song Guo. Design reuse oriented partial retrieval of cad models. *Comput. Aided Des.*, 42:1069–1084, 2010. URL `https://api.semanticscholar.org/CorpusID:13454470`.

Weijuan Cao, Trevor Robinson, Yang Hua, Flavien Boussuge, Andrew R. Colligan, and Wanbin Pan. Graph representation of 3d cad models for machining feature recognition with deep learning. In *Volume 11A: 46th Design Automation Conference (DAC)*, pp. V11AT11A003, 2020. doi: 10.1115/DETC2020-22355.

Andrew R. Colligan, Trevor T. Robinson, Declan C. Nolan, Yang Hua, and Weijuan Cao. Hierarchical cadnet: Learning from b-reps for machining feature recognition. *Computer-Aided Design*, 147:103226, 2022. doi: 10.1016/j.cad.2022.103226.

Yongkang Dai, Xiaoshui Huang, Yunpeng Bai, Hao Guo, Hongping Gan, Ling Yang, and Yilei Shi. Brepformer: Transformer-based b-rep geometric feature recognition. In *Proceedings of the 2025 International Conference on Multimedia Retrieval*, pp. 155–163, 2025. doi: 10.1145/3731715.3733283.

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, 2022. doi: 10.1109/3DV57658.2022.00024.

Hao Guo, Xiaoshui Huang, Hao Jiacheng, Yunpeng Bai, Hongping Gan, and Yilei Shi. Brepgiff: Lightweight generation of complex b-rep with 3d gat diffusion. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26587–26596, 2025. doi: 10.1109/CVPR52734.2025.02476.

Jiacheng Hao, Chunying Liu, Hao Guo, Ruohan Wang, Hongping Gan, and Yilei Shi. PP-Brep: Few-shot B-rep classification with hybrid graph representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2026.

Junhao Hou, Chenqi Luo, Feiwei Qin, Yanli Shao, and Xiaxuan Chen. FuS-GCN: Efficient B-rep based graph convolutional networks for 3D-CAD model classification and retrieval. *Advanced Engineering Informatics*, 56:102008, 2023. doi: 10.1016/j.aei.2023.102008.

Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. GraphMAE: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM*

SIGKDD International Conference on Knowledge Discovery and Data Mining, 2022. doi: 10.48550/arXiv.2205.10803.

Pradeep Kumar Jayaraman, Aditya Sanghi, Joseph G. Lambourne, Karl D. D. 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 (CVPR)*, pp. 11698–11707, 2021. doi: 10.1109/CVPR46437.2021.01153.

Baoning Ji, Jie Zhang, Yuan Li, and Jiazhen Pang. Free-form CAD model retrieval approach for engineering reuse based on local feature segmentation. *Computers & Graphics*, 111:111–121, 2023. doi: 10.1016/j.cag.2023.01.006.

Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9593–9603, 2019. doi: 10.1109/CVPR.2019.00983.

Joseph G. Lambourne, Karl D. D. 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 (CVPR)*, pp. 12768–12777, 2021. doi: 10.1109/CVPR46437.2021.01258.

Jinwon Lee, Changmo Yeo, Sang-Uk Cheon, and Duhwan Mun. Brepgat: Graph neural network to segment machining feature faces in a b-rep model. *Journal of Computational Design and Engineering*, 10(6):2384–2400, 11 2023. doi: 10.1093/jcde/qwad106.

C. Li, H. Pan, A. Bousseau, and N. J. Mitra. Free2cad: Parsing freehand drawings into cad commands. *ACM Transactions on Graphics (TOG)*, 41(4):1–16, 2022.

Yang Li, Eugene Li, Michael Lenover, Stephen Mann, and Sanjeev Bedi. Edge adjacency graph and neural network architecture for machining feature recognition. *The International Journal of Advanced Manufacturing Technology*, 136(2):897–908, 2025. doi: 10.1007/s00170-024-14903-y.

Zhi Li, Xionghui Zhou, and W. Liu. A geometric reasoning approach to hierarchical representation for b-rep model retrieval. *Comput. Aided Des.*, 62:190–202, 2015. URL `https://api.semanticscholar.org/CorpusID:11032525`.

Yunzhong Lou, Xueyang Li, Haotian Chen, and Xiangdong Zhou. Brep-bert: Pre-training boundary representation bert with sub-graph node contrastive learning. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 1657–1666, 2023. doi: 10.1145/3583780.3614795.

V. Naga Malleswari and P. Gourav Pragvamsa. Automatic machining feature recognition from step files. *International Journal of Computer Integrated Manufacturing*, 36(6):863–880, 2023. doi: 10.1080/0951192X.2022.2162590.

Peter Meltzer, Hooman Shayani, Amir Khasahmadi, Pradeep Kumar Jayaraman, Aditya Sanghi, and Joseph Lambourne. Uvstyle-net: Unsupervised few-shot learning of 3d style similarity measure for b-reps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9670–9679, 2021. doi: 10.1109/ICCV48922.2021.00955.

Chaoxi Niu, Guansong Pang, and Ling Chen. Affinity uncertainty-based hard negative mining in graph contrastive learning. *IEEE transactions on neural networks and learning systems*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

Feiwei Qin, Gaoyang Zhan, Meie Fang, C. L. Philip Chen, and Ping Li. VGNet: Multimodal feature extraction and fusion network for 3D CAD model retrieval. *IEEE Transactions on Multimedia*, 27:1432–1447, 2025a. doi: 10.1109/TMM.2024.3521706.

Feiwei Qin, Liangzhe Zhu, Zijian Xu, Meie Fang, and Ping Li. Cadgcl: Unsupervised retrieval of cad models via boundary representations. *The Visual Computer*, 41(9):6493–6505, 2025b. doi: 10.1007/s00371-025-03949-y.

Queen's University Belfast. Mfcad++ dataset. https://pure.qub.ac.uk/en/datasets/mfcad-dataset-dataset-for-paper-hierarchical-cadnet-learning-from/, 2022. Accessed: 2026-04-15.

Ari Seff, Yaniv Ovadia, Wenda Zhou, and Ryan P. Adams. Sketchgraphs: A large-scale dataset for modeling relational geometry in computer-aided design, 2020. URL https://arxiv.org/abs/2007.08506.

Xiangguo Sun, Hong Cheng, Jia Li, Bo Liu, and Jihong Guan. All in One: Multi-Task Prompting for Graph Neural Networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, KDD '23, pp. 2120–2131, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701030.

Songqiao Tao, Zhengdong Huang, Lujie Ma, and Shunsheng Guo. Partial retrieval of CAD models based on local surface region decomposition. *Computer-Aided Design*, 45(11):1239–1252, 2013. doi: 10.1016/j.cad.2013.05.008.

Karl D. D. Willis, Yewen Pu, Jieliang Luo, Hang Chu, Tao Du, Joseph G. Lambourne, Armando Solar-Lezama, and Wojciech Matusik. Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. *ACM Transactions on Graphics*, 40 (4):1–24, 2021a. doi: 10.1145/3450626.3459818.

Karl DD Willis, Pradeep Kumar Jayaraman, Hang Chu, Yunsheng Tian, Yifei Li, Daniele Grandi, Aditya Sanghi, Linh Tran, Joseph G Lambourne, Armando Solar-Lezama, and Wojciech Matusik. Joinable: Learning bottom-up assembly of parametric cad joints. *arXiv preprint arXiv:2111.12772*, 2021b.

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. doi: 10.1016/j.rcim.2023.102661.

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 (ICCV)*, pp. 6752–6762, 2021. doi: 10.1109/ICCV48922.2021.00670.

Mingyuan Xia, Xianwen Zhao, and Xiaofeng Hu. Machining feature and topological relationship recognition based on a multi-task graph neural network. *Advanced Engineering Informatics*, 62: 102721, 2024. doi: 10.1016/j.aei.2024.102721.

Can Yao, Kang Wu, Zuheng Zheng, Siyuan Xing, and Xiao-Ming Fu. Brepmae: Self-supervised masked brep autoencoders for machining feature recognition, 2026.

Shuming Zhang, Zhidong Guan, Hao Jiang, Xiaodong Wang, and Pingan Tan. Brepmfr: Enhancing machining feature recognition in b-rep models through deep learning and domain adaptation. *Computer Aided Geometric Design*, 111:102318, 2024. doi: 10.1016/j.cagd.2024.102318.

Zhibo Zhang, Prakhar Jaiswal, and Rahul Rai. Featurenet: Machining feature recognition based on 3d convolution neural network. *Computer-Aided Design*, 101:12–22, 2018. doi: 10.1016/j.cad.2018.03.006.

Shengdi Zhou, Tianyi Tang, and Bin Zhou. Cadparser: A learning approach of sequence modeling for b-rep cad. In *International Joint Conference on Artificial Intelligence*, 2023. URL https://api.semanticscholar.org/CorpusID:260860271.

Junyou Zhu, Langzhou He, Chao Gao, Dongpeng Hou, Zhen Su, Philip S. Yu, Juergen Kurths, and Frank Hellmann. SDMG: Smoothing your diffusion models for powerful graph representation learning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79815–79835. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/zhu25g.html.

Qiang Zou and Lizhen Zhu. Bringing attention to cad: Boundary representation learning via transformer. *Computer-Aided Design*, 189:103940, 2025. doi: 10.1016/j.cad.2025.103940.

APPENDIX

A BREPDENOISE B-REP PRETRAINING

A.1 DENOISING U-NET

BrepDenoise is designed as a conditional representation-learning framework rather than an unconditional graph generator. Given the normalized clean face features $X_0 \in R^{N\times 256}$, the structural encoder first applies a learned linear projection:

$$H^{(0)} = \sigma\left(X_0 W_{\text{in}} + b_{\text{in}}\right), \qquad H^{(0)} \in R^{N\times 128}. \tag{11}$$

The projected features are subsequently transformed through regional-hypergraph convolution and relation-aware graph message passing:

$$S = f_{\text{str}}(X_0, G) \in R^{N\times 128}. \tag{12}$$

In addition to the structural condition $\boldsymbol{S}$, the time embedding $\boldsymbol{T}_t$ is encoded from the diffusion timestep via a lightweight MLP. The topological positional embedding $\boldsymbol{P}$ is produced by the topological positional encoder, which takes random-walk arrival probabilities on the face adjacency graph as input. Thus, the denoiser does not receive the original 256-dimensional clean face features through a direct skip connection. The condition S is lower-dimensional and neighborhood-mixed: each conditional feature is produced by aggregating information over graph relations and regional hyperedges. The structural encoder and denoising network are jointly optimized. The role of S is to provide a stable structural prior rather than to serve as an independent reconstruction target.

Unlike point-wise MSE reconstruction, our objective compares normalized original and reconstructed features after repeated neighborhood aggregation. Consequently, the model is not required to recover every numerical component of X_0. Instead, it is encouraged to preserve feature directions and neighborhood-level structural consistency across multiple scales.

To examine whether the denoiser bypasses the corrupted input through the clean-feature-derived structural condition, we conduct inference-time pathway diagnostics using the same pretrained checkpoint without retraining. In the condition-only variant, X_t is replaced with zeros while S,T_t , and P are retained. In the variant without structural conditioning, S is replaced with zeros while the remaining inputs are unchanged.

As shown in Table 5, the condition-only variant performs substantially worse than the full model, with MAP decreasing from 64.72% to 9.15%. This indicates that the structural condition S alone is insufficient to produce effective retrieval representations and that the denoiser cannot simply bypass the corrupted input through the clean conditioning path. Removing S also reduces MAP from 64.72% to 33.48%, demonstrating that the structural context provides important complementary information. Overall, the results show that X_t serves as the primary input signal, while S acts as an auxiliary structure-aware prior. Their combination, rather than either pathway alone, is necessary for effective representation learning.

Table 5: Inference-time pathway diagnostics on LocBrep-Bench. All variants use the same pretrained checkpoint without retraining.

Method	P@100(%↑)	R@100(%↑)	MAP(%↑)
Condition Only	11.13	15.98	9.15
w/o Structural Condition	30.21	36.72	33.48
Full Model	**55.25**	**60.26**	**64.72**

A.2 SPECTRAL ANALYSIS OF LF–PE COMPLEMENTARITY

Motivation. Table 2 shows a substantial performance advantage for the joint use of the low-frequency smoothing constraint (LF) and topological positional encoding (PE). Retaining LF without PE yields a MAP of 33.40%, while retaining PE without LF yields 24.52%. Their joint configuration achieves 64.72%, exceeding the stronger single-component variant by 31.32 percentage

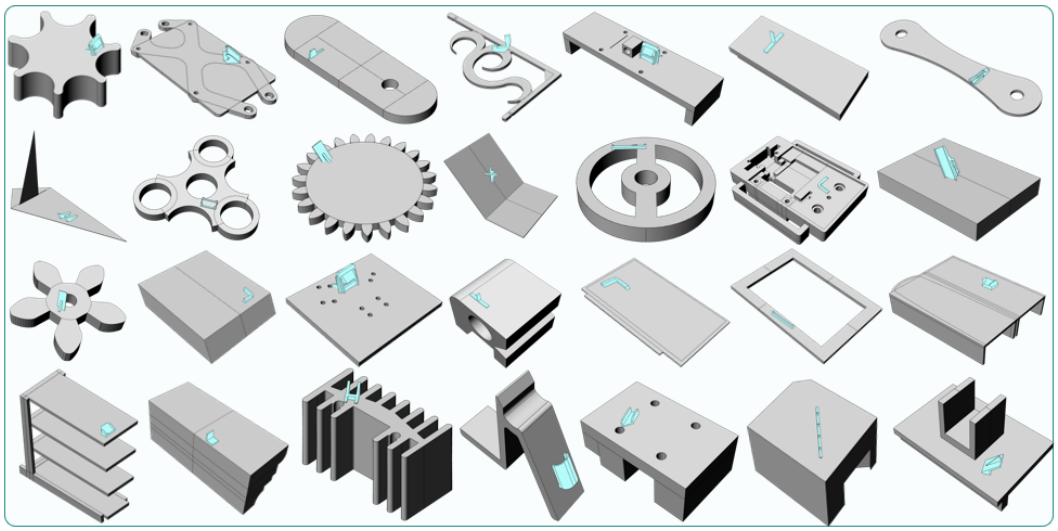


Figure 7: Representative models from LocBrep-Bench. Target local structures are highlighted in cyan, illustrating the diversity of base geometries and local structure categories.

points. To better characterize this joint effect, we examine the spectral energy distributions of the learned representations. This analysis distinguishes the spectral behavior associated with LF from the complementary role of topology-dependent positional conditioning.

Spectral characterization. We analyze the final graph embedding layer of our pre-trained encoder on the test graph set of LocBrep-Bench. For each graph, let $H \in \mathbb{R}^{N\times d}$ denote the analyzed feature matrix, and let $L = U\Lambda U^\top$ denote the symmetric graph Laplacian used in the analysis. The energy associated with the i-th graph frequency is

$$e_i = \left\| u_i^\top H \right\|_2^2,$$

where u_i is the eigenvector corresponding to eigenvalue λ_i. Using the normalized frequency $f_i = \lambda_i/\lambda_{\max}$, the cumulative energy fraction is defined as

$$C(f) = \frac{\sum_{i:f_i\le f} e_i}{\sum_i e_i}.$$

The per-graph energy distributions are combined using average aggregation over all test graphs. Figure 8 presents the cumulative energy distributions and their partition into low-frequency ($f \le 0.2$), mid-frequency ($0.2 < f \le 0.6$), and high-frequency ($f > 0.6$) bands. From our measured cumulative spectral data: the full model obtains **82.22%** cumulative energy within the low-frequency band $f \le 0.2$; w/o-PE (LF-only) obtains **77.34%**; w/o-LF (PE-only) obtains **67.55%**; GraphMAE obtains **75.74%**.

Spectral effect of LF. With PE retained, the full model allocates approximately 82.22% of its spectral energy to the low-frequency band, compared with 67.55% for w/o LF. A clear rise is observed for the corresponding high-frequency fraction. This redistribution is consistent with the intended effect of the LF objective: emphasizing feature consistency after neighborhood aggregation rather than requiring equally detailed point-wise reconstruction. The accompanying retrieval improvement, from 24.52% to 64.72% MAP, shows that the configuration exhibiting this stronger low-frequency emphasis also provides substantially better retrieval performance. We interpret the spectral result as evidence that LF changes the relative frequency emphasis of the learned representations, rather than assuming that all high-frequency components represent noise.

PE contributes beyond spectral smoothing. The comparison with w/o PE reveals a different pattern. The LF-only variant (w/o-PE) already exhibits a considerable low-frequency cumulative energy fraction of 77.34%, which is close to the full model's 82.22%, and the two variants show comparable high-frequency energy levels. Nevertheless, its MAP remains substantially lower, at 33.40% versus 64.72%. Thus, the retrieval advantage of the joint configuration is not adequately characterized by the amount of high-frequency suppression alone.

A complementary-role interpretation follows from the different functions of the two components. LF specifies a neighborhood-consistency objective across aggregation scales, whereas PE provides topology-dependent positional information to condition denoising. Our hypothesis is that LF encourages consistent neighborhood-level representations, while PE helps retain distinctions associated with different topological contexts. Their joint use can therefore favor representations that combine neighborhood consistency with topology-dependent differentiation. Importantly, band-energy fractions summarize the magnitudes of spectral coefficients, but do not specify which structural distinctions those coefficients encode. Similar band-energy fractions consequently need not imply similar retrieval representations.

Relation to alternative pretraining and external baselines. GraphMAE assigns approximately 75.74% of its energy to the low-frequency band but achieves only 27.88% MAP, further illustrating that a relatively high low-frequency fraction alone does not establish effective local-structure retrieval. We also note that GraphMAE and both single-component variants perform below the PP-Brep baseline of 48.77% MAP. These results support the effectiveness of the joint pretraining configuration under the reported evaluation protocol, rather than a claim that either LF or PE independently outperforms alternative complete methods. Moreover, GraphMAE is an alternative pretraining strategy, not the configuration obtained by removing both LF and PE from the same framework, and therefore does not provide a common reference for estimating additive component gains.

Taken together, the retrieval ablations and spectral observations are consistent with complementary roles for LF and PE. LF produces an observable shift in spectral energy allocation, while the substantial additional benefit associated with PE cannot be reduced to further spectral smoothing. The analysis supports this complementary-role interpretation, **and we do not treat band-energy statistics as a causal decomposition of the MAP improvement**.

Table 6: Cumulative spectral energy at $f \leq 0.2$ from measured test-set features.

Model	Cumulative energy @ $f \leq 0.2$
Ours (LF+PE)	0.8222
w/o PE (LF-only)	0.7734
w/o LF (PE-only)	0.6755
GraphMAE	0.7574

B Label-Free Hard Mining Fine-Tuning

B.1 Affinity Uncertainty Estimator: Joint Normalization and Objective Analysis

Joint output parameterization. For each query–candidate hyperedge pair, the uncertainty estimator takes the concatenated representation $[\boldsymbol{q}_i, \boldsymbol{c}_j]$ as input and produces three logits:

$$\mathbf{z}_{ij} = \mathrm{MLP}([\boldsymbol{q}_i, \boldsymbol{c}_j]) = [z^0_{ij}, z^1_{ij}, z^u_{ij}].$$

The two affinity-class probabilities and the reservation probability are obtained through a single three-way Softmax:

$$[p^0_{ij}, p^1_{ij}, u_{ij}] = \mathrm{Softmax}(\mathbf{z}_{ij}).$$

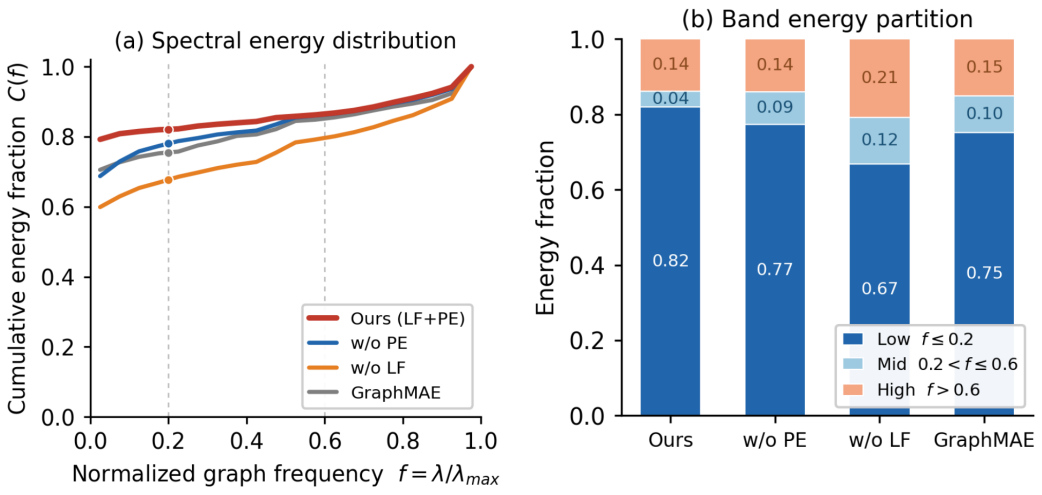
Consequently,

$$p^0_{ij} + p^1_{ij} + u_{ij} = 1, \qquad p^0_{ij}, p^1_{ij}, u_{ij} \geq 0.$$

Here, u_{ij} represents the probability mass allocated to reservation. It is not an independently normalized output. Likewise, p^0_{ij} and p^1_{ij} are taken directly from the joint three-way distribution and are not renormalized over the two affinity classes before computing the loss.

Reward coefficient and objective. The reward coefficient ρ is a fixed hyperparameter satisfying $\rho > 1$, with $\rho = 1.9$. It is not optimized together with the estimator. For a pair with affinity-partition target $y_{ij} \in \{0, 1\}$, the per-pair loss is

$$\ell_{ij} = -\log \eta_{ij}, \qquad \eta_{ij} = p^{(y_{ij})}_{ij} + \frac{u_{ij}}{\rho}.$$



(a) Cumulative spectral energy distribution

(b) Band energy partition

Figure 8: Spectral analysis of LF–PE complementarity. (a) Cumulative spectral energy fraction $C(f)$ versus normalized graph frequency $f = \lambda/\lambda_{\max}$. Curves show how spectral energy accumulates as graph frequency increases. (b) Band energy partition, splitting total energy into low-frequency ($f \leq 0.2$), mid-frequency ($0.2 < f \leq 0.6$), and high-frequency ($f > 0.6$) components. The full model (Ours) concentrates more energy in low-frequency components than w/o LF. The w/o PE variant retains a similar low-frequency energy fraction to the full model, yet achieves much lower retrieval performance, demonstrating that PE contributes topological discrimination beyond spectral smoothing.

The uncertainty objective in Eq. (5) averages this loss over the query–candidate pairs. The partition targets describe relative affinity groups rather than semantic positive or negative retrieval labels.

Range of the logarithm argument. The joint normalization implies $p_{ij}^{(y_{ij})} \leq 1 - u_{ij}$. Therefore, for finite logits and $\rho > 1$,

$$0 < \eta_{ij} \leq 1 - u_{ij} + \frac{u_{ij}}{\rho} = 1 - \left(1 - \frac{1}{\rho}\right) u_{ij} \leq 1.$$

Thus, the logarithm argument cannot exceed one, and $\ell_{ij} \geq 0$. This bound follows from the joint probability constraint and the reward coefficient, rather than from independently restricting each output to the interval $[0, 1]$.

Effect of reservation on optimization. Because all three outputs share the same normalization, increasing reservation is not equivalent to adding an independent positive term while leaving both affinity-class probabilities unchanged. For an individual labeled pair, the full-reservation limit gives

$$u_{ij} \to 1 \quad \Longrightarrow \quad \ell_{ij} \to \log \rho > 0,$$

whereas a confident assignment to the target affinity class gives

$$p_{ij}^{(y_{ij})} \to 1, \quad u_{ij} \to 0 \quad \Longrightarrow \quad \ell_{ij} \to 0.$$

Hence, for an individually identifiable pair, full reservation is inferior to a confident correct assignment.

The gradient with respect to the reservation logit further characterizes this competition:

$$\frac{\partial \ell_{ij}}{\partial z_{ij}^{u}} = u_{ij} \left(1 - \frac{1}{\rho \eta_{ij}}\right).$$

When $\eta_{ij} > 1/\rho$, this derivative is positive, so a gradient-descent step on the reservation logit decreases it. When $\eta_{ij} < 1/\rho$, the derivative is negative, allowing reservation to increase. The

objective therefore does not impose an unconditional incentive to increase reservation; its effect depends on the current allocation of probability mass.

Interpretation and scope. These properties establish the range of the effective target score and the competing optimization incentives under joint normalization. They do not constitute a universal convergence guarantee against high-reservation solutions for a shared neural estimator. In LocBrep, u_{ij} is used as a reservation-based affinity-uncertainty score for candidate mining, rather than as a calibrated probability of semantic irrelevance. It is combined with affinity-based candidate selection and subsequent multi-view filtering before encoder fine-tuning.

B.2 Relative Affinity Partitioning

For each query hyperedge q_i, we compute its mean affinity over all candidate hyperedges as

$$\bar{A}_i = \frac{1}{N} \sum_{j=1}^{N} A_{ij}. \tag{13}$$

We then define the relative affinity-partition target as

$$y_{ij} = \mathbb{I}\left(A_{ij} \geq \bar{A}_i\right), \qquad y_{ij} \in \{0, 1\}. \tag{14}$$

Here, $y_{ij} = 1$ and $y_{ij} = 0$ indicate that the candidate belongs to the relatively high- and low-affinity partitions, respectively. Importantly, these targets do not represent semantic positive or negative relations. In particular, the high-affinity partition is not treated as a pseudo-positive set; it only contains candidates that are relatively closer to the query under the current encoder. The partition targets are used exclusively to train the affinity uncertainty estimator and are not directly used as supervision for encoder fine-tuning.

Candidates with both high affinity and high uncertainty are regarded as ambiguous hard candidates. Before they are used as hard negatives, the DINOv2-based filtering module removes candidates with high visual similarity to the query as potential false negatives.

B.3 DINOv2-Based False-Negative Filtering

We use a pretrained DINOv2 ViT-S/14 model to extract multi-view image features from CAD renderings. Each input image is resized to 224×224, and at most six views are used for each CAD part. Image similarity is computed at the part level between the query part and each mined candidate part. Before a candidate is added to the hard-negative set, it is regarded as a potential false negative and removed if its image similarity is not lower than the threshold $\gamma = 0.75$.

C LocBrep Retriever

C.1 Global and Local Matching Feature Construction

Given the face-level representations Z_q and Z_c of the query and candidate parts, respectively, the global branch applies attention-based readout to obtain the part-level representations g_q and g_c. The global matching feature is then constructed using a pairwise feature formulation:

$$r_{\text{global}} = \left[g_q,\, g_c,\, |g_q - g_c|,\, g_q \odot g_c\right]. \tag{15}$$

The local branch uses the query and candidate representations produced by the two cross-graph convolution layers. For the outputs $\left(H_q^t,\, H_c^t\right)$of the t-th layer, attention-based readout is applied to obtain the local representations l_q^t and l_c^t. The corresponding local matching feature is defined as

$$r_{\text{local}}^t = \left[l_q^t,\, l_c^t,\, \left|l_q^t - l_c^t\right|,\, l_q^t \odot l_c^t\right], \qquad t \in \{1, 2\}. \tag{16}$$

In addition, the second-layer candidate representations H_c^2 are used to predict the response of each candidate hyperedge to the query structure. These responses are aggregated into a statistical feature r_{stat} , which indicates whether the candidate part contains local regions matching the query.

Finally, the global, local, and statistical features are concatenated and fed into an MLP:

$$s_{q,c} = \mathrm{MLP}\left(\left[r_{\mathrm{global}},\, r^1_{\mathrm{local}},\, r^2_{\mathrm{local}},\, r_{\mathrm{stat}}\right]\right). \tag{17}$$

The resulting score $s_{q,c}$ measures the relevance between the query local structure and the candidate part and is used to re-rank the coarse retrieval candidates during inference.

C.2 Decoupled Pretraining and Reranker Training

To verify whether the fine reranker can achieve zero-shot transfer without sharing the same dataset used for pretraining, we conduct BrepDenoise pretraining, reranker training, and downstream evaluation on DeepCAD, MFInstSeg, and LocBrep-Bench, respectively. Specifically, BrepDenoise is pretrained solely on unlabeled DeepCAD data, while the fine reranker is trained once on query–candidate graph pairs constructed using the instance annotations of MFInstSeg. The encoder and fine reranker are then directly transferred to LocBrep-Bench for the downstream local retrieval task. The experimental results are shown in table. 7.

Table 7: Retrieval performance on LocBrep-Bench under decoupled pretraining and reranker training. BrepDenoise is pretrained on unlabeled DeepCAD, while the reranker is trained on instance-annotated MFInstSeg and directly transferred to LocBrep-Bench.

Setting	P@100 (%↑)	R@100 (%↑)	MAP (%↑)
Zero-shot	52.08	59.44	61.93
One-shot	70.81	79.27	77.73

The results demonstrate that the fine reranker can be directly transferred to an unseen downstream dataset without using downstream annotations or additional fine-tuning. They also show that the self-supervised pretraining of BrepDenoise and the supervised training of the fine reranker can be performed on different datasets.

D Experiment and Dataset Detail

All experiments were conducted on a single NVIDIA A100 80GB PCIe GPU. The compute node was equipped with two NVIDIA A100 80GB PCIe GPUs and two Intel Xeon Platinum 8380 CPUs at 2.30 GHz, providing 80 physical cores and 160 logical threads in total. Each experiment was independently repeated five times with fixed random seeds $\{10, 20, 30, 40, 50\}$, and all reported P@100, R@100 and MAP results are averaged over the five runs. Representative models from LocBrep-Bench are shown in Fig.7, with the target local structures highlighted in cyan.

D.1 Dataset Split and Query Setup

LocBrep-Bench contains 4500 B-rep models across 50 local-structure categories. In each experimental run, one sample is randomly drawn per category to serve as the query. Different one-shot exemplars are selected under each random seed across the five repeated runs.

When constructing composite samples by assembling components, we perform Boolean union operations on individual parts to produce a single connected B-rep solid and remove contact faces and stitching seams between components. During preprocessing, we validate the resulting geometry and filter out samples with failed Boolean operations, disconnected geometry or corrupted topology. Consequently, artificial artifacts such as stitching seams and disconnected multi-body geometry are eliminated from our dataset, preventing the model from exploiting these construction-induced artifacts as shortcut features.

D.2 Details for FabWave and Fusion 360 Assembly Evaluation

For the FabWave dataset, we manually re-annotate the local structure labels. The annotation guideline defines the boundary of each local engineering feature based on the functional and topological definition of CAD features. To guarantee annotation quality, we adopt cross-validation: two annotators independently label each sample, and disagreements are reviewed and resolved by a senior CAD researcher. We record inter-annotator agreement to control annotation consistency.

For the Fusion 360 assembly experiments, we adopt the full Fusion 360 Gallery Assembly dataset containing 8,251 assembly models. The ground-truth local structure correspondences are established by tracing feature provenance from the parametric design timeline of each assembly model, which accurately records the geometric regions belonging to each local feature.

It is worth clarifying that the assembly-based partial retrieval experiment is not the core task of this paper and is only intended to demonstrate the generalization capability of our method on multi-body assembly scenarios. Therefore, we only conduct internal ablation experiments to compare zero-shot and one-shot settings of LocBrep, and do not introduce external baseline comparisons in this auxiliary experiment. Comprehensive quantitative comparisons against state-of-the-art B-rep learning methods are fully performed on our core evaluation benchmarks, LocBrep-Bench and FabWave, which focus on the primary single-body B-rep partial retrieval task of this work.

D.3 Relevance Criterion

A database model is judged relevant to a query if and only if it contains annotated face patches of the same local-structure category as the query. A model is treated as a positive sample regardless of the position, pose, or scaling of the target local structure. Models with only global shape similarity but without target local-structure faces are regarded as negative samples.

D.4 Query-Mask Protocol

Face-level masking is applied for each constructed query. Only the faces belonging to the target local structure are retained as query input, while all remaining background faces of the original part are masked out. The model can only access geometric and topological information from the masked local-structure region.

E Baseline Implementation Details

UVStyle-Net, UV-Net, BRT and PP-Brep are originally proposed for whole-part representation or classification. We adopt the following adaptation pipeline to support our local-structure retrieval task.

E.1 Local Query Input Adaptation

For each query, only the masked local face sub-graph is fed into the baseline encoder to generate local query embedding. For database entries, full part graphs are used to compute global part embeddings. Retrieval ranking is obtained via cosine similarity between query embedding and global part embedding.

E.2 Pre-training Source and Supervision

For fair comparison, all baselines are re-pre-trained on the MFInstSeg dataset to avoid bias from different pre-training corpora. PP-Brep receives B-rep classification supervised pre-training on MFInstSeg, UV-Net undergoes self-supervised pre-training on MFInstSeg, UVStyle-Net uses unsupervised contrastive pre-training on MFInstSeg, and BRT is also pre-trained on MFInstSeg. Locally retrained weights are used for all baselines.

E.3 One-shot Adaptation Procedure

The adaptation pipeline described here **only applies to one-shot experiments**. For zero-shot evaluation, all baselines keep weights frozen and directly extract features without parameter updates.

Under one-shot setting, no baseline leverages DINOv2 filtering or transductive hard-example mining over the full candidate corpus. Parameter updates originate only from the single one-shot exemplar per category. Each baseline follows its native few-shot paradigm. PP-Brep adopts its original Graph-Prompt scheme, where the backbone encoder is frozen and only prompt parameters are optimized. Since UVStyle-Net, UV-Net and BRT have no graph-prompt module, supervised

fine-tuning (SFT) is performed and their encoder backbones are fine-tuned with contrastive loss from the one-shot exemplar.

E.4 Classical Handcrafted B-rep Local CAD Retrieval Methods

We cite Tao et al. (2013); Li et al. (2015); Ji et al. (2023) in related work to introduce classical handcrafted B-rep partial CAD retrieval methods. Unfortunately, none of these works release open-source code or executable implementations. Their pipelines are tightly coupled with customized CAD kernel logic, requiring manually tuned surface region decomposition, hand-designed graph construction, and subgraph isomorphism modules. Reproducing them requires reimplementing the full geometry parsing and matching pipeline from scratch, which demands engineering efforts comparable to building an independent system and exceeds the scope of this work.

Moreover, although all methods take B-rep as input, there are fundamental differences in problem assumptions and matching paradigms. Tao Tao et al. (2013) matches decomposed continuous surface patches and supports similar region description codes; Li Li et al. (2015) constructs hierarchically partitioned graphs and adopts the VF2 subgraph isomorphism algorithm for matching; Ji Ji et al. (2023) uses a bag-of-words model to match handcrafted shape descriptors for CAD retrieval. These traditional methods tend to prioritize exact topological alignment between query and target and are mostly suited to retrieve local regions with highly consistent topological structures. In contrast, our learned method tolerates minor topological variations in local B-rep regions (e.g., face merging/splitting, small topological perturbations), and can retrieve functionally and geometrically similar local structures under variant topological representations. The underlying matching criteria and evaluation protocols differ substantially, making a fully fair direct comparison difficult.

We further test our model on multi-body CAD assembly scenarios using the Fusion 360 Assembly Gallery dataset. As shown in Table 8, our approach achieves 36.19% MAP under zero-shot setting and 46.92% MAP under one-shot setting. The consistent performance gain from one-shot adaptation demonstrates that our local B-rep representation can generalize to assembly-level partial retrieval, even when dealing with multi-component CAD models.

Table 8: Partial retrieval results on CAD assemblies from the Fusion 360 Gallery dataset

	P@5	R@5	MAP
1-shot	38.44%	57.96%	46.92%
0-shot	30.27%	48.86%	36.19%