ITC 3/55
Information Technology and Control
Vol. 55 / No. 3/ 2026
pp. 1530-1544
DOI 10.5755/j01.itc.55.3.45562

CovaEdge-Net: Geometry-Conditioned Adaptive Graph Learning for Industrial Point Cloud Edge Detection

Received 2026/07/17 | Accepted after revision 2026/09/21

HOW TO CITE: Yu, H., Ren, X. (2026) CovaEdge-Net: Geometry-Conditioned Adaptive Graph Learning for Industrial Point Cloud Edge Detection. *Information Technology and Control*, 55(3), 1529-xxxx. https://doi.org/10.5755/j01.itc.55.3.45562

CovaEdge-Net: Geometry-Conditioned Adaptive Graph Learning for Industrial Point Cloud Edge Detection

Haifeng Yu

School of Mechatronics Engineering, Shenyang Aerospace University, Shenyang 110136, China; Key Laboratory of Rapid Development & Manufacturing Technology for Aircraft (Shenyang Aerospace University), Ministry of Education, Shenyang Aerospace University, Shenyang 110136, China; e-mail: yuhaifeng124@stu.sau.edu.cn

Xu Ren*

Office of Smart Campus Construction (Information Network Center), Shenyang Aerospace University, Shenyang 110136, China; e-mail: renxuxx99@outlook.com

Corresponding author: renxuxx99@outlook.com

Reliable edge detection in industrial point clouds is difficult when Euclidean neighborhoods cross surface discontinuities and local descriptors vary with object pose. CovaEdge-Net addresses these effects through geometry-conditioned neighborhood construction, rotation-aware pairwise encoding, and temperature-controlled aggregation. Adaptive Covariance Neighbor Selection (ACoNS) ranks candidate neighbors with a covariance-weighted dissimilarity that combines spatial, normal, and curvature terms. The Tangent Orientation Descriptor (TOD) separates scalar rotation-invariant relations from learned feature differences, while Adaptive Temperature-controlled Aggregation (ATA) uses a positive, lower-bounded temperature to control neighborhood selectivity. On the ABC benchmark, CovaEdge-Net reports an F1 score of 0.973 and is evaluated under density, rotation, noise, ablation, and one model-level efficiency setting. The evaluation covers one dataset and reports aggregate results across three runs; cross-dataset generalization, formal convergence, and adversarial robustness remain outside the evaluated scope. The ablation results are used to interpret the roles of ACoNS, TOD, and ATA without extending the conclusions to unmeasured intermediate quantities.

KEYWORDS: Point Cloud, Industrial Edge Detection, Covariance Geometry, Rotation Invariance, Adaptive Graph, Temperature-Controlled Aggregation

1. Introduction

Industrial point clouds support dimensional inspection, reverse engineering, robotic handling, and assembly planning [1, 3, 8, 16, 30]. In these pipelines, sharp edges, corners, holes, and surface seams are not merely visual details: they determine tolerance measurements, defect localization, and stable pose estimates.

Classical detectors use normal variation, local covariance, or surface fitting [7, 31]. Their performance can deteriorate when sampling density changes, normals are noisy, or a neighborhood straddles two surfaces. Deep point-set models improve representation learning, but the quality of their local graph remains a limiting factor.

PointNet and PointNet++ obtain permutation invariance through shared mappings and symmetric aggregation [9, 10]. DGCNN rebuilds feature-space neighborhoods, while KPConv uses deformable kernels [18, 21]. Transformer and deep local-aggregation architectures enlarge the receptive field [11, 28, 29], often with additional memory and computation.

Three coupled gaps remain for industrial edge detection. First, Euclidean KNN can connect points from opposite sides of a discontinuity even when they are spatially close. Second, descriptors that contain raw coordinate offsets are equivariant rather than invariant and therefore cannot by themselves justify rotation-invariant predictions. Third, fixed max or mean aggregation provides no explicit control over how selectively a point accepts messages from its neighbors.

Prior edge-aware networks sharpen reconstructed geometry or introduce boundary losses [15, 23, 27, 32], whereas rotation-oriented representations typically address descriptor design [17, 25]. Registration models encode structured geometric relations [13, 14]. These advances do not, by themselves, establish a mechanism linking neighborhood purity, invariant local evidence, and adaptive message selectivity.

This paper addresses that gap with CovaEdge-Net. Rather than treating its modules as independent engineering additions, we define an explicit mechanism hypothesis: ACoNS should reduce cross-boundary neighbor selection; TOD should reduce rotation-induced descriptor error; and ATA should adapt attention entropy to local geometric ambiguity. The evaluation is organized around prediction quality and the intermediate mechanisms for which measurements are available.

ACoNS uses a symmetric, regularized covariance-weighted dissimilarity score. We deliberately avoid calling this score a metric because no triangle-inequality proof is required for KNN ranking. TOD is reformulated with scalar invariants under SO(3), and ATA separates a geometry penalty from a genuine positive temperature.

The main contributions are:

1. ACoNS, a symmetric geometry-conditioned neighbor-ranking rule based on regularized local covariances, normal agreement, and curvature contrast.
2. TOD, a strictly rotation-invariant scalar descriptor with an explicit SO(3) invariance argument and implementation pseudocode.
3. ATA, a temperature-controlled attention mechanism that keeps the temperature positive and separates selectivity from geometry regularization.
4. A mechanism-oriented analysis that relates the module ablations to neighborhood selection, rotation-aware encoding, and adaptive aggregation while distinguishing measured outcomes from unmeasured intermediate quantities.

The numerical evaluation uses the ABC benchmark. Because only aggregate results from three runs are reported and no second dataset is evaluated, the analysis does not estimate run-to-run uncertainty or claim cross-dataset generalization; unmeasured intermediate mechanisms are treated as hypotheses rather than empirical results.

2. Related Work

2.1. Point-Cloud Representation and Neighborhood Construction

PointNet introduced direct learning on unordered point sets, and PointNet++ added hierarchical local regions [9, 10]. DGCNN dynamically constructs feature-space KNN graphs, whereas KPConv learns flexible kernels on irregular samples [18, 21]. Point

transformers and deeper local aggregation networks extend interaction range [11, 28, 29]. These backbones improve representation capacity, but their neighborhoods are not designed specifically to prevent message passing across industrial surface discontinuities.

2.2. Edge-Aware and Geometry-Conditioned Learning

EC-Net and sharpness-field methods introduce edge-aware reconstruction objectives [15, 27]. Anisotropic operators and improved normal estimation strengthen local geometry [23, 32]. BADet, EdgeFormer, EdgeMovingNet, SED-Net, and PIE-NET further exploit boundary-aware detection, reconstruction edges, primitive structure, or parametric edge curves [12, 6, 20, 24, 26]. For industrial welding, Wang et al. proposed an adaptive point-cloud framework for robust weld-seam extraction from complex free-form workpieces without relying on CAD models [19]. Their objectives differ from the present task: CovaEdge-Net modifies the graph before message passing and evaluates the resulting point-wise edge predictions on industrial CAD geometry.

Graph and attention models learn relational structure at different levels [14, 13, 2, 5]. Their objectives concern registration or general set/graph representation rather than point-wise industrial edge detection. The present method instead uses a deterministic covariance-conditioned ranking stage before learned message passing.

The resulting distinction is precise: ACoNS is a deterministic ranking rule conditioned on local covariance, normals, and curvature; it is not claimed to satisfy every axiom of a mathematical distance metric.

2.3. Rotation Invariance and Application Constraints

Rotation-invariant point learning includes sorted Gram-matrix representations and geometric-algebra attention [17, 25]. A vector such as $p_j - p_i$ rotates with the object and is therefore equivariant; an in-

Table 1

A comparison of the neighborhood design, edge or rotation treatment, core advantages, and limitations of representative point-cloud methods. The comparison is qualitative and complements, rather than replaces, the quantitative benchmark.

Method	Neighborhood / representation	Edge or rotation treatment	Advantages	Limitations
PointNet++ [10]	Euclidean local regions	Symmetric pooling; no explicit edge mechanism	Simple hierarchical local aggregation; efficient and widely used.	Euclidean grouping may cross discontinuities; no explicit edge or rotation mechanism.
DGCNN [21]	Dynamic feature-space KNN	EdgeConv; no strict rotation invariance	Dynamic neighborhoods provide strong local relational modeling.	Graph rebuilding adds cost; neighborhoods may mix points across boundaries.
EC-Net [27]	Input/local neighborhoods	Edge-aware consolidation objective	Explicitly incorporates edge awareness into point-set consolidation.	Designed for consolidation rather than pointwise industrial edge classification.
EdgeMoving-Net [26]	Joint geometric features	Edge-preserving reconstruction	Jointly models geometric reconstruction and edge preservation.	Reconstruction-oriented performance depends on local sampling and learned point movement.
SED-Net [6]	Primitive-oriented features	Surface-and-edge prediction	Joint surface-and-edge prediction exploits primitive-aware structure.	Primitive assumptions may be restrictive for irregular or non-primitive geometry.
CovaEdge-Net	Covariance-conditioned ranking	Invariant TOD + positive-temperature ATA	Geometry-conditioned ranking, scalar SO(3)-invariant encoding, and adaptive aggregation.	Requires normals, curvature, and covariance solves; current evidence is limited to ABC.

variant descriptor must instead be built from quantities preserved by $R \in SO(3)$, including norms, dot products, and cross-product magnitudes. TOD follows this descriptor-level definition while making no strict invariance claim for the learned edge feature or complete network.

3. Methodology

3.1. Overview and Notation

Let $P = \{p_i \in R^3\}_{i=1}^N$ be an input point cloud. Each of four CovaEdge blocks performs: (i) ACoNS neighbor ranking, (ii) TOD pairwise encoding, and (iii) ATA message aggregation. A shared classification head maps the final point features to edge probabilities. Figure 1 gives the architecture, while Algorithms 1–3 specify the revised module interfaces.

3.2. Input Features and Local Geometry

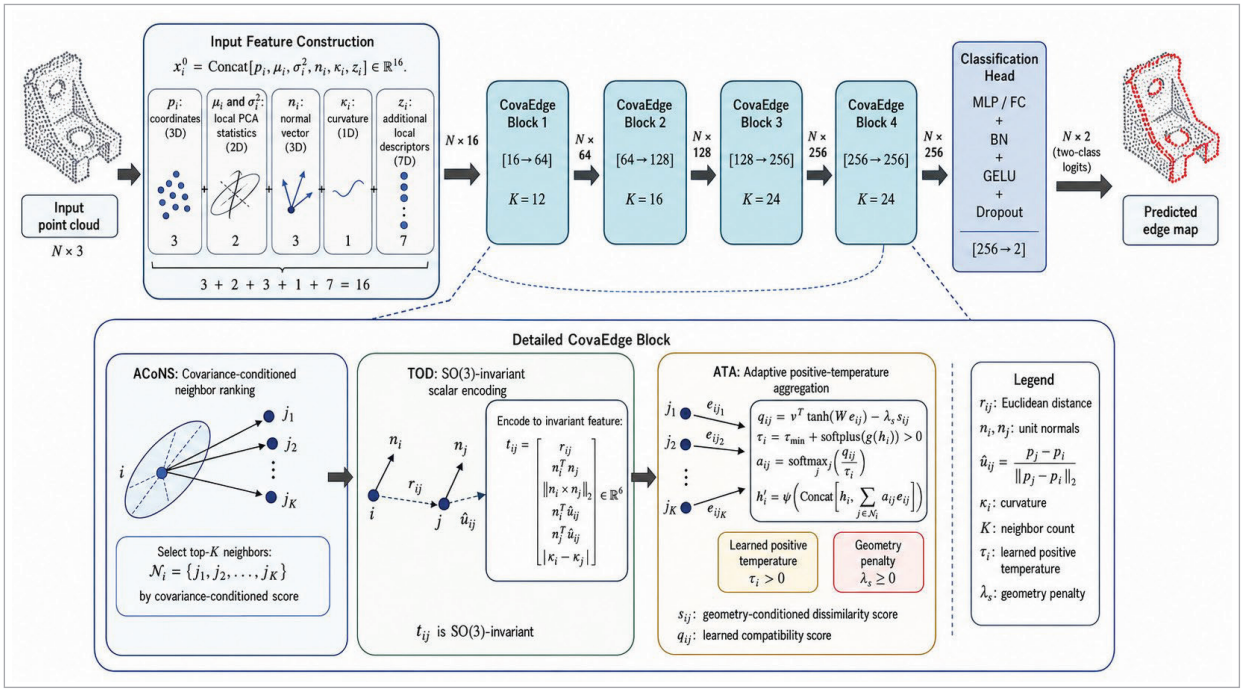
For point i, an initial K_0-neighbor Euclidean patch is used only to estimate local geometry. Its 16-dimensional input feature concatenates the coordinates p_i, the mean μ_i and variance σ_i^2 of neighbor distances, the oriented unit normal n_i, normalized curvature κ_i, and a seven-dimensional handcrafted descriptor z_i:

$$x_i^0 = \mathrm{Concat}[p_i, \mu_i, \sigma_i^2, n_i, \kappa_i, z_i] \in R^{16}. \quad (1)$$

Let $\lambda_1 \le \lambda_2 \le \lambda_3$ be the eigenvalues of the local covariance, let $\tilde{\lambda}m = \lambda m/(\lambda_1+\lambda_2+\lambda_3+\delta)$, and let rK_0 be the distance to the farthest point in the K_0-neighbor patch. We define $z_i = [\rho_i, L_i, P_i, S_i, A_i, O_i, E_i]$, where $\rho_i = 3K_0/(4\pi r K_0^3)$ is local point density, $L_i = (\lambda_3-\lambda_2)/(\lambda_3+\delta)$ is linearity, $P_i = (\lambda_2-\lambda_1)/(\lambda_3+\delta)$ is planarity, $S_i = \lambda_1/(\lambda_3+\delta)$ is scattering, $A_i = (\lambda_3-\lambda_1)/(\lambda_3+\delta)$ is anisotropy, $O_i = (\tilde{\lambda}_1\tilde{\lambda}_2\tilde{\lambda}_3)^{1/3}$ is omnivariance, and $E_i = -\Sigma m = 1^3\tilde{\lambda}m \log(\tilde{\lambda}m+\delta)$ is eigenentropy [22]. The normal n_i is the eigenvector of λ_1, and $\kappa_i = \lambda_1/(\lambda_1+\lambda_2+\lambda_3)$. Normals are consistently oriented within each object before ACoNS and TOD are evaluated; both modules therefore assume a consistent normal-sign convention, and sign-sensitive descriptor terms may change if

Figure 1

Overall CovaEdge-Net architecture. The input point features comprise coordinates, local distance statistics, normals, curvature, and handcrafted local descriptors. Each CovaEdge block applies covariance-conditioned neighbor ranking (ACoNS), SO(3)-invariant scalar pair encoding (TOD), and adaptive positive-temperature aggregation (ATA), followed by a two-class classification head.



an individual normal is flipped. The seven channels in z_i are fixed handcrafted inputs rather than hidden learned features.

3.3. Adaptive Covariance Neighbor Selection (ACoNS)

ACoNS ranks candidate points by a symmetric geometry-conditioned dissimilarity. Each unregularized local covariance $\hat{\Sigma}_i$ is computed once from the K_0-point patch and then regularized exactly once as $\Sigma_i=\hat{\Sigma}_i+\varepsilon I_3$, with $\varepsilon>0$.

$$\Sigma_i = K_0^{-1}\sum_{k\in E_i}(p_k-\bar{p}_i)(p_k-\bar{p}_i)^T+\varepsilon I_3, \quad \varepsilon>0\,. \tag{2}$$

With $\Delta p_{ij}=p_j-p_i$, $\bar{\Sigma}_{ij}=\tfrac{1}{2}(\Sigma_i+\Sigma_j)$, $\alpha\geq 0$, and $\beta\geq 0$, the ranking score is

$$s_{ij}=\Delta p_{ij}^T\bar{\Sigma}_{ij}^{-1}\Delta p_{ij}+\alpha(1-n_i^Tn_j)+\beta|\kappa_i-\kappa_j| \tag{3}$$

$$N_i = \arg\,\mathrm{TopK}\text{-}\min_{j\neq i} s_{ij}\,. \tag{4}$$

The spatial term is nonnegative because the regularized average covariance is positive definite. All three terms are symmetric in i and j. We do not claim the triangle inequality; KNN requires only a reproducible ordering of candidate neighbors. α and β control the relative contributions of normal and curvature disagreement.

Algorithm 1 — Adaptive Covariance Neighbor Selection (ACoNS)

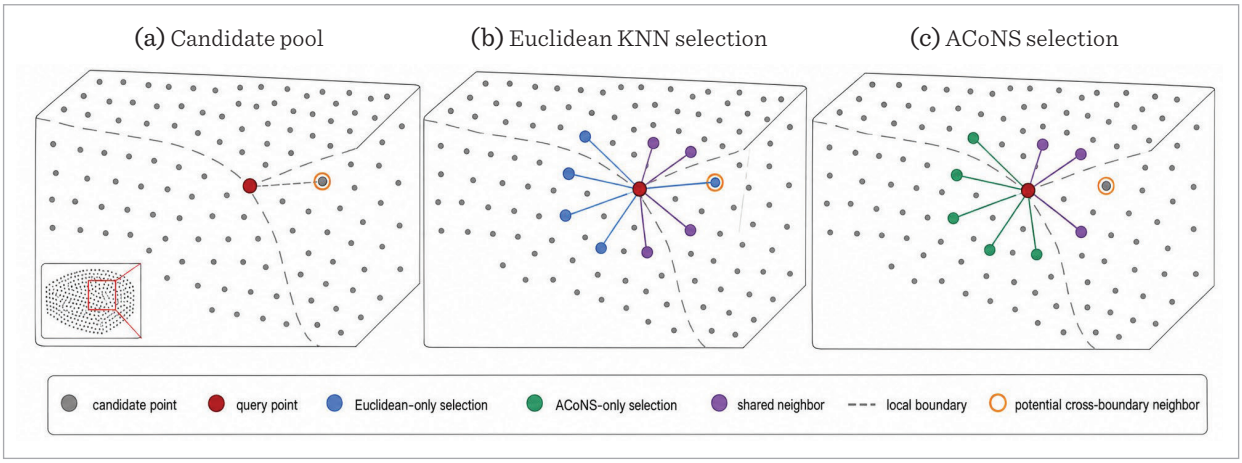
Input: $P=\{p_i\}_{i=1}^N$, K_0, K, Kc, α, β, and $\varepsilon>0$.
Output: directed neighborhoods $\{N_i\}$ and pair scores $\{s_{ij}\}$.

For every i, use one Euclidean search to build a Kc-point candidate pool $C_i\subset\{1,\ldots,N\}\setminus\{i\}$ and take its first K_0 points as E_i, with $Kc\geq\max(K,K_0)$.
For every i, precompute $\bar{p}_i$ and the unregularized covariance $\hat{\Sigma}_i$ from E_i; obtain n_i and κ_i from its eigendecomposition.
Consistently orient $\{n_i\}$ within each object, and set $\Sigma_i\leftarrow\hat{\Sigma}_i+\varepsilon I_3$ once.
for each $i=1,\ldots,N$ **do**
  for each $j\in C_i$ **do**
    $\Delta p_{ij}\leftarrow p_j-p_i$; $\bar{\Sigma}_{ij}\leftarrow\tfrac{1}{2}(\Sigma_i+\Sigma_j)$.
    Solve $\bar{\Sigma}_{ij}u_{ij}=\Delta p_{ij}$ without forming an explicit inverse.
    $s_{ij}\leftarrow\Delta p_{ij}^Tu_{ij}+\alpha(1-n_i^Tn_j)+\beta|\kappa_i-\kappa_j|$.
  end for
  Stably sort C_i by (s_{ij},j) and set N_i to its first K indices.
end for
return $\{N_i\},\{s_{ij}\}$.

The intended effect of ACoNS is to reduce cross-boundary message passing while retaining local surface support. Table 3 shows that replacing ACoNS with Euclidean KNN reduces F1 from 0.973 to 0.945, and Figures 2 and 6 provide qualitative examples near discontinuities. These observations

Figure 2

Mechanism illustration of geometry-conditioned neighborhood selection. (a) Candidate pool around the query point. (b) Euclidean KNN selection, including method-specific and shared neighbors. (c) ACoNS selection after covariance-conditioned ranking. The dashed curve denotes the local boundary and the orange circle marks a potential cross-boundary candidate. The illustration is qualitative and does not report a measured neighborhood-purity rate.



support the contribution of geometry-conditioned ranking but do not constitute a direct measurement of cross-boundary neighbor purity.

Computational complexity. Let Csearch(N,Kc) denote the cost of the Euclidean candidate search. Because $Kc \geq K_0$, the same search supplies both the Kc candidate pool and the nested K_0 geometry patch. Local-geometry precomputation therefore costs $O(Csearch(N,Kc)+NK_0)$, with O(N) constant-size 3×3 eigendecompositions. Pair scoring requires NKc constant-size linear solves and O(NKc) arithmetic, while stable candidate ranking costs O(NKc log Kc). TOD encoding and ATA aggregation over the selected edges cost O(NK). Therefore, the per-block cost is $O(Csearch(N,Kc)+NK_0+NKc \log Kc+NK)$, rather than O(NK); Csearch depends on the neighbor-search backend.

3.4. Tangent Orientation Descriptor (TOD)

TOD forms an explicit geometric descriptor from scalar quantities that are invariant to a common global rotation. Let $r_{ij}=\|p_j-p_i\|_2$ and $\hat{u}_{ij}=(p_j-p_i)/\max(r_{ij},\delta)$, where $\delta>0$ prevents division by zero. This descriptor assumes that surface normals have first been oriented consistently within each object.

$$t_{ij}=[r_{ij}, n_i^T n_j, \|n_i \times n_j\|_2, n_i^T \hat{u}_{ij}, n_j^T \hat{u}_{ij}, |\kappa_i-\kappa_j|]. \quad (5)$$

Equation (5) contains no raw vector offset or cross-product vector. Accordingly, the stated SO(3) result applies only to the scalar geometric descriptor t_{ij}. It does not establish strict rotation invariance of the learned edge feature e_{ij} or of the complete network, because h_i and h_j may contain learned features derived from raw coordinates.

Proposition 1 (descriptor invariance under a common SO(3) rotation). For any R∈SO(3), replacing p by Rp and consistently oriented n by Rn leaves t_{ij} unchanged.

$$\|(Rn_i)\times(Rn_j)\|_2 = \|R(n_i \times n_j)\|_2 = \|n_i \times n_j\|_2. \quad (6)$$

The remaining directional dot products are also preserved because $R^TR=I$ and $R\hat{u}_{ij}$ is the rotated unit offset. Curvature is derived from covariance eigenvalues, which are unchanged by orthogonal similarity transforms. Thus, each component of (5) is invariant.

$$e_{ij}=\phi \mathrm{TOD}(\mathrm{Concat}[h_i, h_j-h_i, t_{ij}]). \quad (7)$$

Here h_i and h_j are incoming point features and ϕTOD is a shared multilayer perceptron. The learned edge feature e_{ij} combines the invariant geometric descriptor with learned semantic differences; e_{ij} is therefore not claimed to be strictly rotation invariant. For unit normals, $n_i^T n_j$ and $\|n_i \times n_j\|_2$ are algebraically related, but both are retained as explicit angular channels: the dot product preserves the signed cosine under the adopted orientation convention, whereas the cross-product magnitude provides the nonnegative sine magnitude directly. We do not claim that these two channels contain independent information.

Evaluation scope. Proposition 1 applies only to the scalar descriptor in Equation (5). Table 6 measures prediction-level behavior of the complete network at three tested angles, not descriptor-level invariance or group-wide invariance. The empirical conclusion is therefore limited to practical rotation tolerance under the reported protocol.

Algorithm 2 — Tangent Orientation Descriptor (TOD)

Input: $(p_i,p_j,n_i,n_j,\kappa_i,\kappa_j,h_i,h_j)$, with consistently oriented unit normals and $\delta>0$.

Output: rotation-invariant geometric descriptor t_{ij} and learned edge feature e_{ij}.

1: $\Delta p_{ij} \leftarrow p_j-p_i$; $r_{ij} \leftarrow \|\Delta p_{ij}\|_2$.
2: if $r_{ij}>\delta$ then $\hat{u}_{ij} \leftarrow \Delta p_{ij}/r_{ij}$ else $\hat{u}_{ij} \leftarrow 0$.
3: $c_1 \leftarrow n_i^T n_j$; $c_2 \leftarrow \|n_i \times n_j\|_2$.
4: $c_3 \leftarrow n_i^T \hat{u}_{ij}$; $c_4 \leftarrow n_j^T \hat{u}_{ij}$; $c_5 \leftarrow |\kappa_i-\kappa_j|$.
5: $t_{ij} \leftarrow [r_{ij},c_1,c_2,c_3,c_4,c_5]$.
6: $z_{ij} \leftarrow \mathrm{Concat}[h_i,h_j-h_i,t_{ij}]$.
7: $e_{ij} \leftarrow \phi\mathrm{TOD}(z_{ij})$, using the same mapping for every directed edge.
8: return t_{ij},e_{ij}.

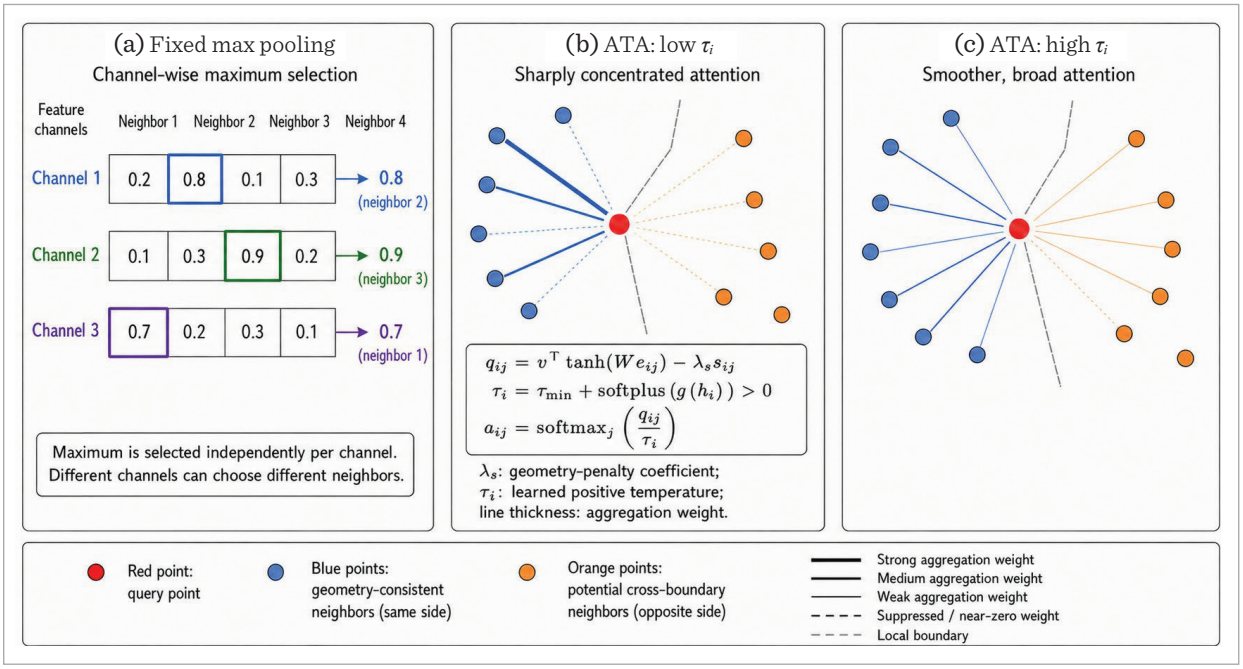
Descriptor check: evaluate $\|t_{ij}(RP,RN)-t_{ij}(P,N)\|_2$ for sampled R∈SO(3); this check does not apply to e_{ij} or to the complete network.

3.5. Adaptive Temperature-controlled Aggregation (ATA)

Fixed max pooling can discard context, whereas mean pooling can oversmooth edge responses. ATA learns a point-specific positive temperature and uses the ACoNS score as a separate geometry penalty. The up-

Figure 3

Comparison of fixed max pooling and Adaptive Temperature-controlled Aggregation (ATA). (a) Fixed max pooling selects the channel-wise maximum independently, so different channels may select different neighbors. (b) A low learned positive temperature concentrates attention on a small subset of geometry-consistent neighbors. (c) A high temperature produces a smoother attention distribution. The geometry penalty is denoted by λ_s, the learned temperature by τ_i, and line thickness indicates aggregation weight.



date preserves the incoming feature by direct concatenation with the aggregated message; it is a feature-fusion path rather than an additive residual connection.

$$q_{ij} = v^T \tanh(We_{ij}) - \lambda_s s_{ij}, \quad \lambda_s \geq 0 \tag{8}$$

$$\tau_i = \tau\min + \text{softplus}(g(h_i)), \quad \tau\min > 0 \tag{9}$$

$$a_{ij} = \exp(q_{ij}/\tau_i) / \sum_{k \in N_i} \exp(q_{ik}/\tau_i) \tag{10}$$

$$h_i' = \psi(\text{Concat}[h_i, \sum_{j \in N_i} a_{ij} e_{ij}]) . \tag{11}$$

A lower τ_i yields a sharper neighbor distribution, whereas a higher τ_i yields smoother mixing. The positive lower bound prevents division by zero and limits its gradient amplification. λ_s controls geometric penalization and is distinct from the temperature. The formulation does not establish a convergence guarantee, and the reported experiments are interpreted only as evidence for the tested implementation.

Algorithm 3 — Adaptive Temperature-controlled Aggregation (ATA)

Input: complete point-feature set $\{h_i\}_{i=1}^N$, edge sets $\{\{e_{ij}, s_{ij}\}_{j \in N_i}\}_{i=1}^N$, $\lambda_s \geq 0$, and $\tau\min > 0$.

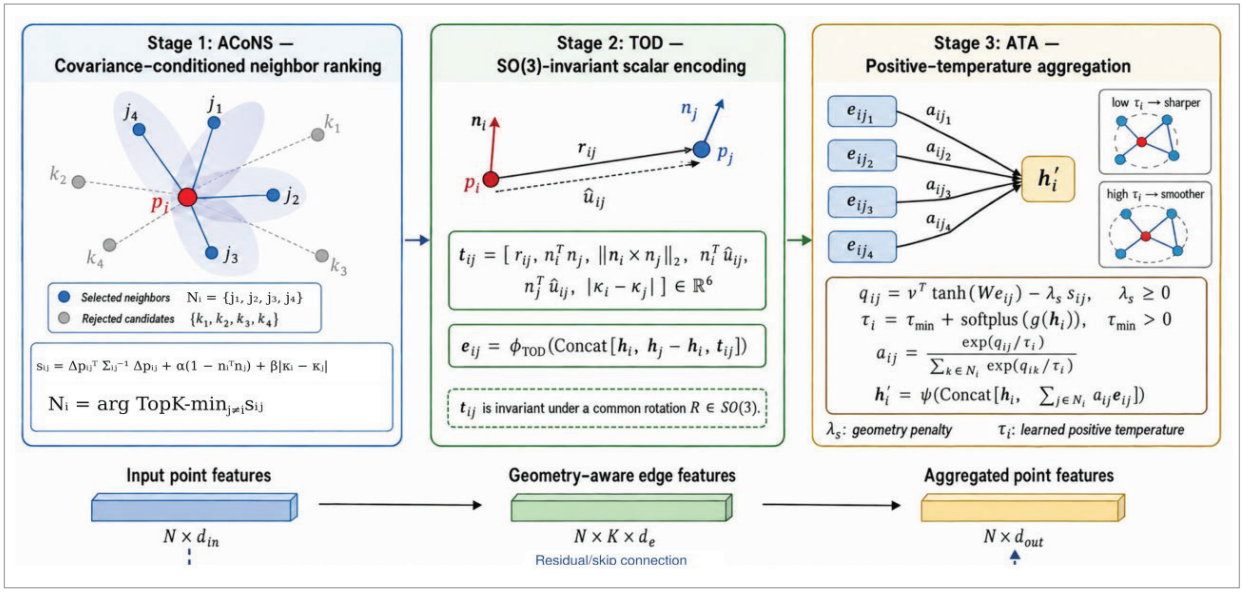
Output: updated point features $\{h_i'\}$ and normalized weights $\{a_{ij}\}$.

1: **for** each point $i=1,\ldots,N$ **do**
2: $\tau_i \leftarrow \tau\min + \text{softplus}(g(h_i))$.
3: **for** each $j \in N_i$ **do**
4: $\ell_{ij} \leftarrow v^T \tanh(We_{ij})$; $q_{ij} \leftarrow \ell_{ij} - \lambda_s s_{ij}$.
5: **end for**
6: $b_i \leftarrow \max_{j \in N_i}(q_{ij}/\tau_i)$ // numerical stabilization.
7: $\hat{w}_{ij} \leftarrow \exp(q_{ij}/\tau_i - b_i)$ for every $j \in N_i$.
8: $a_{ij} \leftarrow \hat{w}_{ij} / \Sigma_{k \in N_i} \hat{w}_{ik}$.
9: $m_i \leftarrow \Sigma_{j \in N_i} a_{ij} e_{ij}$.
10: $h_i' \leftarrow \psi(\text{Concat}[h_i, m_i])$.
11: **end for**
12: **return** $\{h_i'\}, \{a_{ij}\}$.

The update in line 10 is concatenative feature fusion, not residual addition.

Figure 4

CovaEdge block composed of covariance-conditioned neighbor ranking (ACoNS), SO(3)-invariant scalar pair encoding (TOD), and positive-temperature adaptive aggregation (ATA). The dashed blue path indicates that the incoming feature h_i is passed directly to the concatenation in Equation (11); it does not denote an additive residual connection.



The mechanism hypothesis is that ATA lowers attention entropy near genuine boundaries while retaining broader aggregation on smooth regions. Figure 3 illustrates this behavior, and Figure 4 shows the complete block.

Stacking CovaEdge blocks enlarges the receptive field while preserving the same module contract. Within each block, the incoming feature is retained through concatenative fusion with the aggregated message. The block-level outputs are not assumed to establish causality by themselves; Section 4 specifies the measurements used to evaluate the complete implementation.

4. Experiments

4.1. Experimental Setup and Reproducibility

Dataset. Experiments use the ABC CAD dataset [4]. The dataset is divided into 14,000/3,000/3,000 models for training/validation/testing and evaluates 1,024, 2,048, 4,096, and 8,192 sampled points, with 4,096 as the default. The evaluation uses the same preprocessing and label-generation procedure for all methods; however, the exact edge-label tolerance, sampling seed, and class balance are not reported.

Training. Models are trained with AdamW and cosine annealing for 200 epochs using a batch size of 16, random rotation, scaling, jitter, and point dropout. For each run, precision, recall, and F1 are first computed separately for each test shape from its point-wise binary edge predictions and then macro-averaged over the 3,000 test shapes. The reported table entries are the arithmetic means of the three seed-level shape-macro scores. Because precision, recall, and F1 are averaged independently, the reported F1 is not recomputed from the rounded aggregate precision and recall values. Run-wise outputs were not archived, so standard deviations and confidence intervals are not reported.

$$t_{ij} = [r_{ij},\ n_i^{T} n_j,\ \|n_i \times n_j\|_2,\ n_i^{T} \hat{u}_{ij},\ n_j^{T} \hat{u}_{ij},\ |\kappa_i - \kappa_j|]\,. \quad (12)$$

The loss weights in Equation (12) are fixed across all experiments. Their sensitivity was not evaluated separately; consequently, the ablation results isolate the tested architectural changes under a fixed loss configuration.

Evaluation scope. All comparisons use the ABC split and aggregate results. Statistical significance and cross-dataset generalization are not assessed, and robustness conclusions are restricted to the

Table 2
Performance on ABC (4,096 points) under the common evaluation protocol. Precision, recall, and F1 are shape-macro metrics: each metric is computed per test shape, macro-averaged over the 3,000 test shapes, and then averaged across three random seeds. F1 is therefore not recomputed from the rounded aggregate precision and recall values.

Method	Acc	Prec	Rec	F1	Params (M)
PointNet++	0.892	0.867	0.909	0.888	1.5
DGCNN	0.910	0.883	0.928	0.905	1.8
EdgeMovingNet	0.925	0.897	0.944	0.920	4.6
SED-Net	0.935	0.914	0.950	0.930	6.2
CovaEdge-Net	0.972	0.966	0.980	0.973	4.1

Table 3
Module ablation on ABC. ΔF1 is the signed change relative to the full model.

Variant	F1	ΔF1
Full CovaEdge-Net	0.973	0.000
ACoNS → Euclidean KNN	0.945	−0.028
TOD removed	0.952	−0.021
ATA → max pooling	0.955	−0.018
Learned τ → fixed τ	0.959	−0.014

Table 4
Sensitivity of ACoNS to the normal-consistency weight α and curvature-contrast weight β.

Varied parameter	α/α_0	β/β_0	F1	ΔF1
α	0	1	0.954	−0.019
α	0.5	1	0.961	−0.012
Default	1	1	0.973	0.000
α	2	1	0.966	−0.007
β	1	0	0.968	−0.005
β	1	0.5	0.974	+0.001
β	1	2	0.965	−0.008

density, rotation, and Gaussian-noise conditions evaluated below.

4.2. Predictive Results and Bounded Mechanism Analysis

Table 2 summarizes the ABC results at 4,096 points. Under this evaluation protocol, CovaEdge-Net obtains F1=0.973, which is 0.043 higher than SED-Net. This comparison is restricted to the listed baselines and the evaluated split; it is not presented as a universal state-of-the-art claim.

Table 3 reports the available module ablation. Replacing ACoNS with Euclidean KNN produces the largest F1 decrease, while removing TOD or replacing ATA also reduces F1. These results establish component relevance at the prediction level. They do not directly measure neighborhood purity, descriptor separability, or attention entropy.

Table 4 evaluates α/β sensitivity by varying one coefficient at a time while holding the other at its default value; α_0 and β_0 denote the default coefficients of the full model. The α sweep reaches its highest F1 at $\alpha/\alpha_0=1$: removing the normal-consistency term reduces F1 by 0.019, whereas doubling its weight reduces F1 by 0.007. For β, the setting $\beta/\beta_0=0.5$ obtains F1=0.974, only 0.001 above the default result. Because run-wise uncertainty is unavailable, this small difference is not interpreted as statistically significant. Removing β reduces F1 by 0.005, and doubling it reduces F1 by 0.008. Under the evaluated protocol, the larger decrease after removing α indicates that normal consistency has a stronger observed contribution than curvature contrast.

Figure 5 provides a qualitative view of the reported ablations. The full model produces more concentrated boundary responses, whereas the displayed variants show broader or smoother patterns. The image is used to illustrate the trends in Table 3 and is not treated as a substitute for feature-space or neighborhood-purity measurements.

Figure 6 illustrates why spatial proximity can connect points across holes, grooves, and intersecting surfaces. The side-by-side examples provide qualitative support for geometry-conditioned selection. Point-wise surface-membership labels are not used; therefore, the figure is interpreted qualitatively and no numerical cross-boundary rate is inferred.

4.3. Density, Rotation, Noise, and Efficiency

Table 5 reports density sensitivity from 1,024 to 8,192 points. F1 increases with sampling density for all methods, and the reported CovaEdge-Net values remain highest among the listed baselines. This result is limited to the ABC sampling protocol.

Figure 5

Qualitative module ablation. (a) The full model produces concentrated boundary responses. (b) Removing ACoNS is associated with potential cross-boundary leakage. (c) Removing TOD yields a blurred orientation response. (d) Replacing ATA with fixed pooling produces over-smoothed or noisy edges. Gray points denote input samples and red points denote predicted edges; quantitative conclusions are based on Table 3.

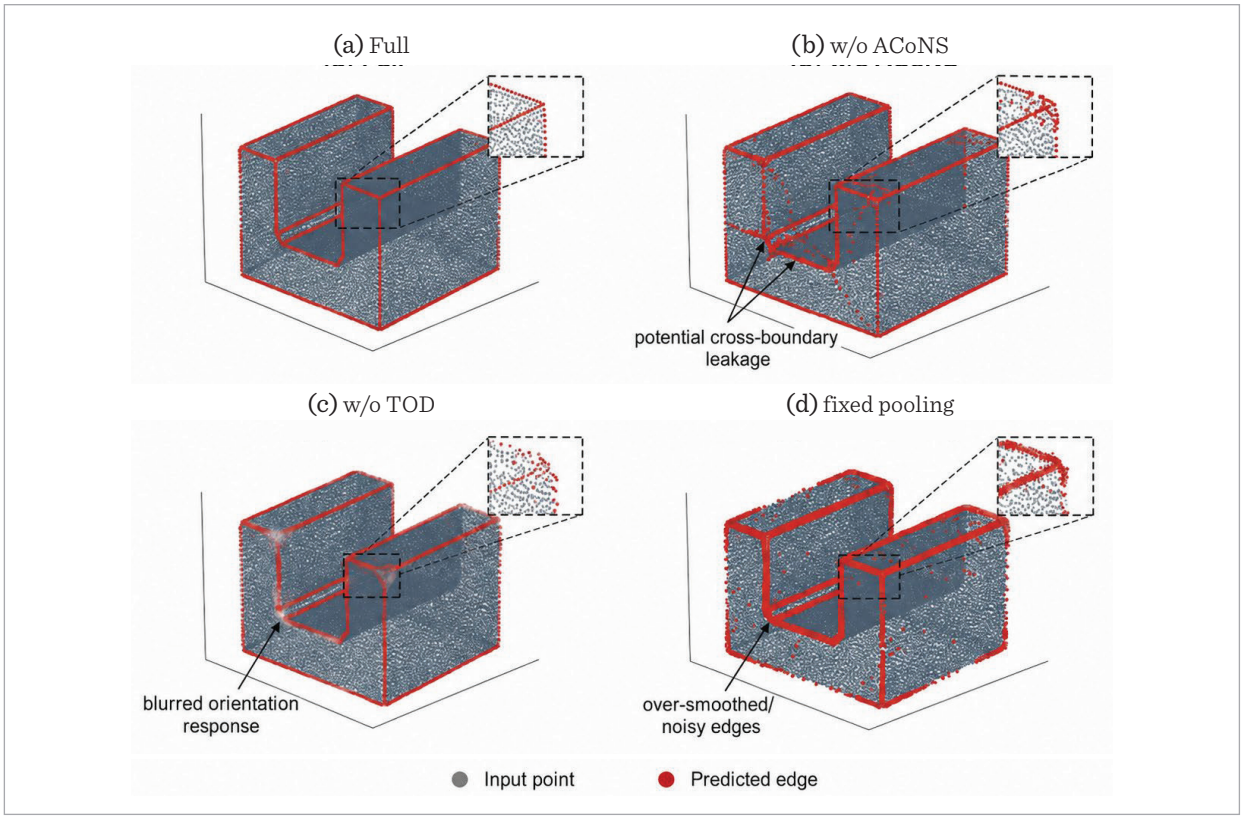


Figure 6

Qualitative neighborhood-selection comparison near three structural discontinuities. In each row, the left panel locates the local boundary region, the middle panel shows Euclidean KNN, and the right panel shows ACoNS. Black points denote point-cloud samples; the red point is the query; blue and green points are method-specific selections; purple points are shared neighbors; dashed gray ellipses indicate candidate regions; and orange circles mark potential cross-boundary neighbors. Blue and green lines denote the respective neighbor connections. This visualization is qualitative, and no cross-boundary rate is inferred.

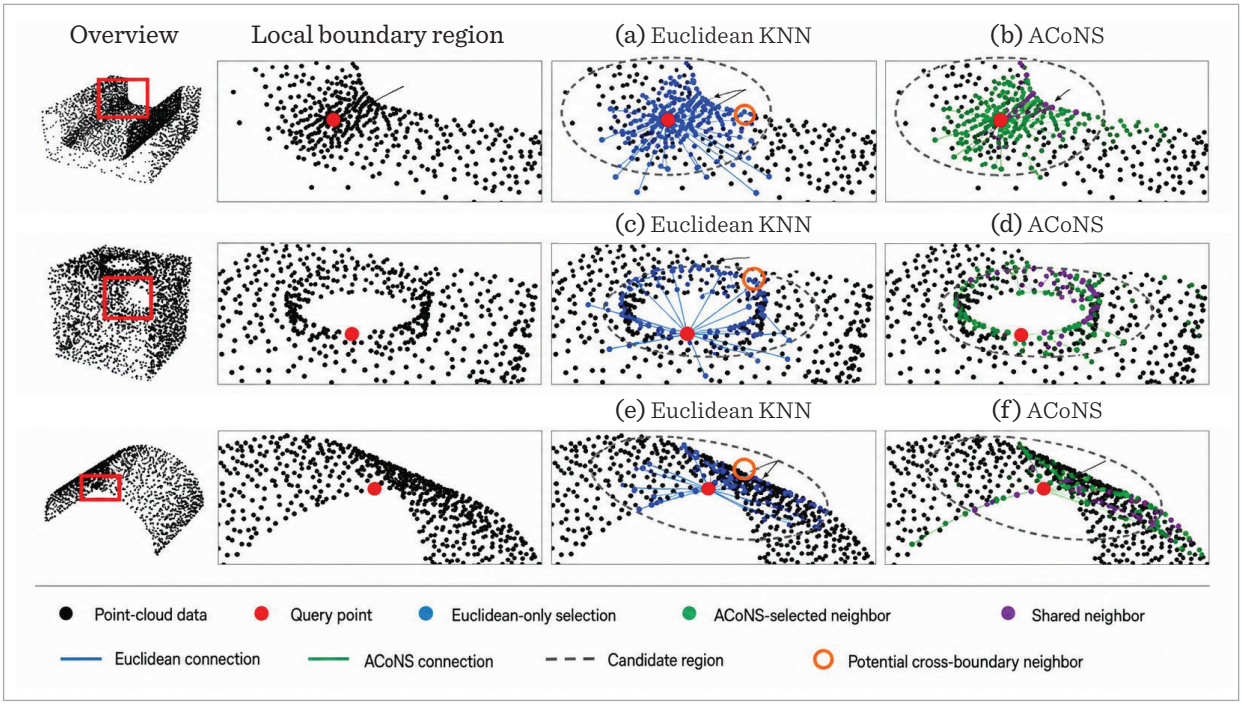


Table 5

F1 across point densities on the ABC benchmark.

Method	1024	2048	4096	8192
PointNet++	0.832	0.868	0.892	0.905
DGCNN	0.850	0.883	0.910	0.923
EdgeMovingNet	0.866	0.900	0.925	0.939
SED-Net	0.883	0.912	0.935	0.947
CovaEdge-Net	0.921	0.952	0.973	0.978

Rotation robustness.

Table 6 reports prediction-level F1 under rotations of 0°, 45°, and 90°. The maximum-minus-minimum F1 range is 0.001 for CovaEdge-Net, compared with 0.005–0.008 for the listed baselines. This small variation supports practical rotation tolerance at the tested angles but does not constitute a complete descriptor-level invariance or group-generalization test.

Noise robustness. Table 7 reports the Gaussian-noise evaluation. CovaEdge-Net retains the highest F1 among the listed methods at each noise level. Because the experiment does not isolate the effects of normal and covariance recomputation, the

Table 6
Rotation evaluation at 4,096 points for the tested angles. F1 range denotes the maximum minus the minimum across 0°, 45°, and 90°.

Method	0◦	45◦	90◦	F1 range
DGCNN	0.905	0.897	0.900	0.008
EdgeMovingNet	0.920	0.913	0.916	0.007
SED-Net	0.930	0.925	0.927	0.005
CovaEdge-Net	0.973	0.972	0.973	0.001

Table 7
Gaussian-noise evaluation at 4,096 points.

Method	Clean	σ=0.01	σ=0.02	σ=0.05
DGCNN	0.905	0.895	0.870	0.792
EdgeMovingNet	0.920	0.908	0.885	0.824
SED-Net	0.930	0.921	0.900	0.847
CovaEdge-Net	0.973	0.964	0.947	0.892

results represent end-to-end robustness of the complete preprocessing pipeline.

Efficiency. CovaEdge-Net contains 4.1 M parameters. On an RTX 4090 with batch size 32, a 4,096-point batch requires 14.2 ms, corresponding to 2,254 samples/s. These values report one end-to-end model-level efficiency setting. Baseline timing, peak training memory, per-module latency, and large-N scaling were not evaluated.

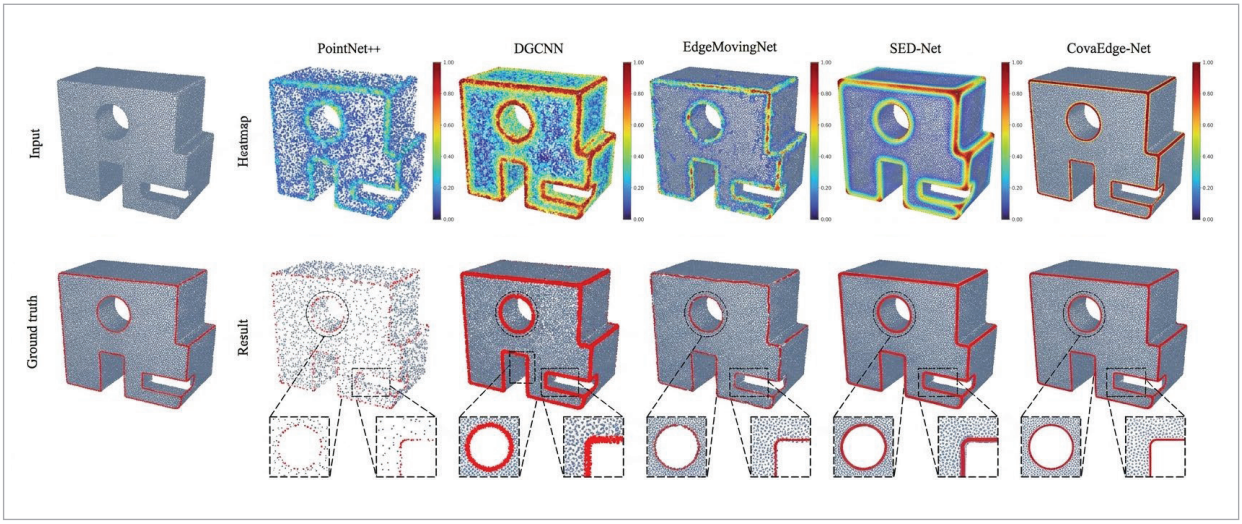
4.4. Qualitative Analysis

Figure 7 presents a qualitative comparison of industrial point-cloud edge predictions. Under the displayed visualization settings, CovaEdge-Net produces responses concentrated around circular holes and stepped junctions. Because per-shape errors are not reported, the comparison is interpreted qualitatively.

Table 8
Model size and available end-to-end inference efficiency. The CovaEdge-Net complexity entry refers to the decomposition in Section 3.3 because total cost depends on the candidate-search backend; it is not reduced to O(NK). The reported 14.2 ms is the latency of one batch of 32 samples on an RTX 4090; 32/0.0142 s gives approximately 2,254 samples/s. Corresponding baseline measurements were not recorded.

Method	Params (M)	Complexity	Batch latency (ms; batch=32)	Throughput (samples·s^{-1})
PointNet++	1.5	O(N log N)	Not reported	Not reported
DGCNN	1.8	O(NK)	Not reported	Not reported
EdgeMovingNet	4.6	O(NK)	Not reported	Not reported
SED-Net	6.2	O(NK)	Not reported	Not reported
CovaEdge-Net	4.1	See Sec. 3.3 (search-dependent)	14.2	2,254

Figure 7
Qualitative comparison of predicted edges on industrial point clouds.



5. Discussion

The formulation clarifies the intended role of each module. ACoNS changes which points communicate, TOD changes how local orientation relations are encoded, and ATA changes the concentration of neighbor messages. The module ablations support the relevance of all three components, with the largest F1 decrease caused by replacing ACoNS. Direct neighborhood-purity, feature-separability, and attention-entropy measurements were not performed; accordingly, the causal chain remains a mechanism hypothesis rather than a direct empirical conclusion.

Limitations. The study is limited to the ABC CAD benchmark and aggregate results from three runs; run-wise uncertainty and cross-dataset generalization were not evaluated. The method also depends on normals, curvature, covariance regularization, and several hyperparameters, which may be less reliable on extremely sparse or noisy scans. The efficiency evaluation does not separate preprocessing, module-level latency, or peak training memory. The study therefore does not establish domain generalization, statistical significance, real-time deployment, or mathematical convergence.

Future work will develop a hierarchical pipeline for million-point scans. Each scan will be partitioned into overlapping spatial blocks, representative points will be selected at multiple resolutions, and ACoNS candidate retrieval will proceed from coarse block-level search to fine neighborhood refinement within the selected blocks. Multiscale boundary features will then be fused before point-level prediction. A parallel normal-free branch will learn orientation cues from scalar local relations and will be compared with the current normal-dependent formulation. Evaluation will include real industrial LiDAR under occlusion and calibration drift, together with end-to-end measurements of sampling, preprocessing, and edge-inference latency.

6. Conclusion

CovaEdge-Net combines geometry-conditioned neighbor ranking, rotation-aware local encoding, and positive-temperature aggregation for industrial point-cloud edge detection. On the ABC benchmark, the model achieves F1=0.973, and the module ablations show that each component contributes to prediction quality. The analysis clarifies the mathematical role of the three modules and limits the conclusions to the evaluated benchmark, baselines, and aggregate results.

References

1. Bronstein, M. M., Bruna, J., Cohen, T., Veličković, P. Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. arXiv, 2021, arXiv:2104.13478. DOI: 10.48550/arXiv.2104.13478.
2. Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y. Hypergraph Neural Networks. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(1), 3558-3565. https://doi.org/10.1609/aaai.v33i01.33013558
3. Jiang, C., Wang, K., Yu, H., Li, Y. Dimensional Transfor-mation-Based Point Cloud Feature Extraction for Pilot-Point Guided Path Planning in the Robotic Welding. Measurement, 2026, 272, 121071. https://doi.org/10.1016/j.measurement.2026.121071
4. Koch, S., Matveev, A., Jiang, Z., Williams, F., Artemov, A., Burnaev, E., Alexa, M., Zorin, D., Panozzo, D. ABC: A Big CAD Model Dataset for Geometric Deep Learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recog-nition, 2019, 9593-9603. https://doi.org/10.1109/CVPR.2019.00983
5. Lee, J., Lee, Y., Kim, J., Kosiorek, A. R., Choi, S., Teh, Y. W. Set Transformer: A Framework for Attention-Based Permuta-tion-Invariant Neural Networks. Proceedings of the 36th Inter-national Conference on Machine Learning, PMLR, 2019, 97, 3744-3753. arXiv:1810.00825.
6. Li, Y., Liu, S., Yang, X., Guo, J., Guo, J., Guo, Y. Surface and Edge Detection for Primitive Fitting of Point Clouds. ACM SIG-GRAPH 2023 Conference Proceedings, 2023, 44, 1-10. https://doi.org/10.1145/3588432.3591522
7. Limberger, F. A., Oliveira, M. M. Real-Time Detection of Planar Regions in Unorganized Point Clouds. Pattern Recognition, 2015, 48(6), 2043-2053. https://doi.org/10.1016/j.patcog.2014.12.020
8. Liu, Z., Xin, X., Xu, Z., Zhou, W., Wang, C., Chen, R., He, Y. Robust and Accurate Feature Detection on Point Clouds. Computer-Aided Design, 2023, 164, 103592. https://doi.org/10.1016/j.cad.2023.103592
9. Qi, C. R., Su, H., Mo, K., Guibas, L. J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 77-85. https://doi.org/10.1109/CVPR.2017.16
10. Qi, C. R., Yi, L., Su, H., Guibas, L. J. PointNet++: Deep Hierar-chical Feature Learning on Point Sets in a Metric Space. Advances in Neural Information Processing Systems, 2017, 30, 5099-5108. arXiv:1706.02413.
11. Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., Ghanem, B. PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies. Advances in Neural Information Processing Systems, 2022, 35, 23192-23204. https://doi.org/10.52202/068431-1685
12. Qian, R., Lai, X., Li, X. BADet: Boundary-Aware 3D Object Detection from Point Clouds. Pattern Recognition, 2022, 125, 108524. https://doi.org/10.1016/j.patcog.2022.108524
13. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Ilic, S., Xu, K. GeoTransformer: Fast and Robust Point Cloud Registration with Geometric Transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(8), 9806-9821. https://doi.org/10.1109/TPAMI.2023.3259038
14. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K. Geometric Transformer for Fast and Robust Point Cloud Registration. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, 11143-11152. https://doi.org/10.1109/CVPR52688.2022.01086
15. Raina, P., Mudur, S., Popa, T. Sharpness Fields in Point Clouds Using Deep Learning. Computers & Graphics, 2019, 78, 37-53. https://doi.org/10.1016/j.cag.2018.11.003
16. Rusu, R. B., Cousins, S. 3D Is Here: Point Cloud Library (PCL). Proceedings of the 2011 IEEE International Conference on Ro-botics and Automation, 2011, 1-4. https://doi.org/10.1109/ICRA.2011.5980567
17. Spellings, M. Geometric Algebra Attention Networks for Small Point Clouds. arXiv, 2021, arXiv:2110.02393. DOI: 10.48550/arXiv.2110.02393.
18. Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J. KPConv: Flexible and Deformable Convolution for Point Clouds. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, 6411-6420. https://doi.org/10.1109/ICCV.2019.00651
19. Wang, K., Yu, H., Jiang, C., Li, Y. A Novel Adaptive Point Cloud Framework for Robust Weld Seam Extraction in Complex Free-Form Workpieces. Measurement,

2026, 282, 122065. https://doi.org/10.1016/j.measurement.2026.122065

20. Wang, X., Xu, Y., Xu, K., Tagliasacchi, A., Zhou, B., Mahda-vi-Amiri, A., Zhang, H. PIE-NET: Parametric Inference of Point Cloud Edges. Advances in Neural Information Processing Sys-tems, 2020, 33, 20167-20178.
21. Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Sol-omon, J. M. Dynamic Graph CNN for Learning on Point Clouds. ACM Transactions on Graphics, 2019, 38(5), 146, 1-12. https://doi.org/10.1145/3326362
22. Weinmann, M., Jutzi, B., Hinz, S., Mallet, C. Semantic Point Cloud Interpretation Based on Optimal Neighborhoods, Rele-vant Features and Efficient Classifiers. ISPRS Journal of Pho-togrammetry and Remote Sensing, 2015, 105, 286-304. https://doi.org/10.1016/j.isprsjprs.2015.01.016
23. Wiersma, R., Nasikun, A., Eisemann, E., Hildebrandt, K. Del-taConv: Anisotropic Operators for Geometric Deep Learning on Point Clouds. ACM Transactions on Graphics, 2022, 41(4), 105, 1-10. https://doi.org/10.1145/3528223.3530166
24. Xie, Y., Tu, Z., Yang, T., Zhang, Y., Zhou, X. EdgeFormer: Local Patch-Based Edge Detection Transformer on Point Clouds. Pattern Analysis and Applications, 2025, 28(1), 11. https://doi.org/10.1007/s10044-024-01386-6
25. Xu, J., Tang, X., Zhu, Y., Sun, J., Pu, S. SGMNet: Learning Ro-tation-Invariant Point Cloud Representations via Sorted Gram Matrix. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, 10448-10457. https://doi.org/10.1109/ICCV48922.2021.01030
26. Yang, X., Ji, D., Li, Y., Xie, J., Guo, J., Guo, Y. EdgeMovingNet: Edge-Preserving Point Cloud Reconstruction via Joint Geometry Features. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025, 22150-22160. https://doi.org/10.1109/CVPR52734.2025.02063
27. Yu, L., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A. EC-Net: An Edge-Aware Point Set Consolidation Network. Computer Vi-sion-ECCV 2018, Part VII, 2018, 398-414. https://doi.org/10.1007/978-3-030-01234-2_24
28. Zeng, Z., Dong, M., Zhou, J., Qiu, H., Dong, Z., Luo, M., Li, B. DeepLA-Net: Very Deep Local Aggregation Networks for Point Cloud Analysis. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025, 1330-1341. https://doi.org/10.1109/CVPR52734.2025.00132
29. Zhao, H., Jiang, L., Jia, J., Torr, P. H. S., Koltun, V. Point Transformer. Proceedings of the IEEE/CVF International Con-ference on Computer Vision, 2021, 16259-16268. https://doi.org/10.1109/ICCV48922.2021.01595
30. Zhao, W., Zhang, R., Wang, Q., Cheng, G., Huang, K. BFANet: Revisiting 3D Semantic Segmentation with Boundary Feature Analysis. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025, 29395-29405. https://doi.org/10.1109/CVPR52734.2025.02737
31. Zhou, J., Huang, H., Liu, B., Liu, X. Normal Estimation for 3D Point Clouds via Local Plane Constraint and Multi-Scale Selec-tion. Computer-Aided Design, 2020, 129, 102916. https://doi.org/10.1016/j.cad.2020.102916
32. Zhou, J., Jin, W., Wang, M., Liu, X., Li, Z., Liu, Z. Improvement of Normal Estimation for Point Clouds via Simplifying Surface Fitting. Computer-Aided Design, 2023, 161, 103533. https://doi.org/10.1016/j.cad.2023.103533

This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).