STAR-Edge: Structure-aware Local Spherical Curve Representation for Thin-walled Edge Extraction from Unstructured Point Clouds

Supplementary Material

1. Appendix

This appendix provides details on network architecture for edge detection, the visualization and composition of the thin-walled structure dataset, and the evaluation of PIE-NET with patch-based improvements. We also show more comparison visualization results of STAR-Edge, and we also verify its effectiveness with real scanned point clouds. We then perform ablation studies on normal estimation and edge optimization as well as a running time analysis. Finally, the limitations of the proposed approach are discussed.

1.1. Network architecture for point classification

Fig. 1 shows the used classification network composed of a series of fully connected (FC) layers. The input layer accepts features of dimension B, representing the bandwidth of the spherical harmonics. The output layer produces a categorical label, predicting whether a given point is an edge

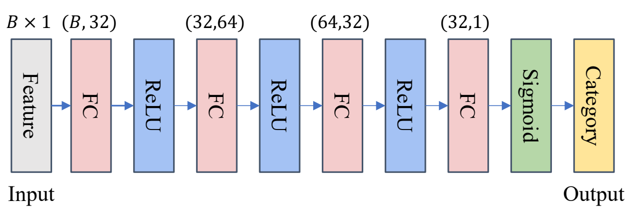

Figure 1. Detailed network architecture for edge point classification.

1.2. Thin-walled structure dataset

Fig. 2 visualizes the used thin-walled structure dataset. These structures exhibit diverse curved edges with varying shapes, sizes, curvatures, and thicknesses ranging from 2 to 4. The dataset comprises 42 unique 3D models, of which 32 are used for the training set and 10 for the test set.

1.3. More careful comparison with PIE-Net

To compare with the SOTA method, PIE-Net, in our early tests, we applied Farthest Point Sampling (FPS) to thin-walled shapes and fed the downsampled point clouds into PIE-Net to extract edges. However, as shown in the main paper, PIE-Net performed poorly on the thin-walled structure dataset. Upon analysis, we determined that the primary issue was the sparsity of the downsampled version of the input data. To address this, we normalized the point clouds, divided each shape into patches, and processed each patch individually through PIE-Net. The extracted edge points from all patches were then aggregated to reconstruct the full edge. The results of both implementations are presented in Tab. 1. While the patch-based method showed slightly better results on the ECD metric, its performance remained sub-optimal compared to other state-of-the-art methods. We attribute this limitation to the presence of nearby surfaces along the edges of thin-walled shapes. The *KNN*-based feature extraction approach employed by PIE-Net struggles to accurately capture and represent this specific characteristic.

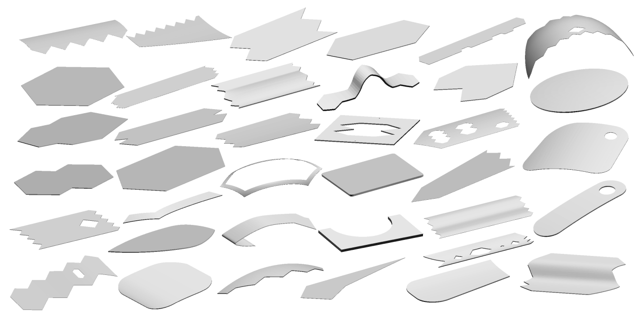

Figure 2. Visualization of thin-walled structure dataset.

Table 1. Comparisons of different PIE-NET implementations.

Metric	PIE-Net (fps) [2]	PIE-Net (patch) [2]
Recall↑	0.05	**0.275**
Precision↑	**0.03**	0.019
F1↑	**0.037**	0.035
Accuracy↑	**0.935**	0.829
ECD↓	52.12	**43.92**

1.4. More visualization results on thin-walled structures and ABC dataset

Fig. 3 compares the performance of STAR-Edge with state-of-the-art methods on the thin-walled structures under a challenging sampling resolution. Specifically, the tested thin-walled shapes have thicknesses ranging from 2 to 3, with a sampling resolution of 0.8. This implies that side-ended faces may be covered by as few as three points.

In these scenarios, EC-Net and MFLE tend to generate redundant points along the ground-truth edges, including a significant number of misclassified points. As analyzed in Sec. 1.3, PIE-Net also performs poorly on the thin-

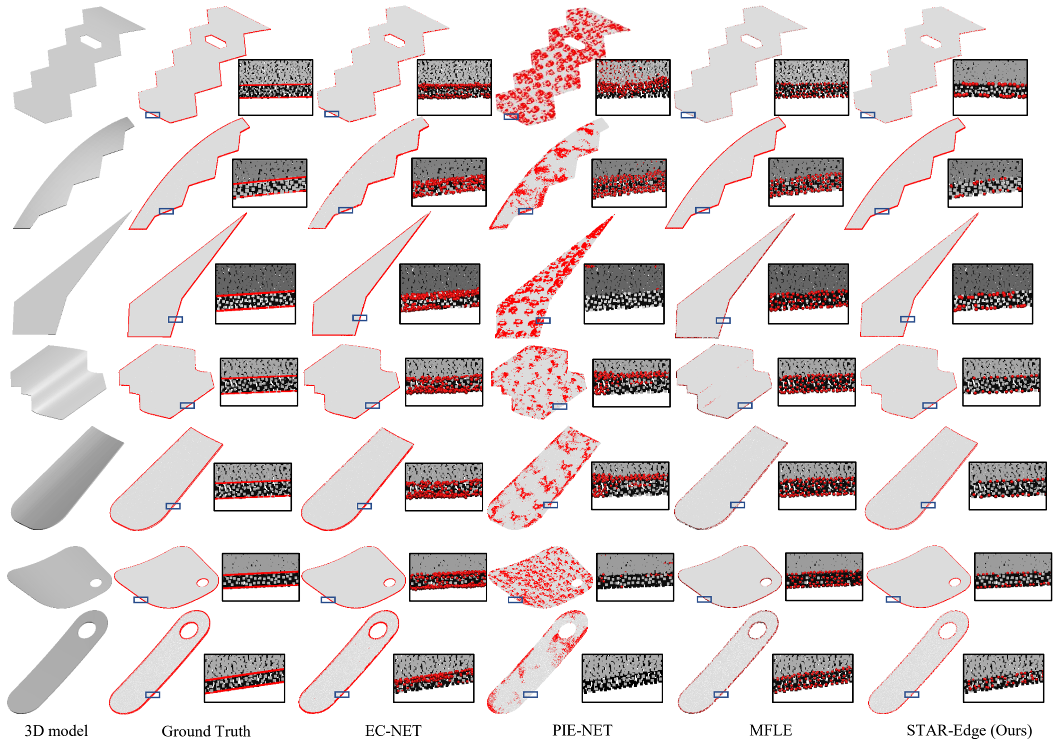

Figure 3. Visual comparison of different methods on the thin-walled structure dataset at a resolution of 0.8. Red points represent edge points. Our method exhibits superior accuracy in extracting edge points.

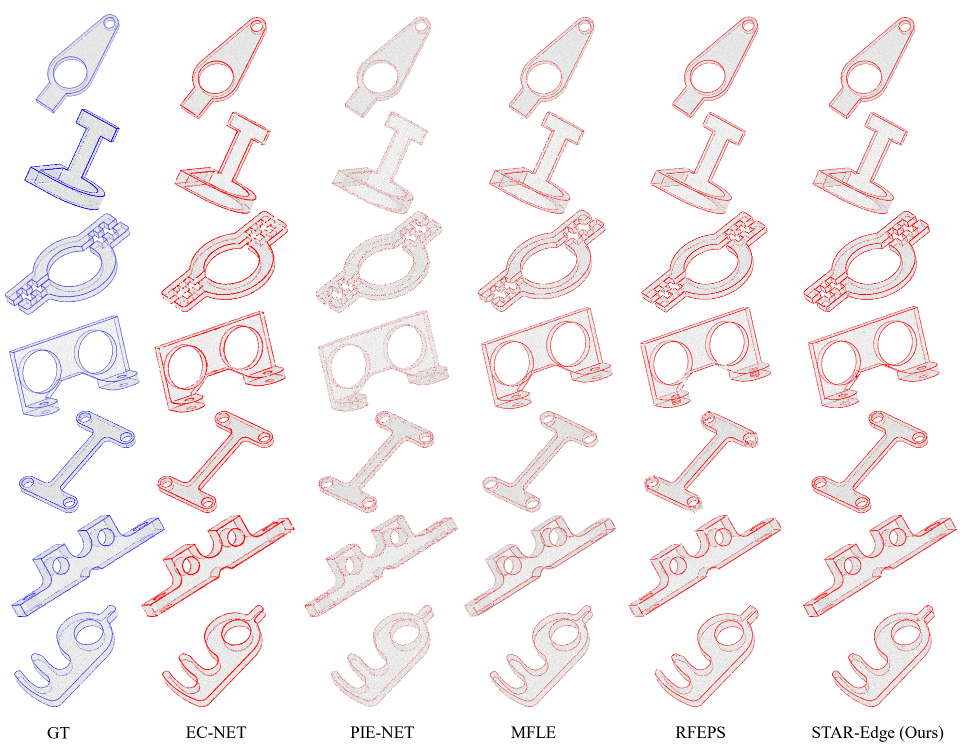

Figure 4. Visual comparison of edge extraction results on 3D shapes from the ABC dataset.

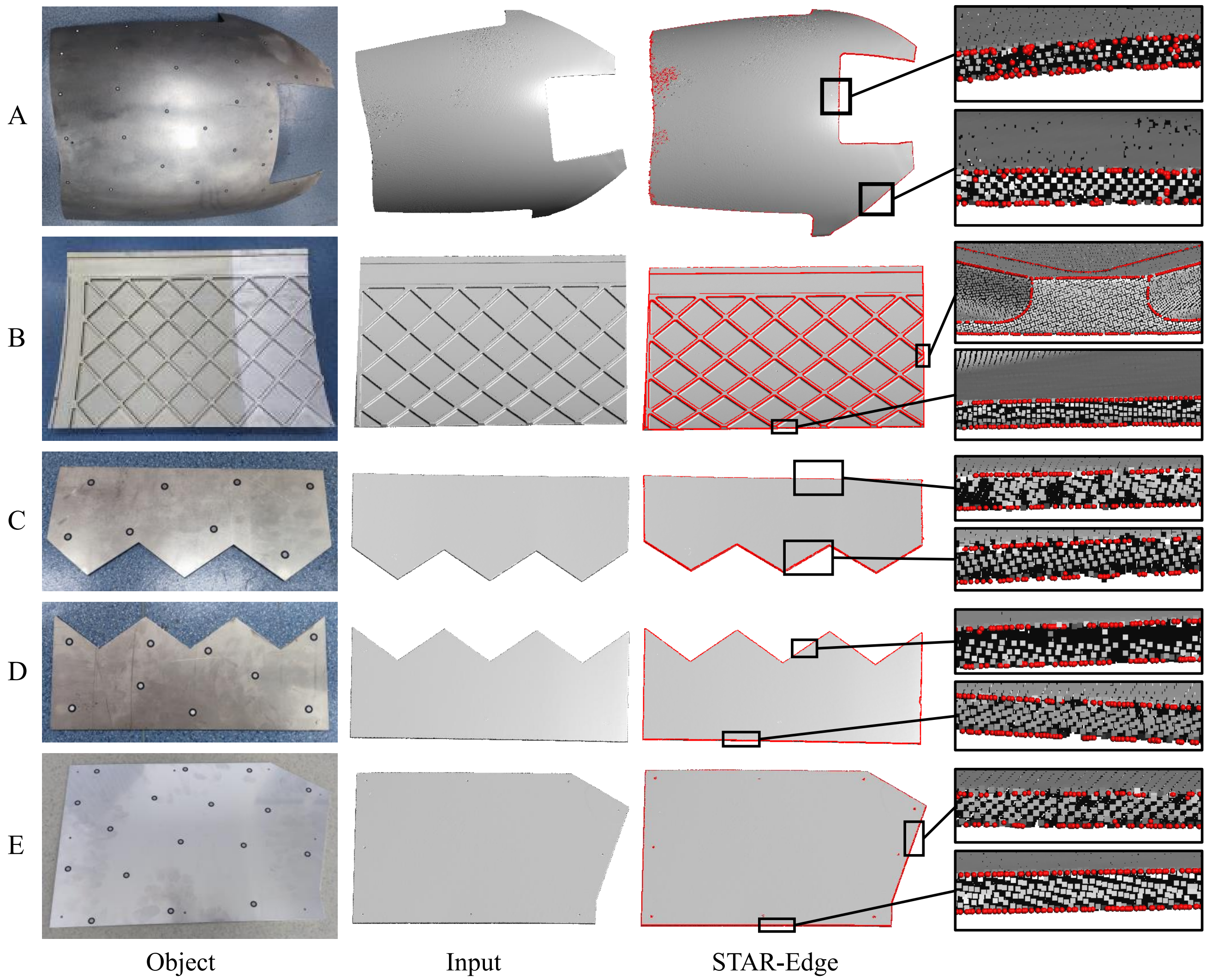

Figure 5. Visualization of our method applied to real-scanned thin-walled structure data. In these close-up views, our method effectively extracts edge points with high accuracy. Edge points are colored in red.

walled structure dataset. In contrast, our method outperforms these baselines by producing more precise and clearer edge points.

We also evaluate our method on the ABC dataset. As shown in Fig. 4, STAR-Edge achieves more precise edge extraction than other baselines. It effectively captures critical features, particularly in regions with sharp corners.

1.5. Evaluation on Real-scanned 3D point clouds

In addition to the synthetic data, we evaluate our method on several real-scanned point clouds of practical thin-walled workpieces. Object A is a 1-meter-long skin panel with a thickness of 2 mm. Object B is a wall panel with a minimum thickness of approximately 2.5mm. Objects C, D, and E are thin-walled structures with a uniform thickness of 3mm. All point cloud data were collected using the SIMSCAN scanner, configured with a resolution of 0.5mm.

Fig. 5 presents the edge extraction results on these scans. Notably, Object A poses a significant challenge due to its extremely thin-walled structure, where the cross-sectional side edges are represented by only about four points. Despite some misidentifications, STAR-Edge effectively identifies edge points and demonstrates robustness against interference from the very close upper and lower surfaces. Moreover, as illustrated in Fig. 6, the proposed method performs well not only on thin-walled shapes but also effectively extracts edges from common real-world objects, further validating its effectiveness.

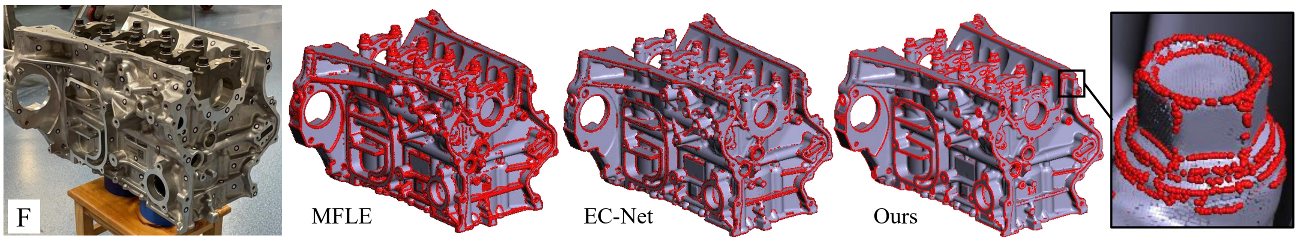

Figure 6. Visualization of a common complex object (engine).

1.6. Effect of different normals for edge point optimization

To evaluate the impact of different normal estimation methods on the final edge point results, we compare two variants of STAR-Edge, as detailed in Tab. 2. *Variant A*: The edge point optimization module is removed, relying solely on the classification results. *Variant B*: The edge point optimization uses the commonly employed PCA method for local normal estimation. *Ours*: The proposed normal estimation method is applied to optimize the edge points. Both *Variant A* and *Variant B* exhibit noticeable performance degradation, highlighting the effectiveness of our normal estimation method in refining edge points.

Table 2. Effect of different normals for edge point optimization.

Method Variants	Normal Estimation Method	ECD↓
A: w/o optimization	–	0.2921
B: w/ optimization	PCA normal	0.1134
Ours: w/ optimization	ours normal	**0.0587**

1.7. Running time performance

Table 3. Running time for different number of points.

#Points	92,000	327,000	2,090,000
EC-Net [4]	25s	1min 23s	11min 8s
PIE-Net (fps) [2]	0.4s	0.4s	0.4s
PIE-Net (patch) [2]	1.6s	5.7s	35.3s
MFLE [1]	0.7s	1.1s	7.6s
RFEPS [3]	6min 56s	19min	322min
STAR-Edge (Ours)	59.6s	3min 30s	22min 24s

We report the running time statistics in Tab. 3, which include point clouds with different numbers of points. As reported, the running efficiency of our method is moderate.

1.8. Limitations

Our method demonstrates robust edge detection performance for thin-walled structure data. However, it relies somewhat on the distribution of local neighborhood spherical projections, which can lead to misidentifications in the presence of sharp noise. Additionally, insufficient points on the side edges may hinder accurate detection. Due to the need for per-point neighborhood calculations and iterative optimization, our approach exhibits lower efficiency compared to other methods.

References

[1] Honghua Chen, Yaoran Huang, Qian Xie, Yuanpeng Liu, Yuan Zhang, Mingqiang Wei, and Jun Wang. Multiscale feature line extraction from raw point clouds based on local surface variation and anisotropic contraction. *IEEE Transactions on Automation Science and Engineering*, 19(2):1003–1016, 2021. 4

[2] Xiaogang Wang, Yuelang Xu, Kai Xu, Andrea Tagliasacchi, Bin Zhou, Ali Mahdavi-Amiri, and Hao Zhang. Pie-net: Parametric inference of point cloud edges. *Advances in neural information processing systems*, 33:20167–20178, 2020. 1, 4

[3] Rui Xu, Zixiong Wang, Zhiyang Dou, Chen Zong, Shiqing Xin, Mingyan Jiang, Tao Ju, and Changhe Tu. Rfeps: Reconstructing feature-line equipped polygonal surface. *ACM Transactions on Graphics (TOG)*, 41(6):1–15, 2022. 4

[4] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Ec-net: an edge-aware point set consolidation network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 386–402, 2018. 4