Computers & Graphics 138 (2026) 104672

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Technical Section

Dynamic correlation network and structure-aware matching for robust point cloud registration☆

Lei Lu [a], Meichen Pan [a], Ling Cao [b,c], Renlong Qi [e], Peng Li [a], Wei Pan [b,d,*]

[a] Institute for Complexity Science, Henan University of Technology, Zheng Zhou, 450001, He Nan, China
[b] OPT Machine Vision, Department of R&D, 66 Xingfa South Road, Dongguan, Guangdong Province, China
[c] College of Electronics and Information Engineering, Shenzhen University, Nanshan District, Shenzhen, Guangdong Province, China
[d] OPT Machine Vision Tech Co., LTD. Japan, Department of R&D, Aomi 2-7-4-735, Koto, Tokyo, Japan
[e] Zhengzhou University of Science and Technology, Zheng Zhou, 450064, He Nan, China

ARTICLE INFO

Dataset link: https://github.com/pmc555/DC-SAM

Keywords:
Point cloud registration
Rotation-invariant features
Dynamic correlation learning
Linear self-attention
Structure-aware matching
3D perception
Deep learning

ABSTRACT

Relative rotations between point clouds pose significant challenges for reliable correspondence estimation. Although many recent learning-based methods attempt to improve rotation robustness through data augmentation, such strategies cannot adequately cover the continuous SO(3) space, often leading to unstable matching performance under unseen orientations. To address this issue, we propose a rotation-robust point cloud registration framework that integrates a Dynamic Correlation Network (DCN) with a Structure-Aware Point Matching (SAPM) module. The proposed DCN builds upon rotation-equivariant features and dynamically models local feature correlations through adaptive channel and spatial weighting, enabling consistent correlation modeling in a rotation-invariant feature space and alleviating correspondence instability caused by inconsistent feature correlations. This design improves the consistency and discriminative power of feature correlation modeling under varying rotations. Based on the enhanced features, we further introduce the SAPM module to refine patch-level correspondences. SAPM incorporates contextual interactions and enforces bidirectional consistency to further mitigate matching instability caused by inconsistent feature correlations. Extensive experiments on both indoor (3DMatch, 3DLoMatch) and outdoor (KITTI) benchmarks demonstrate that the proposed method achieves superior accuracy and robustness under varying rotations compared to state-of-the-art approaches.

1. Introduction

Accurate registration of partially overlapping 3D point clouds is a fundamental task in many real-world applications, including autonomous driving, robotic manipulation, augmented reality, and digital twin construction. In a typical industrial or digital reconstruction pipeline, multiple scans of the same scene are captured from different viewpoints using LiDAR or RGB-D sensors. These scans are then aligned through a point cloud registration process that estimates the rigid transformation between overlapping point sets. Reliable registration is essential for downstream tasks such as object reconstruction and scene understanding.

A key challenge in point cloud registration lies in maintaining consistent feature correlations under varying relative rotations, as inconsistencies in feature correlations can directly lead to unstable correspondence estimation. Although real-world point clouds often suffer from noise and uneven sampling, recent geometry-processing techniques [1–3] help stabilize local neighborhood structures, highlighting the importance of reliable low-level geometry for correspondence estimation. However, even with improved preprocessing, learning-based feature extractors often struggle to maintain consistent correlation modeling under large rotations.

Classic learning-based point cloud networks such as PointNet [4], PointNet++ [5], and PointNeXt [6] extract discriminative representations but lack inherent rotation robustness. Convolution-based methods including PointCNN [7] and PAConv [8] learn spatially adaptive kernels but rely on raw coordinate differences, making them sensitive to orientation changes and limiting their robustness under varying rotations. Graph-based approaches such as DGCNN [9], GAT [10], and adaptive graph convolution [11] effectively model local geometry but still depend on static correlation mechanisms that fail to generalize across varying geometric configurations.

To improve robustness to rotations, recent registration frameworks explore equivariant or invariant neural architectures. Many learning-based methods also rely on data augmentation to enhance rotation

☆ This article was recommended for publication by Yuki Koyama.

* Corresponding author at: OPT Machine Vision Tech Co., LTD. Japan, Department of R&D, Aomi 2-7-4-735, Koto, Tokyo, Japan.
E-mail address: vpan@foxmail.com (W. Pan).

https://doi.org/10.1016/j.cag.2026.104672
Received 15 January 2026; Received in revised form 8 June 2026; Accepted 18 June 2026
Available online 25 June 2026

generalization. However, such strategies cannot fully cover the continuous SO(3) space, often leading to degraded performance when encountering unseen rotations. For example, topology-aware transformers such as TopFormer [12] incorporate structural priors into attention mechanisms, while semantic-constrained registration methods like SemReg [13] exploit high-level contextual cues to improve pose estimation. Although effective, these approaches do not explicitly model consistent feature correlations under rotation, and their feature representations may still become unstable across varying orientations.

Motivated by these limitations, we propose a structure-aware point cloud registration framework that combines adaptive correlation modeling with efficient contextual reasoning. The network first extracts rotation-equivariant point features using a vector-neuron-based encoder. Building upon this, we introduce a **Dynamic Correlation Network (DCN)**, which predicts consistent correlation weights in a rotation-invariant feature space based on local geometric relationships, enabling stable feature correlation modeling across different orientations and improving both consistency and discriminative power.

To further enhance matching stability, we introduce a **Structure-Aware Point Matching (SAPM)** module. SAPM encodes patch-level contextual interactions using linear self-attention and enforces bidirectional matching consistency, thereby mitigating correspondence instability caused by inconsistent feature correlations.

The main contributions of this work are summarized as follows:

(1) We propose a Dynamic Correlation Network (DCN) that performs adaptive correlation modeling in a rotation-invariant feature space, enabling robust and consistent feature correlations under arbitrary rotations.

(2) We introduce a lightweight Structure-Aware Point Matching (SAPM) module that captures contextual dependencies and enforces bidirectional consistency, improving the reliability of fine-grained correspondences.

(3) We integrate the proposed components into a unified coarse-to-fine registration framework and demonstrate through extensive experiments on 3DMatch, 3DLoMatch, and KITTI that the proposed method achieves superior accuracy and robustness compared with recent state-of-the-art approaches.

2. Related work

2.1. Learning-based point cloud registration

Deep learning has significantly advanced point cloud registration by enabling more expressive and robust geometric representations. Early point-based networks such as PointNet [4] and PointNet++ [5] introduce foundational point-wise and hierarchical feature extraction paradigms. These ideas are further developed by PointNeXt [6] with improved training strategies and by PointCNN [7], which applies $\mathcal{X}$-transforms to learn point-set canonical ordering.

Graph-based approaches play an essential role in capturing local geometric relations. DGCNN [9] constructs dynamic graphs to learn edge features, while RGCNN [14] introduces regularization to stabilize graph convolution. Attention-based graph operators, including GAC-Net [15] and GAT [10], enhance neighborhood modeling by learning feature-dependent attention weights. Adaptive graph convolution [11] and attentive filtering [16] further improve local structure adaptivity.

More recently, several works directly target registration. TopFormer [12] introduces topology-aware transformer encoding to improve correspondence estimation by leveraging structural priors, demonstrating the importance of topology-sensitive attention mechanisms. SemReg [13] integrates semantic constraints into the registration pipeline to enhance robustness in complex scenes. Although these advances highlight the value of incorporating topology or semantics, they still rely on rotation-sensitive feature extractors or do not explicitly learn adaptive invariant correlations at the geometric level. In contrast, our method introduces a Dynamic Correlation Network (DCN) that constructs rotation-invariant and structure-adaptive features directly from equivariant representations.

2.2. Transformers in 3D vision

Transformers have become powerful tools for geometric reasoning in 3D vision. Point Transformer [17] demonstrates the effectiveness of attention-based local aggregation, while PCT [18] and Stratified Transformer [19] extend transformer architectures with patch grouping and stratified attention. Swin3D [20] applies hierarchical windowed attention to large-scale indoor scene understanding. These developments are inspired by advances in self-attention [21,22], which enable expressive modeling of long-range dependencies.

Beyond general 3D perception tasks, transformers have also been extensively adopted for point cloud registration by modeling cross-cloud feature interactions and long-range correspondences. CoFiNet [23] employs a combination of self-attention and cross-attention for coarse feature matching, followed by optimal transport for fine-grained correspondence refinement. GeoTransformer [24] further enhances registration robustness by introducing rotation-invariant geometric positional encoding, achieving strong performance and efficiency. Building upon this paradigm, subsequent works explore complementary improvements, including RoITr [25], which designs a rotation-invariant global transformer, and PEAL [26], which explicitly injects overlap priors to improve matching in low-overlap scenarios. More recently, DFAT [27] adopts double-layer focused attention to further enhance correspondence quality.

However, standard attention mechanisms suffer from quadratic computational cost and typically rely on descriptors produced by static correlation modules. Even topology-aware transformers such as TopFormer [12] focus primarily on high-level attention design rather than local invariant correlation modeling. Our Structure-Aware Point Matching (SAPM) module adopts *linear* self-attention to efficiently capture contextual cues while enforcing mutual matching consistency through a row–column masked softmax, complementing DCN for structurally coherent registration.

2.3. Convolution-based methods

Convolution-based neural networks have been widely explored for 3D geometric representation learning. Voxel-based 3D convolutional networks such as VoxelNet [28], SECOND [29], and PV-RCNN [30] provide strong performance in detection and large-scale perception by discretizing point clouds into regular grids. However, voxelization inevitably sacrifices fine-grained geometric details, which are critical for correspondence-level registration.

Beyond voxel-based formulations, several convolution-based approaches operate directly on point clouds for registration. FCGF [31] employs a sparse 3D convolutional encoder–decoder architecture for dense descriptor learning. SpinNet [32] introduces cylindrical convolutions to extract rotation-invariant patch-wise descriptors. Predator [33] combines graph convolution and cross-attention to enhance descriptor discrimination and overlapping region prediction, achieving robust performance in low-overlap scenarios.

Several approaches further exploit group-equivariant representations, including YOHO [34], RoReg [35], and RoITr [25], which leverage group convolution or invariant feature construction to improve robustness under arbitrary rotations. PARE-Net [36] proposes a position-aware convolution to better capture unique local geometric patterns and jointly learns rotation-equivariant and rotation-invariant representations through continuous SO(3)-equivariant Vector Neurons.

Although these methods have significantly improved rotational robustness, they mainly focus on learning rotation-invariant or rotation-equivariant descriptors. How to maintain stable feature correlations for subsequent local feature aggregation and correspondence estimation under arbitrary rotations remains insufficiently explored.

3. Methods

Given two point clouds $P = \{p_i \in \mathbb{R}^3 \mid i = 1, \dots, N\}$ and $Q = \{q_j \in \mathbb{R}^3 \mid j = 1, \dots, M\}$, the objective of rigid registration

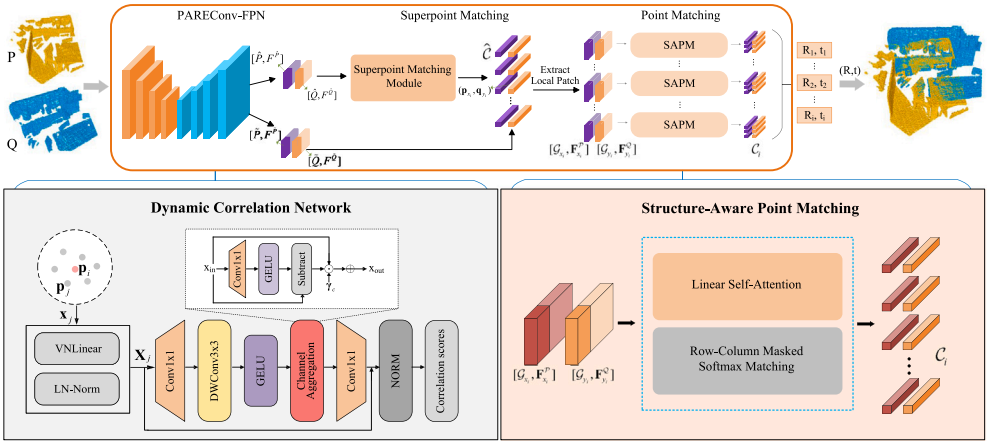

Fig. 1. Overview of the proposed registration framework. A PAREConv-FPN backbone augmented with a Dynamic Correlation Network extracts rotation-robust multi-scale features from input point clouds, yielding coarse superpoints and dense point-level descriptors. Coarse superpoint correspondences define local patches, within which the proposed Structure-Aware Point Matching module establishes dense correspondences. Multiple rigid transformation hypotheses are estimated from dense correspondences using SVD, and the optimal solution is selected as the final transformation.

is to estimate a transformation $T = \{R, t\}$, where $R \in SO(3)$ and $t \in \mathbb{R}^3$, such that P is aligned to Q. Our method follows a coarse-to-fine paradigm, combining (1) a Dynamic Correlation Network (DCN) for geometry-aware, rotation-robust feature extraction, and (2) a Structure-Aware Point Matching (SAPM) module for context-enhanced dense correspondence estimation.

We first use a DCN-enhanced PAREConv-FPN backbone [36] to extract multiscale geometric features. The backbone outputs (i) coarse superpoints $\{\hat{P}, \hat{Q}\}$ with features $\{F^{\hat{P}}, F^{\hat{Q}}\}$, and (ii) dense point-level features $\{F^{\widetilde{P}}, F^{\widetilde{Q}}\}$ associated with the upsampled points $\{\widetilde{P}, \widetilde{Q}\}$. Superpoints are used to construct coarse correspondences, while SAPM refines them by modeling local structural dependencies via linear self-attention. Each refined patch produces a dense correspondence set, and each set yields a transformation hypothesis through SVD-based orthogonal Procrustes. Finally, a feature-based hypothesis proposer selects the optimal hypothesis $T = \{R, t\}$ without requiring RANSAC.

In the following subsections, we describe each major component in detail.

3.1. Overview

Our approach begins by extracting multiscale features using a DCN-augmented PAREConv-FPN backbone, where the Dynamic Correlation Network adaptively modulates convolutional kernels based on rotation-invariant geometric cues. The backbone produces two types of representations:

- **Superpoints**: coarsely sampled points $\{\hat{P}, \hat{Q}\}$ with features $\{F^{\hat{P}}, F^{\hat{Q}}\}$, used for coarse-level matching and patch construction.
- **Dense features**: upsampled points $\{\widetilde{P}, \widetilde{Q}\}$ with invariant descriptors $\{F^{\widetilde{P}}, F^{\widetilde{Q}}\}$, used for fine-grained correspondence estimation.

Following GeoTransformer-style [24] superpoint matching, each matched superpoint pair induces a pair of dense local patches. Within each patch, our Structure-Aware Point Matching (SAPM) refines local descriptors using linear self-attention and performs robust bidirectional softmax matching. Dense correspondences aggregated from all patches form a correspondence set C from which multiple rigid transformation hypotheses $\{T_i\}$ are computed via SVD. A feature-based hypothesis proposer selects the final transformation $T = \{R, t\}$. An overview of the pipeline is shown in Fig. 1.

3.2. Backbone

We build our backbone upon the PAREConv-FPN framework [36], which adopts a vector-neuron (VN) equivariant encoder–decoder architecture to hierarchically extract geometric features from point clouds. In our implementation, the source and target point clouds are concatenated and processed jointly through a single backbone network in a weight-sharing Siamese manner. Both branches share identical parameters during training and inference, ensuring consistent feature embedding in a unified representation space.

The encoder follows a multi-stage hierarchical design consisting of progressive subsampling and residual PAREConv blocks. At each stage, local neighborhoods are constructed via k-nearest neighbors, and rotation-equivariant geometric features are aggregated. These features include relative offsets, local centroid differences, and cross-product directions, which are encoded using vector-neuron layers to preserve equivariance under arbitrary 3D rotations. Through successive downsampling, the encoder produces a sparse set of high-level points at the deepest stage, referred to as superpoints. These superpoints capture large-scale structural context with enlarged receptive fields and serve as the basis for coarse correspondence estimation.

To address the limited adaptability of fixed correlation modeling, we introduce a Dynamic Correlation Network (DCN) to upgrade the kernel aggregation mechanism in PAREConv. Unlike conventional MLP-based correlation functions with static parameterization, DCN formulates correlation estimation as a dynamic, structure-conditioned process. It exploits rotation-invariant descriptors to derive stable geometric relationships and incorporates channel-aware reallocation to emphasize informative feature dimensions, thereby enabling geometry-adaptive and rotation-robust convolution.

At the coarsest level, the superpoint features are transformed into rotation-invariant descriptors through a VN standardization layer. These invariant features are used to establish superpoint-level correspondences and define local patches for subsequent refinement.

The decoder adopts a feature pyramid design with skip connections from intermediate encoder stages. By propagating features from coarse to fine resolutions, the decoder progressively recovers dense point-wise representations aligned with the original point cloud. At the final decoding stage, both rotation-invariant and rotation-equivariant features are produced. The invariant descriptors are used for fine-grained correspondence matching, while the equivariant features preserve orientation information for accurate transformation estimation.

Overall, the backbone provides a unified hierarchical representation that integrates geometry-aware aggregation and rotation robustness. The superpoint features extracted by the backbone are used to establish coarse correspondences and define local patches. Dense features within each patch are subsequently refined by the Structure-Aware Point Matching (SAPM) module to obtain accurate point-level correspondences.

3.3. Dynamic Correlation Network (DCN)

3.3.1. Motivation

Local feature extraction is critical for reliable correspondence estimation in point cloud registration. However, under varying relative rotations, maintaining consistent feature correlations within local neighborhoods remains challenging. Many existing convolutional designs rely on fixed kernel aggregation, which applies identical correlation modeling across different local regions and thus struggles to preserve stable relationships between features under varying orientations.

For instance, PAConv [8] improves expressiveness through position-adaptive kernels, but it depends on raw coordinate differences and remains sensitive to pose variations. PARE-Net [36] introduces rotation-equivariant vector-neuron (VN) features, yet its correlation modeling is realized by a fixed MLP, limiting its ability to maintain consistent feature correlations under varying rotations.

To address these limitations, we propose the Dynamic Correlation Network (DCN), which focuses on learning consistent and adaptive feature correlations under rotation. Specifically, DCN leverages rotation-equivariant features and predicts correlation weights in a rotation-invariant feature space, enabling adaptive kernel aggregation that preserves consistent feature correlations under rotation.

3.3.2. Rotation-equivariant input encoding

Given a point cloud $P = \{p_i \in \mathbb{R}^3 \mid i = 1, \dots, N\}$, we use a vector neuron [37] encoder to extract rotation-equivariant features. To extract rotation-equivariant local geometric features, we first calculate the relative coordinate vector $\mathbf{p}_{ij}$ between the query point and its neighboring points. Subsequently, calculate the average direction vector of this local neighborhood.

To encode local geometric orientation cues, we further compute the cross product $\mathbf{v}_{ij}$ between $\mathbf{p}_{ij}$ and $\bar{\mathbf{p}}_i$. Finally, we concatenate these features to represent the local spatial structure:

$$\mathbf{x}_j = [\mathbf{p}_{ij} = \mathbf{p}_j - \mathbf{p}_i,\ \frac{1}{|\mathcal{N}_i|}\sum_j \mathbf{p}_{ij},\ \mathbf{v}_{ij}] \in \mathbb{R}^{d\times 3}, \qquad \mathbf{X}_j = \mathrm{VN}(\mathbf{x}_j). \tag{1}$$

where $\mathcal{N}_i$ is the K-nearest neighbor set of the point $\mathbf{p}_i$. To construct rotation-invariant correlation descriptors, we take the magnitude of VN features:

$$\tilde{\mathbf{X}}_j = \|\mathbf{X}_j\|_2, \tag{2}$$

ensuring that orientation information is removed while geometric structure is preserved.

3.3.3. Channel Reallocation Network

Inspired by recent channel-aware architectures (MogaNet [38]), we introduce a Channel Reallocation Network as a core component of DCN. Instead of directly regressing correlation scores with a static MLP, Channel reallocation network performs data-dependent channel-wise reallocation, allowing the network to adaptively emphasize geometry-relevant feature channels when constructing local correlations. Fixed MLP-based correlation modeling applies the same channel mixing to all local neighborhoods. By contrast, the channel reallocation network enables input-adaptive channel competition. This results in more discriminative and structure-aware correlation embeddings.

As illustrated in Fig. 1, the channel reallocation network can be summarized as

$$\begin{aligned}\mathbf{U} &= \mathrm{GELU}\big(\mathrm{DWConv}_{3\times3}\left(\mathrm{Conv}_{1\times1}(\tilde{\mathbf{X}}_j)\right)\big),\\ \mathbf{Z} &= \mathrm{Conv}_{1\times1}(\mathrm{CA}(\mathbf{U})) + \tilde{\mathbf{X}}_j,\end{aligned} \tag{3}$$

where $\tilde{\mathbf{X}}_j$ denotes the input rotation-invariant descriptor, DWConv is a depthwise convolution operating independently on each channel, and the residual connection preserves the original feature responses.

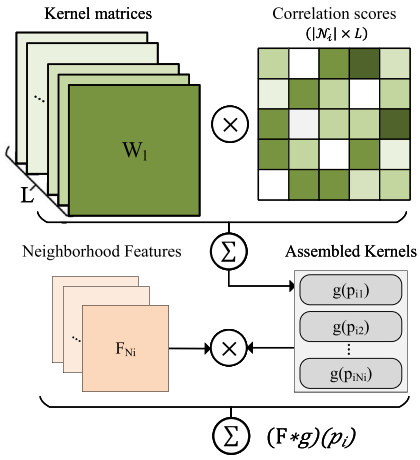

Fig. 2. Workflow of dynamic kernel assembly for point cloud convolution. Let $\mathcal{N}_i$ denote the neighborhood of the center point $\mathbf{p}_i$, containing N_i neighboring points. The workflow consists of four key components: (1) a learnable kernel bank $\{\mathbf{W}_l\}_{l=1}^{L}$, (2) an $N_i \times L$ Softmax-normalized correlation score matrix, (3) dynamically assembled kernels $\{g(\mathbf{p}_{i1}), \dots, g(\mathbf{p}_{iN_i})\}$, and (4) neighborhood features $\{\mathbf{F}_j\}$. The dynamic kernel associated with the jth neighbor is defined as $g(\mathbf{p}_{ij}) = \sum_{l=1}^{L} \gamma_{jl}\mathbf{W}_l$, where γ_{jl} denotes the correlation score between the jth neighbor and the lth kernel basis.

The core operation $\mathrm{CA}(\cdot)$ does not compute explicit attention weights. Instead, it redistributes channel responses by decomposing features into shared and complementary components:

$$\mathrm{CA}(\mathbf{U}) = \mathbf{U} + \boldsymbol{\gamma}_c \odot \left(\mathbf{U} - \mathrm{GELU}(\mathbf{U}\mathbf{W}_r)\right). \tag{4}$$

where $\mathbf{W}_r$ is a channel-reducing projection and $\boldsymbol{\gamma}_c$ is a channel-learning scaling factor, initialized with small values for stable optimization.

The projection $\mathbf{U}\mathbf{W}_r$ captures the shared response across channels, while the residual term $(\mathbf{U} - \mathrm{GELU}(\mathbf{U}\mathbf{W}_r))$ represents complementary channel-specific information. By scaling and reinjecting this residual, the channel reallocation network implicitly induces channel competition, encouraging discriminative channels to be emphasized while suppressing redundant responses.

3.3.4. Dynamic Correlation Computation

Given a set of learnable kernel matrices $\{\mathbf{W}_l \mid l = 1, \dots, L\}$ and a set of point features $\mathbf{F} = \{\mathbf{F}_j \in \mathbb{R}^C\}$, DCN first calculates point-wise correlation scores via Softmax normalization:

$$\gamma_j = \mathrm{Softmax}(\mathbf{Z}), \tag{5}$$

where $\boldsymbol{\gamma}_j \in \mathbb{R}^L$ stacks the correlation weights $\{\gamma_{jl}\}_{l=1}^{L}$ associated with the jth neighboring point, and all correlation weights form an $N_i \times L$ correlation score matrix as illustrated in Fig. 2.

The resulting correlation scores are subsequently used to aggregate local neighborhood features:

$$(\mathbf{F} * g)(\mathbf{p}_i) = \sum_{\mathbf{p}_j \in \mathcal{N}_i} \sum_{l=1}^{L} \gamma_{jl}\,\mathbf{W}_l\mathbf{F}_j. \tag{6}$$

The resulting output feature is rotation-equivariant. This property follows from two observations: (i) the correlation scores $\boldsymbol{\gamma}_j$ are computed exclusively from rotation-invariant descriptors, and therefore remain unchanged under global rotations; and (ii) the aggregation in Eq. (6) consists of scalar-weighted linear combinations of rotation-equivariant features, which preserves equivariance. As a result, DCN ensures that kernel selection is rotation-invariant, while the aggregated features consistently transform under rotations.

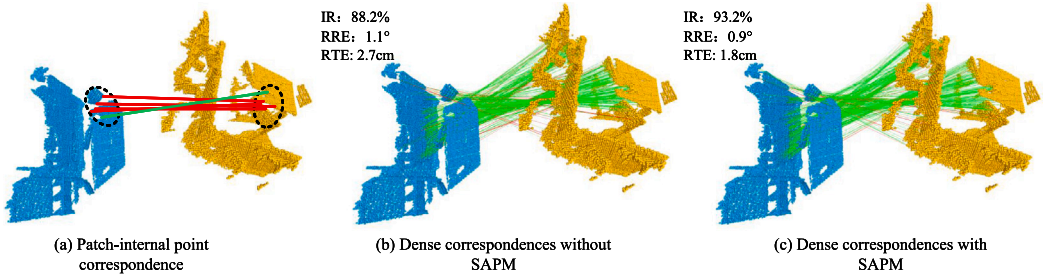

Fig. 3. Visualization of dense correspondences and matching behavior. (a) Illustration of potential mismatches among points within a superpoint pair. (b) Dense correspondences without SAPM, where inconsistent matches frequently occur. (c) Dense correspondences with SAPM, where correspondences become more coherent and stable due to enhanced contextual modeling.

3.4. Structure-Aware Point Matching (SAPM)

3.4.1. Motivation

After coarse-level superpoint matching, each matched superpoint pair $(\hat{\mathbf{p}}_{x_i}, \hat{\mathbf{q}}_{y_i})$ defines two local dense patches, $\mathcal{G}^{\mathcal{P}}_{x_i} \subset \tilde{\mathcal{P}}$ and $\mathcal{G}^{\mathcal{Q}}_{y_i} \subset \tilde{\mathcal{Q}}$, which are formed by neighboring points at the first downsampling level. Although the backbone network provides rotation-invariant point descriptors, establishing reliable correspondences at the patch level remains challenging.

As illustrated in Fig. 3(a), even within a correctly matched superpoint pair, point-wise correspondences may still be erroneous, leading to unstable correspondence estimation.

A widely adopted paradigm constructs a pairwise similarity matrix using point-wise descriptors and enforces matching constraints via optimal transport with Sinkhorn normalization. While effective, these approaches primarily rely on raw feature similarity and iterative normalization, without explicitly modeling contextual relationships within local patches. As a result, they may still produce inconsistent correspondences, as shown in Fig. 3(b).

To address these limitations, we propose the Structure-Aware Point Matching (SAPM) module, which reformulates correspondence estimation through contextual encoding and direct consistency enforcement. Specifically, a linear self-attention operator is employed to capture patch-level contextual interactions, allowing point descriptors to be refined based on their local neighborhoods. In this way, each point is represented not only by its own features, but also by its surrounding contextual information.

Instead of relying on optimal transport, SAPM enforces matching consistency through a row- and column-wise softmax with element-wise fusion, providing a direct and efficient alternative to iterative normalization. Furthermore, a predefined validity mask is incorporated into the softmax operation to filter out invalid points, and per-point confidence weighting is introduced to suppress unreliable correspondences. As shown in Fig. 3(c), incorporating SAPM leads to more accurate and stable correspondences.

3.4.2. Preliminaries

Given a matched superpoint pair $(\hat{\mathbf{p}}_{x_i}, \hat{\mathbf{q}}_{y_i})$, we extract their associated local point patches by selecting the K-nearest neighbor points:

$$\mathcal{G}^{\mathcal{P}}_{x_i} = \{\tilde{\mathbf{p}}_{x_i,1}, \dots, \tilde{\mathbf{p}}_{x_i,K}\} \subset \tilde{\mathcal{P}}, \qquad \mathcal{G}^{\mathcal{Q}}_{y_i} = \{\tilde{\mathbf{q}}_{y_i,1}, \dots, \tilde{\mathbf{q}}_{y_i,K}\} \subset \tilde{\mathcal{Q}},$$

where K is a fixed patch size.

To ensure the locality of each patch, we verify whether each neighbor point actually belongs to the local region represented by its corresponding superpoint. Specifically, we construct binary validity masks $\mathbf{M}^{\mathcal{P}}_{x_i}, \mathbf{M}^{\mathcal{Q}}_{y_i} \in \{0,1\}^K$, where $\mathbf{M}^{\mathcal{P}}_{x_i}[k] = 1$ indicates that point $\tilde{\mathbf{p}}_{x_i,k}$ is closer to superpoint $\hat{\mathbf{p}}_{x_i}$ than to any other superpoint, and thus belongs to its local patch; otherwise, it is considered an outlier from adjacent regions.

Let $n_i = \sum_{k=1}^{K} \mathbf{M}^{\mathcal{P}}_{x_i}[k]$ and $m_i = \sum_{k=1}^{K} \mathbf{M}^{\mathcal{Q}}_{y_i}[k]$ denote the number of valid points in each patch. The feature matrices of the patches are:

$$\mathbf{F}^{\mathcal{P}}_{x_i} \in \mathbb{R}^{K\times C}, \quad \mathbf{F}^{\mathcal{Q}}_{y_i} \in \mathbb{R}^{K\times C}.$$

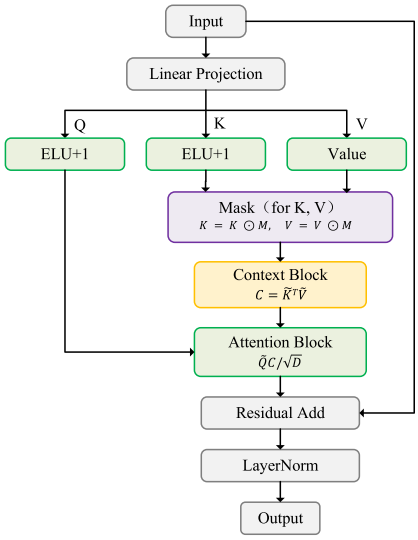

Fig. 4. Architecture of the proposed linear self-attention module. Query and key features are transformed by the ELU-based feature map $\Phi(\cdot) = \mathrm{ELU}(\cdot) + 1$, while validity masks are applied to the key and value branches. The context matrix is computed as $\mathbf{C} = \tilde{\mathbf{K}}^{\top}\tilde{\mathbf{V}}$, followed by attention aggregation $\tilde{\mathbf{Q}}\mathbf{C}/\sqrt{d}$, residual addition, and LayerNorm.

The objective of SAPM is to compute a dense matching confidence matrix $\mathbf{P}_i \in \mathbb{R}^{K\times K}$ that yields reliable point-level correspondences between the valid points of $\mathcal{G}^{\mathcal{P}}_{x_i}$ and $\mathcal{G}^{\mathcal{Q}}_{y_i}$.

3.4.3. Linear self-attention for structure-aware feature enhancement

To encode patch context with linear complexity, we apply an ELU-kernelized linear self-attention operator independently to each patch Fig. 4 shows the linear self-attention. For the ith superpoint correspondence, we process the patches $\mathcal{G}^{\mathcal{P}}_{x_i}$ and $\mathcal{G}^{\mathcal{Q}}_{y_i}$ separately.

We first project the input features into query, key, and value spaces:

$$\mathbf{Q}^{\mathcal{P}}_i = \mathbf{F}^{\mathcal{P}}_{x_i}\mathbf{W}_Q^{\top}, \quad \mathbf{K}^{\mathcal{P}}_i = \mathbf{F}^{\mathcal{P}}_{x_i}\mathbf{W}_K^{\top}, \quad \mathbf{V}^{\mathcal{P}}_i = \mathbf{F}^{\mathcal{P}}_{x_i}\mathbf{W}_V^{\top}, \tag{7}$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d\times C}$ are learnable weight matrices. We employ the ELU activation as a feature map to ensure non-negativity and numerical stability:

$$\Phi(\mathbf{X}) = \mathrm{ELU}(\mathbf{X}) + 1. \tag{8}$$

To exclude invalid points from context aggregation, we multiply the key and value matrices by the validity mask $\mathbf{M}^{\mathcal{P}}_{x_i}$, broadcasting along the feature dimension:

$$\tilde{\mathbf{K}}^{\mathcal{P}}_i = \Phi(\mathbf{K}^{\mathcal{P}}_i) \odot \mathbf{M}^{\mathcal{P}}_{x_i}, \quad \tilde{\mathbf{V}}^{\mathcal{P}}_i = \mathbf{V}^{\mathcal{P}}_i \odot \mathbf{M}^{\mathcal{P}}_{x_i}. \tag{9}$$

Similarly, we compute $\tilde{\mathbf{Q}}^{\mathcal{P}}_i = \Phi(\mathbf{Q}^{\mathcal{P}}_i)$ without masking.

The linear self-attention aggregates contextual information via kernel trick with linear complexity:

$$\mathbf{C}_i^{\mathcal{P}} = (\widetilde{\mathbf{K}}_i^{\mathcal{P}})^{\top}\widetilde{\mathbf{V}}_i^{\mathcal{P}} \in \mathbb{R}^{d\times d},$$
$$\widetilde{\mathbf{F}}_{x_i}^{\mathcal{P}} = \mathrm{LN}\left(\mathbf{F}_{x_i}^{\mathcal{P}} + \frac{1}{\sqrt{d}}\widetilde{\mathbf{Q}}_i^{\mathcal{P}}\mathbf{C}_i^{\mathcal{P}}\right). \tag{10}$$

where $\mathrm{LN}(\cdot)$ denotes channel-wise layer normalization. The matrix product $\mathbf{C}_i^{\mathcal{P}}$ is computed once per patch with $\mathcal{O}(Kd^2)$ complexity, avoiding the quadratic $\mathcal{O}(K^2)$ cost of standard attention. The same operation is applied to $\mathbf{F}_{y_i}^{\mathcal{Q}}$ to obtain $\widetilde{\mathbf{F}}_{y_i}^{\mathcal{Q}}$.

The input features $\mathbf{F}_{x_i}^{\mathcal{P}}$ and $\mathbf{F}_{y_i}^{\mathcal{Q}}$ are rotation-invariant due to the design of the backbone network. All subsequent operations including linear projections, the Φ mapping, matrix multiplication, layer normalization, and residual addition operate solely on these invariant descriptors. Consequently, the enhanced features $\widetilde{\mathbf{F}}_{x_i}^{\mathcal{P}}$ and $\widetilde{\mathbf{F}}_{y_i}^{\mathcal{Q}}$ preserve rotation invariance.

3.4.4. Shared projection and cross-patch similarity

We project the enhanced patch features into a shared metric space using a learnable projection matrix $\mathbf{W}_P \in \mathbb{R}^{d\times C}$:

$$\mathbf{H}_{x_i}^{\mathcal{P}} = \widetilde{\mathbf{F}}_{x_i}^{\mathcal{P}}\mathbf{W}_P^{\top} \in \mathbb{R}^{n_i\times d}, \qquad \mathbf{H}_{y_i}^{\mathcal{Q}} = \widetilde{\mathbf{F}}_{y_i}^{\mathcal{Q}}\mathbf{W}_P^{\top} \in \mathbb{R}^{m_i\times d}.$$

The similarity between points in the two patches is computed via scaled dot-product:

$$\mathbf{S}_i = \frac{\mathbf{H}_{x_i}^{\mathcal{P}}\left(\mathbf{H}_{y_i}^{\mathcal{Q}}\right)^{\top}}{\sqrt{d}} \in \mathbb{R}^{n_i\times m_i}. \tag{11}$$

3.4.5. Row–column masked softmax matching

To enforce mutual consistency and handle visibility variations, we perform row–column masked softmax normalization. Let $\mathcal{M}_{x_i}^{\mathcal{P}} = \mathrm{diag}(\mathbf{M}_{x_i}^{\mathcal{P}})$ and $\mathcal{M}_{y_i}^{\mathcal{Q}} = \mathrm{diag}(\mathbf{M}_{y_i}^{\mathcal{Q}})$ be diagonal matrices derived from the visibility masks. The bidirectional softmax probabilities are computed as:

$$\widetilde{\mathbf{S}}_i^{(r)} = \mathrm{Softmax}_{\mathrm{row}}\left(\mathbf{S}_i + \log\mathcal{M}_{y_i}^{\mathcal{Q}}\right),$$
$$\widetilde{\mathbf{S}}_i^{(c)} = \mathrm{Softmax}_{\mathrm{col}}\left(\mathbf{S}_i + \log\mathcal{M}_{x_i}^{\mathcal{P}}\right). \tag{12}$$

where the log-mask assigns $-\infty$ to padded entries, effectively excluding them from normalization.

We fuse the bidirectional probabilities and incorporate per-point confidence scores $\mathbf{s}_{x_i}^{\mathcal{P}}$ and $\mathbf{s}_{y_i}^{\mathcal{Q}}$ to obtain the final correspondence matrix:

$$\mathbf{P}_i = \widetilde{\mathbf{S}}_i^{(r)} \odot \left(\widetilde{\mathbf{S}}_i^{(c)}\right)^{\top} \odot \left(\mathbf{s}_{x_i}^{\mathcal{P}}\left(\mathbf{s}_{y_i}^{\mathcal{Q}}\right)^{\top}\right). \tag{13}$$

where $\odot$ denotes element-wise multiplication, and the confidence scores are broadcast via outer product.

Dense correspondences are extracted by selecting high-confidence entries from $\mathbf{P}_i$. We employ mutual top-k selection, where a point pair $(\tilde{\mathbf{p}}_{x_i,a}, \tilde{\mathbf{q}}_{y_i,b})$ is selected if (a, b) is among the k largest entries in both its row and column of $\mathbf{P}_i$:

$$C_i = \left\{\left(\tilde{\mathbf{p}}_{x_i,a}, \tilde{\mathbf{q}}_{y_i,b}\right) \mid (a,b) \in \mathrm{mutual\_topk}\left(\mathbf{P}_i\right)\right\}. \tag{14}$$

The union of all patch-level correspondences forms the global dense correspondence set: $\mathcal{C} = \bigcup_{i=1}^{N_c} C_i$, where N_c is the number of matched superpoint pairs.

Unlike conventional optimal transport (OT)-based correspondence estimation methods [24,27], SAPM incorporates a local contextual refinement stage prior to correspondence computation. Specifically, linear self-attention is employed within each local patch to enhance interactions among neighboring point features. In addition, bidirectional matching consistency is enforced through a row–column masked dual-softmax formulation, which avoids iterative Sinkhorn normalization and naturally handles invalid points through masking operations.

The role of attention in SAPM also differs from that of transformer-based correspondence modeling approaches [23,25,33,39]. Rather than using cross-attention to directly establish inter-cloud correspondences, SAPM applies lightweight self-attention only within local patches for feature refinement. Correspondence estimation is subsequently performed through the dual-softmax matching process. Furthermore, coarse-level superpoint matching confidence is explicitly propagated to point-level correspondence estimation, strengthening the interaction between coarse and fine matching stages.

The key design rationale of SAPM is to shift fine-level matching from purely pairwise descriptor comparison to context-aware local patch matching. Unlike OT-based methods that regularize a pre-computed similarity matrix, SAPM first refines point descriptors using intra-patch structural context and then applies deterministic bidirectional consistency. Therefore, SAPM changes where structural information is introduced in the matching pipeline, rather than merely replacing Sinkhorn normalization with another normalization strategy.

Therefore, the contribution of SAPM lies not in introducing a new attention operator or a new matching formulation individually, but in the unified integration of local contextual refinement, deterministic bidirectional matching, and coarse-to-fine confidence propagation within a lightweight correspondence estimation framework.

3.5. Loss function

To train the proposed network in a stable and effective manner, we employ three complementary loss terms, $\mathcal{L} = \mathcal{L}_c + \mathcal{L}_f + \mathcal{L}_r$, which supervise the model from the aspects of superpoint alignment, point-level correspondence, and rotation modeling.

Point matching loss $\mathcal{L}_f$

Since fine-level matching is performed only within superpoint pairs, inaccurate coarse associations may result in insufficient supervisory signals. To alleviate this issue during training, we generate positive superpoint correspondences using the ground-truth transformation. For each paired region $\mathcal{G}_{x_i}^{\mathcal{P}}$ and $\mathcal{G}_{y_i}^{\mathcal{Q}}$, we determine the set of correct point correspondences C_i^* whose spatial discrepancy is below a predefined radius d_p. Points that do not match any partner are collected into two sets, $\mathcal{I}_i$ and $\mathcal{J}_i$.

Given the soft assignment matrix $\mathcal{Z}_i$ and the predicted saliency scores $\sigma_{x_i}^{\mathcal{P}}$ and $\sigma_{y_i}^{\mathcal{Q}}$,the loss at the ith superpoint pair is defined as:

$$\mathcal{L}_{f_i} = -\frac{1}{|C_i^*|}\sum_{(x_i,y_i)\in C_i^*}\log\mathcal{Z}_{x_i,y_i} - \frac{1}{2|\mathcal{I}_i|}\sum_{x_i\in\mathcal{I}_i}\log(1-\sigma_{x_i}^{\mathcal{P}}) - \frac{1}{2|\mathcal{J}_i|}\sum_{y_i\in\mathcal{J}_i}\log(1-\sigma_{y_i}^{\mathcal{Q}}). \tag{15}$$

The full point matching loss is obtained by summing over all patches: $\mathcal{L}_f = \sum_i \mathcal{L}_{f_i}$.

Contrastive rotation loss $\mathcal{L}_r$

Although rotation-related geometric information is encoded in equivariant features, such representations may still degrade when data are affected by missing regions, noise, or aggressive downsampling. To reinforce robustness, we formulate a contrastive loss that explicitly encourages consistent equivariant features for valid matches while pushing apart features of mismatched point pairs.

For each positive pair in C_i^* and each channel c, we enforce proximity after rotating the feature according to the ground-truth rotation matrix $\mathbf{R}_{gt}$. Similarly, negative pairs$\tilde{C}_i$(with spatial distance exceeding d_n) are required to maintain a sufficiently large separation.

$$\mathcal{L}_{r_i} = \frac{1}{|C_i^*|\,\tilde{d}}\sum_{(x_i,y_i)\in C_i^*}\sum_{c=1}^{\tilde{d}}\left[\|\tilde{\mathbf{F}}_{x_i,c}^{\mathcal{P}}\mathbf{R}_{gt}^{\mathrm{T}} - \tilde{\mathbf{F}}_{y_i,c}^{\mathcal{Q}}\|_2^2 - \alpha\right]_+ + \frac{1}{|\tilde{C}_i|\,\tilde{d}}\sum_{(x_i,y_i)\in\tilde{C}_i}\sum_{c=1}^{\tilde{d}}\left[\beta - \|\tilde{\mathbf{F}}_{x_i,c}^{\mathcal{P}}\mathbf{R}_{gt}^{\mathrm{T}} - \tilde{\mathbf{F}}_{y_i,c}^{\mathcal{Q}}\|_2^2\right]_+. \tag{16}$$

where $\alpha = 0.1$ and $\beta = 1.4$ are margins for positive and negative constraints respectively, and $[\cdot]_+$ denotes the ReLU operator.

Overlap-aware circle loss $\mathcal{L}_c$

To supervise superpoint descriptors, we extend the classical triplet-based circle loss by incorporating overlap awareness. Superpoint pairs with more than 10% overlap are regarded as positives, while those below this ratio are treated as negatives. The overlap ratio influences the weight assigned to each term, enabling the learned representation to better distinguish spatially coherent regions.

For a superpoint $\hat{p}_i$ from cloud $\hat{P}$, the loss is expressed as:

$$\mathcal{L}_c^P = \frac{1}{|\mathcal{A}|} \sum_{\hat{p}_i \in \mathcal{A}} \log \left[1 + \sum_{\hat{q}_j \in \xi_p^i} e^{\lambda_i^j \beta_p^{i,j} (d_i^j - \Delta_p)} \cdot \sum_{\hat{q}_k \in \xi_n^i} e^{\beta_n^{i,k} (\Delta_n - d_i^k)} \right]. \tag{17}$$

with $d_i^j = \|\hat{p}_i - \hat{q}_j\|_2$,positive set ξ_p^i, negative set ξ_n^i, and overlap factor $\lambda_i^j = (o_i^j)^{1/2}$. The weighting coefficients follow$\beta_p^{i,j} = \gamma(d_i^j - \Delta_p)$ and $\beta_n^{i,k} = \gamma(\Delta_n - d_i^k)$,where $\Delta_p = 0.1$ and $\Delta_n = 1.4$.

A symmetric loss is computed for $\hat{Q}$, and the final overlap-aware loss is obtained by averaging:

$$\mathcal{L}_c = (\mathcal{L}_c^P + \mathcal{L}_c^Q)/2. \tag{18}$$

4. Experiments

4.1. Implementation details

All experiments are implemented in PyTorch and conducted on a workstation equipped with an Intel Xeon Gold 5218 CPU and an NVIDIA RTX 4090 GPU. Our network is trained separately on the 3DMatch and KITTI datasets following the standard protocols. The Adam optimizer is adopted with an initial learning rate of 1×10^{-4} and a weight decay of 1×10^{-6}. The learning rate is decayed by a factor of 0.95 every epoch on 3DMatch and every four epochs on KITTI. We train the model for 40 epochs on 3DMatch and 100 epochs on KITTI, with a batch size of 1 for both datasets.

During training, we apply the same data augmentation strategies as widely adopted in prior works, including random rotations in $[0, 2\pi]$, Gaussian noise injection, random cropping, and dataset-specific scaling or translation (see Table 1 for full details). Point clouds are voxelized with a voxel size of 0.025 m on 3DMatch and 0.3 m on KITTI.

For neighborhood construction, we use 35 nearest neighbors for all convolution layers. The number of dynamic kernel matrices in the correlation network is set to 4. Following a coarse-to-fine paradigm, we sample $N_c = 256$ coarse superpoint correspondences for coarse matching and retain $N_f = 1000$ fine correspondences for transformation estimation. The acceptance radii for correspondence supervision are set to $\tau_d = 0.1$ m for 3DMatch and 0.6 m for KITTI.

A summary of all hyperparameters and dataset-specific configurations is provided in Table 1.

4.2. Performance on outdoor dataset

Dataset. The KITTI dataset [40] is a widely used large-scale outdoor benchmark for evaluating point cloud registration systems in autonomous driving scenarios. Following the official protocol, we adopt sequences 0–5 for training, sequence 7 for validation, and sequences 8–10 for testing. Due to inaccuracies introduced by GPS-based pose annotations, we follow common practice and refine the provided ground-truth poses using an additional ICP alignment [41] to obtain more reliable reference transformations.

Evaluation Metrics. Consistent with standard practice in prior works [24], three quantitative criteria are used to assess registration quality: relative rotation error (RRE), relative translation error (RTE), and registration recall (RR). Formal definitions can be found in [31].

The relative rotation error computes the angular discrepancy between the estimated rotation matrix and the reference one:

$$\text{RRE} = \arccos\left(\frac{\text{trace}(\mathbf{R}^\top \bar{\mathbf{R}}) - 1}{2}\right). \tag{19}$$

Table 1
Detailed configurations of our method.

Configuration category	3DMatch	KITTI
Training settings		
Batch Size	1	1
Initial Learning Rate	10^{-4}	10^{-4}
Epoch	40	100
Weight Decay	10^{-6}	10^{-6}
Learning Rate Decay	0.95	0.95
Decay Step	1	4
Data augmentation		
Voxel Size	0.025 m	0.3 m
Gaussian Noise	0.005 m	0.01 m
Rotation Range	2π	2π
Scale Range	None	[0.8, 1.1]
Translation Range	None	2 m
Crop Ratio	0.3	0.3
Network parameters		
Number of Nearest Neighbors	35	35
Number of Weight Matrices	4	4
Number of Coarse Correspondences N_c	256	256
Number of Fine Correspondences N_f	1K	1K
Acceptance Radius τ_d	0.1 m	0.6 m

Table 2
Evaluation results on KITTI. Best and second-best results are highlighted in bold and underlined, respectively.

Model	Publication	RTE (cm)	RRE (°)	RR (%)
3DFeat-Net	ECCV 2018 [44]	25.9	0.25	96.0
FCGF	ICCV 2019 [31]	9.5	0.30	96.6
D3Feat	CVPR 2020 [42]	7.2	0.30	99.8
SpinNet	CVPR 2021 [32]	9.9	0.47	99.1
Predator	CVPR 2021 [33]	6.8	0.27	99.8
CoFiNet	NeurIPS 2021 [23]	8.2	0.41	99.8
GeoTransformer	CVPR 2022 [24]	6.8	0.24	99.8
BUFFER	CVPR 2023 [45]	7.1	0.26	99.8
PEAL	CVPR 2023 [26]	6.8	0.23	99.8
DCATr	CVPR 2024 [46]	6.6	<u>0.22</u>	99.7
PARENet	ECCV 2024 [36]	5.4	0.24	99.8
CAST	NeurIPS 2024 [39]	**2.5**	0.27	**100.0**
PTT	IEEE TCSVT 2025 [47]	6.3	0.23	99.8
UGP	CVPR 2025 [48]	7.1	0.24	99.8
Ours	–	<u>4.8</u>	**0.21**	99.8

The relative translation error measures the Euclidean distance between predicted translation and its ground-truth counterpart:

$$\text{RTE} = \|\mathbf{t} - \bar{\mathbf{t}}\|_2. \tag{20}$$

On KITTI, registration recall is defined as the proportion of point cloud pairs whose estimated transformation falls within pre-defined accuracy thresholds, specifically RRE < 5° and RTE < 2 m:

$$\text{RR} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}\left[\text{RRE}_i < 5^\circ \ \wedge \ \text{RTE}_i < 2 \text{ m}\right]. \tag{21}$$

Following conventional evaluation protocols [23,31,33,42,43], the mean RRE and mean RTE are reported only over the subset of successfully aligned pairs, i.e., those counted as correct registrations under KITTI's recall criterion.

Results: We evaluate our method on the KITTI odometry benchmark and compare it with recent learning-based registration approaches, including 3DFeat-Net [44], FCGF [31], D3Feat [42], SpinNet [32], Predator [33], CoFiNet [23], GeoTransformer [24], BUFFER [45], PEAL [26], DCATr [46], PARENet [36], CAST [39], PPT [47], and UGP [48] . The quantitative results are reported in Table 2.

Our method achieves competitive and well-balanced performance across all metrics, with an RTE of **4.8** cm, an RRE of **0.21**°, and an RR

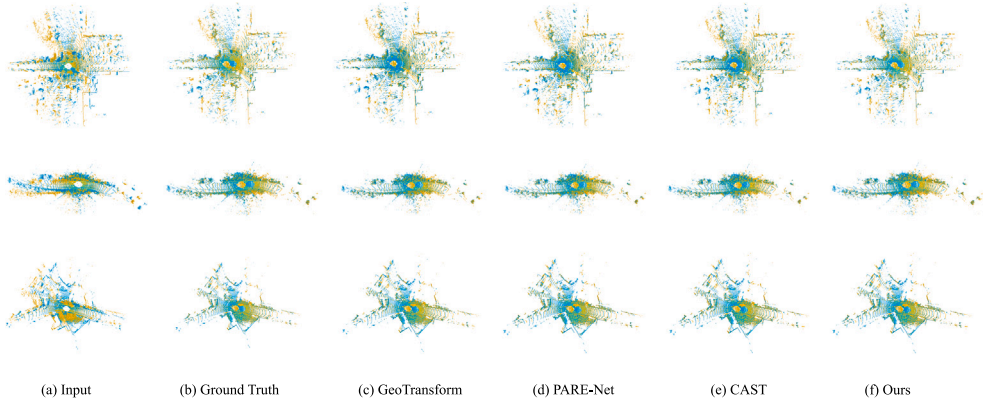

Fig. 5. Visualization results comparison of point cloud registration by various methods on the KITTI odometry dataset.

of 99.8%. In particular, our method attains the lowest rotation error among all compared approaches.

We note that RTE and RRE reflect different aspects of registration quality, and improvements in one metric do not necessarily imply improvements in the other. For example, methods such as CAST achieve lower RTE by emphasizing global consistency, while rotation estimation is more sensitive to rotational variations.

In contrast, our method explicitly promotes consistent feature correlation modeling under varying rotations. The proposed Dynamic Correlation Network constructs correlation weights in a rotation-invariant feature space, reducing inconsistencies caused by orientation changes. In addition, the Structure-Aware Point Matching module stabilizes correspondence estimation by enforcing consistency in local feature interactions.

As a result, our method achieves more stable rotation estimation, as reflected by the lowest RRE, while maintaining competitive translation accuracy. Qualitative results are shown in Fig. 5.

4.3. Performance on indoor dataset

Dataset. The 3DMatch benchmark [49] is a widely used indoor dataset for evaluating geometric registration approaches and consists of 62 reconstructed indoor environments. Following the standard data division adopted in [24,49], we employ 46 scenes for training, 8 for validation, and the remaining 8 for testing. All methods are assessed under two commonly used evaluation protocols: 3DMatch and 3DLoMatch. The former contains point cloud pairs with more than 30% spatial overlap, whereas the latter focuses on more challenging cases in which the shared region between two scans lies within the 10%–30% interval.

Evaluation Metrics. To measure registration accuracy, we report the Registration Recall (RR), defined as the percentage of scan pairs for which the estimated rigid motion falls within the prescribed tolerance. A pair is considered successfully aligned when the root mean squared error (RMSE) between transformed source points and their ground-truth counterparts is below 0.2 m. Given a set of ground-truth correspondences $C^* = \{(\mathbf{p}^*_{x_i}, \mathbf{q}^*_{y_i})\}$, the RMSE is calculated as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|C^*|} \sum_{(\mathbf{p}^*_{x_i}, \mathbf{q}^*_{y_i}) \in C^*} \left\| \mathbf{T}_{\mathbf{P}\to\mathbf{Q}}(\mathbf{p}^*_{x_i}) - \mathbf{q}^*_{y_i} \right\|_2^2}. \tag{22}$$

Registration Recall is then defined as:

$$\mathrm{RR} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}\left[\mathrm{RMSE}_i < 0.2 \text{ m}\right], \tag{23}$$

where M denotes the total number of evaluation pairs and $\mathbb{I}[\cdot]$ is the indicator function.

To validate the robustness of various algorithms against arbitrary rotational perturbations, we follow the evaluation protocol established in prior works [35] and adopt Transformation Recall (TR) as the quantitative metric for registration performance. Specifically, TR is defined as the proportion of successfully registered point cloud pairs across the entire test set, where a pair is considered valid only if both its Relative Rotation Error (RRE) and Relative Translation Error (RTE) fall below the pre-specified thresholds:

$$\mathrm{TR} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}\left[\mathrm{RRE}_i < 15^\circ \ \wedge \ \mathrm{RTE}_i < 0.3 \text{ m}\right]. \tag{24}$$

Table 3
Evaluation results on 3DMatch and 3DLoMatch. Best and second-best results are highlighted in bold and underlined, respectively.

Model	Estimator	Samples	RR(%)	
			3DMatch	3DLoMatch
FCGF [31]	RANSAC	5000	85.1	40.1
D3Feat [42]	RANSAC	5000	81.6	37.2
SpinNet [32]	RANSAC	5000	88.6	59.8
Predator [33]	RANSAC	5000	89.0	59.8
CoFiNet [23]	RANSAC	5000	89.3	67.5
RIGA [50]	RANSAC	5000	89.3	65.1
RoITr [25]	RANSAC	5000	91.9	74.7
RoReg [35]	RANSAC	5000	92.9	70.3
BUFFER [45]	RANSAC	5000	92.9	71.8
CAST [39]	RANSAC	–	<u>95.2</u>	<u>75.1</u>
Cross-PCR [51]	RANSAC	5000	94.5	73.7
BUFFER-X [52]	RANSAC	5000	**95.6**	74.2
Ours	RANSAC	5000	<u>95.2</u>	**79.3**
GeoTrans [24]	LGR	all	91.5	74.0
PEAL [26]	LGR	all	94.2	78.8
DCATr [46]	LGR	all	92.1	75.7
DFAT [27]	LGR	all	94.9	76.8
Ours	LGR	all	**95.0**	<u>78.9</u>
PARENet [36]	FHP	all	94.9	79.3
Ours	FHP	all	**95.4**	**79.6**

Results:

We further evaluate our method on the 3DMatch and 3DLoMatch benchmarks, which correspond to indoor registration scenarios with overlap ratios greater than 30% and between 10% ∼ 30%, respectively.

To ensure a fair comparison, we categorize the evaluated methods into two groups: (1) methods using RANSAC for pose estimation [23,25,31–33,35,39,42,45,50–52]. (2) methods adopting learning-based estimators, including Local-to-Global Registration (LGR) [24,26,27,46] and Feature-based Hypothesis Proposer (FHP) [36]. The Registration Recall (RR) results are reported in Table 3.

In the standard 3DMatch benchmark (overlap > 30%), our method achieves an RR of **95.4%**, showing competitive performance among both RANSAC-based and learning-based approaches. It performs comparably to the best RANSAC-based method BUFFER-X (95.6%) and

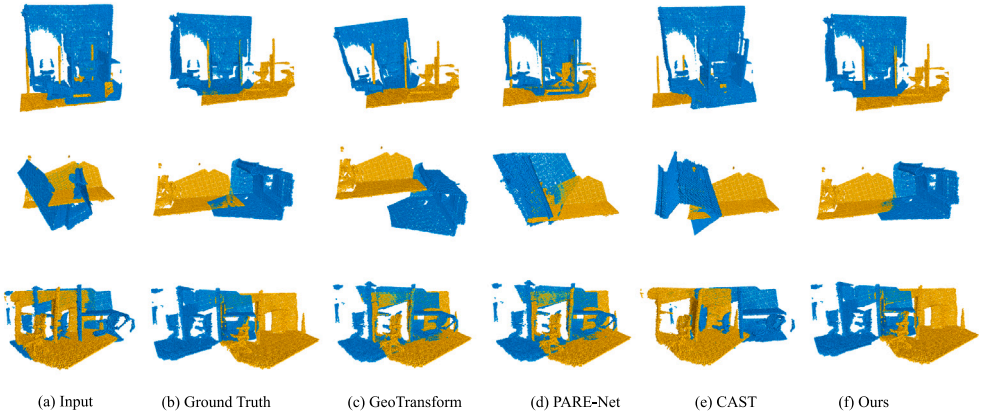

Fig. 6. Visualization results comparison of point cloud registration by various methods on the 3DMatch and 3DLoMatch datasets.

Table 4
Quantitative comparison on 3DLoMatch and its rotated counterpart. The change in TR after applying random rotations is reported as a superscript for each entry, enabling a direct comparison of sensitivity to rotational perturbations. Methods incorporating rotation-invariant or equivariant designs are indicated by a $*$ symbol.

Method	3DLoMatch			Rotated 3DLoMatch		
	RRE (° ↓)	RTE (cm ↓)	TR (% ↑)	RRE (° ↓)	RTE (cm ↓)	TR (% ↑)
FCGF [31]	4.84	12.87	39.6	4.74	13.39	$24.5^{-15.1}$
PREDATOR [33]	3.61	10.65	65.6	3.55	10.30	$64.0^{-1.6}$
GeoTrans [24]	2.91	8.71	75.4	2.94	8.85	$72.6^{-2.8}$
PEAL [26]	2.84	**8.64**	81.2	2.86	**8.53**	$78.7^{-2.5}$
YOHO* [34]	3.54	10.34	66.6	3.61	10.16	$67.1^{+0.5}$
RoITR* [25]	2.95	9.03	75.1	2.97	9.08	$75.5^{+0.4}$
PARENet* [36]	2.87	8.83	81.3	2.84	8.71	$81.8^{+0.5}$
Ours*	**2.78**	8.65	**82.3**	**2.75**	8.53	**82.7** $^{+0.4}$

slightly improves over recent learning-based estimators such as DFAT (94.9%) and PARENet (94.9%).

In the more challenging 3DLoMatch benchmark (overlap 10%–30%), our method achieves an RR of **79.6%**, which ranks the best among all compared methods. Compared to representative approaches such as DFAT (76.8%) and PEAL (78.8%), our method shows consistent improvements under low-overlap conditions.

We also observe that while methods such as BUFFER-X achieve strong performance on 3DMatch, their performance degrades on 3DLoMatch, indicating sensitivity to reduced overlap. In contrast, our method maintains stable performance across both benchmarks.

This robustness can be attributed to the proposed Dynamic Correlation Network and Structure-Aware Point Matching, which enforce consistent feature correlation modeling across varying rotations and alleviate correspondence instability caused by correlation inconsistency, particularly in low-overlap scenarios. Qualitative results are shown in Fig. 6.

To further evaluate robustness under unseen orientations, we conduct additional experiments on the rotated 3DLoMatch benchmark, where arbitrary rotations are applied to the input point clouds. This setting introduces significant challenges for correspondence estimation.

The quantitative results are summarized in Table 4. Compared with the standard 3DLoMatch setting, methods that rely primarily on rotation augmentation, such as FCGF and Predator, suffer noticeable performance degradation, indicating limited generalization to unseen rotations. In contrast, methods with rotation-invariant or equivariant designs (e.g., YOHO, RoITR, and PARENet) exhibit more stable performance.

Our method achieves the best overall performance on both datasets, with a Transformation Recall (TR) of **82.3%** on 3DLoMatch and **82.7%** on Rotated 3DLoMatch. Notably, the performance variation under rotation is minimal, demonstrating strong robustness to arbitrary orientations.

Table 5
Computational efficiency comparison of different models.

Model	Params (M)	FLOPs (G)	Time (ms)
GeoTrans	9.83	140.71	70.40
CAST	8.55	48.74	129.80
PARE	3.84	59.49	122.80
Ours	4.04	72.86	100.29

Note: All timings were measured on a single NVIDIA RTX 4090 GPU.

This improvement can be attributed to the proposed Dynamic Correlation Network, which stabilizes feature correlation modeling in a rotation-invariant feature space, and the Structure-Aware Point Matching module, which further enhances matching reliability through contextual interaction and consistency enforcement. As a result, the proposed method maintains stable correspondence estimation even when the relative rotations are significantly different from those seen during training.

To evaluate computational efficiency, we compare the model complexity and inference time of different methods, as shown in Table 5. Our model contains 4.04M parameters and requires 72.86 GFLOPs, which is significantly lower than GeoTrans while remaining comparable to lightweight architectures. Moreover, our method achieves an inference time of 100.29 ms, outperforming CAST and PARE in runtime efficiency. These results indicate that the proposed SAPM with linear self-attention provides a favorable balance between efficiency and performance.

4.4. Ablation study

We conduct ablation experiments on the KITTI dataset to evaluate the contributions of the Dynamic Correlation Network (DCN) and the

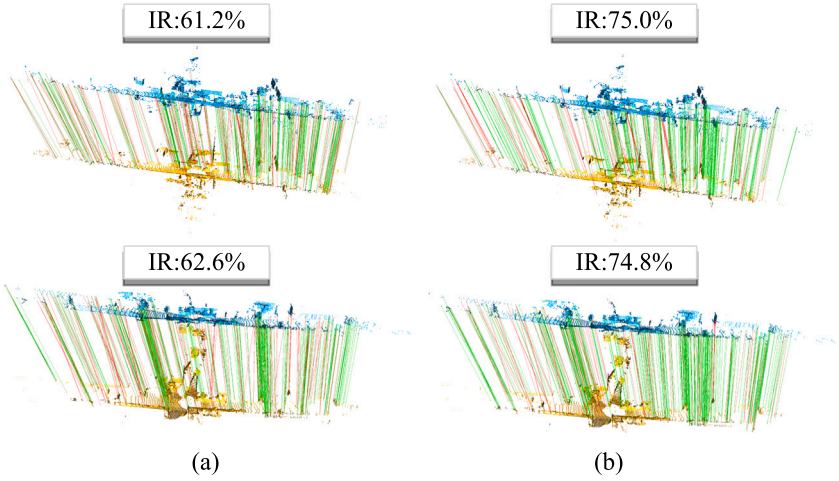

Fig. 7. Point-level correspondences of the models with (a) PARE-Net point matching and (b) structure-aware point matching.

Table 6
Ablation study on the KITTI dataset.

ID	DCN	SAPM	RRE (°)	RTE (cm)
I	✗	✗	0.239	5.5
II	✓	✗	0.227	5.2
III	✗	✓	0.221	5.1
IV	✓	✓	**0.211**	**4.8**

Table 7
Comparative ablation of feature extraction modules on KITTI.

Method	RTE (cm)	RRE (°)	RR (%)
KPConv	6.5	0.24	99.6
VN-Conv	5.6	0.23	99.6
PAREConv	5.3	0.23	**99.8**
Ours	**4.9**	**0.22**	**99.8**

Structure-Aware Point Matching (SAPM). As shown in Table 6, introducing DCN alone (II) reduces the errors compared to the baseline (I), indicating that modeling feature correlations in a rotation-invariant space improves the stability of feature representations under varying orientations.

Similarly, incorporating SAPM alone (III) also improves performance, demonstrating that enforcing consistency in local feature interactions benefits correspondence estimation.

When both modules are combined (IV), the model achieves the best performance, with an RRE of **0.211°** and an RTE of **4.8 cm**. This result indicates that DCN and SAPM are complementary: DCN stabilizes feature correlation modeling under rotation, while SAPM further alleviates correspondence instability. Together, they improve overall registration accuracy.

We further visualize the matching results in Fig. 7. The proposed SAPM produces more stable correspondences with higher inlier ratios, indicating improved matching reliability.

To analyze the impact of feature representation on overall registration performance, we compare different feature extraction backbones within a unified pipeline. Specifically, KPConv, VN-Conv, and PAREConv are employed as alternative feature extractors, while the remaining components of the framework, including the LGR-based transformation estimator, are kept unchanged to ensure a fair comparison.

As shown in Table 7, our full model achieves the best performance across all metrics. These results indicate that, within a unified registration framework, the discriminative power of feature representations plays a critical role in performance, while our model can further exploit high-quality features to achieve superior registration results.

To further investigate the effectiveness of each component, we conduct fine-grained ablation studies on DCN and SAPM, as shown in Table 8.

For DCN, removing either the Channel Reallocation Network or the Dynamic Correlation Computation degrades performance, indicating that both components are important for modeling consistent feature correlations.

For SAPM, removing either the linear self-attention or the row–column masked softmax leads to reduced accuracy, showing that enforcing consistency in feature interactions is essential for reliable correspondence estimation.

Overall, these results validate that each component contributes to improving correlation consistency and that their combination yields the best performance.

5. Conclusion

This paper addresses the problem of unstable correspondence estimation in point cloud registration caused by inconsistent feature correlations under varying rotations. To this end, we propose a rotation-robust registration framework that explicitly models correlation consistency from both feature extraction and matching perspectives.

Specifically, we introduce a Dynamic Correlation Network (DCN), which performs adaptive correlation modeling in a rotation-invariant feature space, enabling the learning of stable and discriminative feature correlations across different orientations. Building upon these enhanced representations, we further develop a Structure-Aware Point Matching (SAPM) module, which improves the reliability of local correspondences through contextual modeling and consistency enforcement.

Extensive experiments on both indoor and outdoor benchmarks demonstrate that the proposed method achieves competitive overall performance, particularly exhibiting strong robustness under rotations and general low-overlap scenarios. However, we observe that in extreme cases, incorrect coarse superpoint associations may occur when the overlap is critically small. Additionally, highly repetitive geometric structures can produce ambiguous local descriptors, leading to multiple plausible correspondences that may mislead the hypothesis selection stage. Future work will focus on enhancing model robustness under these scenarios, extending dynamic correlation modeling to multi-view registration, and developing lightweight variants for real-time applications.

Table 8
Fine-grained ablation study of components on 3DMatch.

Ablated component	RTE (cm)	RRE (°)	RR (%)	Module time (ms)
Full model	5.2	1.693	95.4	Total: 269.8
Channel Reallocation Network	5.8	1.802	94.6	7.9
Dynamic Correlation Computation	5.9	1.768	94.2	6.2
Linear Self-Attention	6.0	1.741	94.4	11.8
Row–Column Masked Softmax	5.6	1.776	94.8	2.4

Note: The accuracy columns (RTE, RRE, RR) report the results obtained by removing the corresponding component from the full model. The "Module Time" column reports the individual execution time of each specific component, not the total runtime of the ablated model. All timings were measured on a single NVIDIA RTX 4060 GPU.

CRediT authorship contribution statement

Lei Lu: Writing – review & editing, Validation, Supervision, Software, Resources, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Meichen Pan:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation. **Ling Cao:** Writing – review & editing, Supervision, Data curation. **Renlong Qi:** Supervision, Investigation. **Peng Li:** Project administration, Investigation, Funding acquisition, Conceptualization. **Wei Pan:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) (Grant No. 62375078); the Science and Technology Innovation Project of the Chinese Academy of Traditional Chinese Medicine (Grant No. ZN2024A02); the Key Research Project Plan for Higher Education Institutions in Henan Province (Grant No. 24ZX011); the Training Plan for Young Backbone Teachers in Undergraduate Universities in Henan Province (Grant No. 2023GGJS058); the Cultivation Programme for Young Backbone Teachers of Henan University of Technology; the Open Fund of the Institute for Complexity Science (Grant No. CSKFJJ-2024-3); and the Dongguan Key Research & Development Program of China (Grant No. 20241200300122).

Data and code availability

Data will be made available on request; The code for this work is publicly available at: https://github.com/pmc555/DC-SAM.

References

[1] Pan W, Lu X, Gong Y, Tang W, Liu J, He Y, Qiu G. HLO: Half-kernel Laplacian operator for surface smoothing. Computer-Aided Des 2020;121:102807.
[2] Chen S, Wang J, Pan W, Gao S, Wang M, Lu X. Towards uniform point distribution in feature-preserving point cloud filtering. Comput Vis Media 2023;9(2):249–63.
[3] Wang W, Lu X, Shao D, Liu X, Dazeley R, Robles-Kelly A, Pan W. Weighted point cloud normal estimation. In: 2023 IEEE international conference on multimedia and expo. IEEE; 2023, p. 2015–20.
[4] Qi CR, Su H, Mo K, Guibas LJ. PointNet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 652–60.
[5] Qi CR, Yi L, Su H, Guibas LJ. PointNet++: Deep hierarchical feature learning on point sets in a metric space. Adv Neural Inf Process Syst 2017;30:5105–14.
[6] Qian G, Li Y, Peng H, Lai J, Wang L. PointNeXt: Revisiting PointNet++ with improved training and scaling strategies. Adv Neural Inf Process Syst 2022;35:23192–204.
[7] Li Y, Bu R, Sun M, Wu W, Di X, Chen B. PointCNN: Convolution on $\mathcal{X}$-transformed points. Adv Neural Inf Process Syst 2018;31:820–30.
[8] Xu M, Ding R, Zhao H, Shi S, Wang W. PAConv: Position adaptive convolution with dynamic kernel assembling on point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 3173–82.
[9] Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. ACM Trans Graph 2019;38(5):146:1–146:12.
[10] Veličković P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Y. Graph attention networks. 2017, arXiv preprint arXiv:1710.10903.
[11] Zhou H, Feng Y, Fang M, Han X, Wang X. Adaptive graph convolution for point cloud analysis. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 4965–74.
[12] Fung S, Pan W, Liu X, Yearwood J, Dazeley R, Lu X. TopFormer: topology-aware transformer for point cloud registration. In: International conference on computational visual media. Springer Nature Singapore Singapore; 2024, p. 112–28.
[13] Fung S, Lu X, Edirimuni Dd, Pan W, Liu X, Li H. Semreg: Semantics constrained point cloud registration. In: European conference on computer vision. Springer Nature Switzerland Cham; 2024, p. 293–310.
[14] Te G, Hu W, Zheng A, Hu Y, Wang M. RGCNN: Regularized graph CNN for point cloud segmentation. In: Proceedings of the 26th ACM international conference on multimedia. 2018, p. 746–54.
[15] Wang L, Huang Y, Hou Y, Liu X. Graph attention convolution for point cloud semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 10296–305.
[16] Zhang N, Pan Z, Li TH, Liu Q, Wang S. Improving graph representation for point cloud segmentation via attentive filtering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 1244–54.
[17] Zhao H, Jiang L, Jia J, Koltun V, Li H. Point transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 16259–68.
[18] Guo MH, Cai JX, Liu ZN, Wu YJ, Huang JB. PCT: Point cloud transformer. Comput Vis Media 2021;7(2):187–99.
[19] Lai X, Liu J, Jiang L, Lu J, Guo Y, Zhang H. Stratified transformer for 3D point cloud segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 8500–9.
[20] Yang YQ, Guo YX, Xiong JY, Li H, Zhao M, Wang S. Swin3D: A pretrained transformer backbone for 3D indoor scene understanding. 2023, arXiv preprint arXiv:2304.06906.
[21] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Adv Neural Inf Process Syst 2017;30:5998–6008.
[22] Zhao H, Jia J, Koltun V. Exploring self-attention for image recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 10076–85.
[23] Yu H, Li F, Saleh M, Busam B, Ilic S. Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. Adv Neural Inf Process Syst 2021;34:23872–84.
[24] Qin Z, Yu H, Wang C, Guo Y, Peng Y, Xu K. Geometric transformer for fast and robust point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 11143–52.
[25] Yu H, Qin Z, Hou J, Saleh M, Li D, Busam B, Ilic S. Rotation-invariant transformer for point cloud matching. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 5384–93.
[26] Yu J, Ren L, Zhang Y, Zhou W, Lin L, Dai G. Peal: Prior-embedded explicit attention learning for low-overlap point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 17702–11.
[27] Fu K, Yuan M, Wang C, Pang W, Chi J, Wang M, Gao L. Dual focus-attention transformer for robust point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2025, p. 11769–78.

[28] Zhou Y, Tuzel O. VoxelNet: End-to-end learning for point cloud based 3D object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 4490–9.
[29] Yan Y, Mao Y, Li B. SECOND: Sparsely embedded convolutional detection. Sensors 2018;18(10):3337.
[30] Shi S, Guo C, Jiang L, Wang Z, Shi J, Duan H. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 10529–38.
[31] Choy C, Park J, Koltun V. Fully convolutional geometric features. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 8958–66.
[32] Ao S, Hu Q, Yang B, Markham A, Guo Y. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 11753–62.
[33] Huang S, Gojcic Z, Usvyatsov M, Wieser A, Schindler K. Predator: Registration of 3d point clouds with low overlap. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 4267–76.
[34] Wang H, Liu Y, Dong Z, Wang W. You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In: Proceedings of the 30th ACM international conference on multimedia. 2022, p. 1630–41.
[35] Wang H, Liu Y, Hu Q, Wang B, Chen J, Dong Z, Guo Y, Wang W, Yang B. RoReg: Pairwise point cloud registration with oriented descriptors and local rotations. IEEE Trans Pattern Anal Mach Intell 2023;45(8):10376–93.
[36] Yao R, Du S, Cui W, Tang C, Yang C. Pare-net: Position-aware rotation-equivariant networks for robust point cloud registration. In: European conference on computer vision. Springer; 2024, p. 287–303.
[37] Deng C, Litany O, Duan Y, Poulenard A, Tagliasacchi A, Guibas LJ. Vector neurons: A general framework for so (3)-equivariant networks. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 12200–9.
[38] Li S, Wang Z, Liu Z, Tan C, Lin H, Wu D, Chen Z, Zheng J, Li SZ. MogaNet: Multi-order gated aggregation network. In: International conference on learning representations. 2024, p. 37393–427.
[39] Huang R, Tang Y, Chen J, Li L. A consistency-aware spot-guided transformer for versatile and hierarchical point cloud registration. Adv Neural Inf Process Syst 2024;37:70230–58.
[40] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE conference on computer vision and pattern recognition. IEEE; 2012, p. 3354–61.
[41] Besl PJ, McKay ND. Method for registration of 3-D shapes. In: Sensor fusion IV: control paradigms and data structures, vol. 1611, Spie; 1992, p. 586–606.
[42] Bai X, Luo Z, Zhou L, Fu H, Quan L, Tai C-L. D3feat: Joint learning of dense detection and description of 3d local features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 6359–67.
[43] Lu F, Chen G, Liu Y, Zhang L, Qu S, Liu S, Gu R, Jiang C. HRegNet: A hierarchical network for efficient and accurate outdoor LiDAR point cloud registration. IEEE Trans Pattern Anal Mach Intell 2023;45(10):11884–97.
[44] Yew ZJ, Lee GH. 3Dfeat-net: Weakly supervised local 3d features for point cloud registration. In: Proceedings of the European conference on computer vision. 2018, p. 607–23.
[45] Ao S, Hu Q, Wang H, Xu K, Guo Y. Buffer: Balancing accuracy, efficiency, and generalizability in point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 1255–64.
[46] Chen H, Yan P, Xiang S, Tan Y. Dynamic cues-assisted transformer for robust point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2024, p. 21698–707.
[47] Wang M, Chen G, Yang Y, Yuan L, Yue Y. Point tree transformer for point cloud registration. IEEE Trans Circuits Syst Video Technol 2025;35(7):6756–72.
[48] Zeng Z, Wu Q, Zhang X, Wu LY, An P, Yang J, Wang J, Wang P. Unlocking generalization power in LiDAR point cloud registration. In: Proceedings of the computer vision and pattern recognition conference. 2025, p. 22244–53.
[49] Zeng A, Song S, Nießner M, Fisher M, Xiao J, Funkhouser T. 3Dmatch: Learning local geometric descriptors from rgb-d reconstructions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 1802–11.
[50] Yu H, Hou J, Qin Z, Saleh M, Shugurov I, Wang K, Busam B, Ilic S. Riga: Rotation-invariant and globally-aware descriptors for point cloud registration. IEEE Trans Pattern Anal Mach Intell 2024;46(5):3796–812.
[51] Zhao G, Guo Z, Du Z, Ma H. Cross-PCR: A robust cross-source point cloud registration framework. In: Proceedings of the AAAI conference on artificial intelligence, vol. 39, 2025, p. 10403–11.
[52] Seo M, Lim H, Lee K, Carlone L, Park J. BUFFER-x: Towards zero-shot point cloud registration in diverse scenes. In: Proceedings of the IEEE/CVF international conference on computer vision. 2025, p. 3851–62.