Expert Systems With Applications 293 (2025) 128591

Contents lists available at ScienceDirect

Expert Systems With Applications

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

EfficientPEAL: Efficient prior-embedded attention learning for partially overlapping point cloud registration

Junle Yu[a,b], Wenhui Zhou[c], Zhehao Shen[d], Yongwei Miao[a,*]

[a] School of Information Science and Technology, Hangzhou Normal University, 311121, Hangzhou, China
[b] Ningbo Institute of Materials Technology & Engineering, Chinese Academy of Sciences, 315201, Ningbo, China
[c] School of Computer Science and Technology, Hangzhou Dianzi University, 310018, Hangzhou, China
[d] School of Electronics and Information Engineering, Hangzhou Dianzi University, 310018, Hangzhou, China

ARTICLE INFO

Keywords:
Point cloud registration
Feature matching
Self-attention mechanism
Efficient transformer
Iterative optimization

ABSTRACT

Learning discriminative point-wise features is critical for partially overlapping point cloud registration. In recent years, the integration of a Transformer into point cloud feature representation has demonstrated remarkable success, which typically involves a self-attention module to learn intra-point-cloud features, followed by a cross-attention module for feature exchange between input point clouds. Transformer models mainly benefit from the use of self-attention to capture the global correlations in feature space. However, the global correlations involved in self-attention may not only result in a significant amount of redundant computational overhead but also introduce feature ambiguities, especially in low-overlap scenarios. This is because overlapping regions of point clouds typically do not span a wide range but are rather concentrated around a localized area. Therefore, the correlations with an extensive range of non-overlapping points are ineffective and may degrade the discriminability of features. To address this issue, we present a **E**fficient **P**rior-**E**mbedded **A**ttention **L**earning model (**E**fficientPEAL). By incorporating overlap prior to the learning process, the point clouds are divided into two parts. One part includes points lying in the putative overlapping region and the other includes points located in the putative non-overlapping region. Then, EfficientPEAL performs localized attention with the putative overlapping points. The proposed attention module significantly reduces the computational complexity of the model while achieving competitive performance. Extensive experiments on 3DMatch/3DLoMatch, ScanNet, and KITTI datasets demonstrate its effectiveness.

1. Introduction

Point cloud registration is a fundamental yet challenging task in 3D computer vision and robotics. Its goal is to estimate the optimal rigid transformation that aligns point clouds acquired from different viewpoints into a common coordinate system, thereby achieving spatial consistency and facilitating 3D reconstruction. In particular, rigid point cloud registration, which focuses on estimating rotation and translation while preserving the shape of the object, plays a pivotal role in various applications such as 3D reconstruction, robotic navigation, simultaneous localization and mapping (SLAM), and autonomous driving (Aoki, Goforth, Srivatsan, & Lucey, 2019; Bai et al., 2021; El Banani, Gao, & Johnson, 2021; Qin et al., 2022).

Benefiting from the superior feature representation of deep networks, learning-based point cloud registration methods have become dominant in recent years (Huang, Gojcic, Usvyatsov, Wieser, & Schindler, 2021; Qin et al., 2023; Yu, Li, Saleh, Busam, & Ilic, 2021). These methods usually adopt the coarse-to-fine strategy and then learn features (or establish correspondences) between the downsampled point clouds (superpoints), which are then propagated to individual points to yield dense correspondences. Thus, the feature learning of superpoints is crucial to the overall performance of point cloud registration. Due to the sparse and loose nature of superpoint matching, existing methods (Huang et al., 2021; Li & Harada, 2022; Qin et al., 2023; Yu et al., 2021, 2023a) widely employ attention mechanisms for superpoint feature learning.

However, standard self-attention usually captures global correlations indiscriminately, including many irrelevant non-overlapping areas. This not only introduces feature ambiguity but also results in substantial computational overhead—particularly in partially overlapping and low-overlap scenarios. In other words, the correlations with non-overlapping superpoints are ineffective and may disrupt the inter-frame geometric consistency learning and degrade the feature discriminativeness for registration, which makes the resultant

* Corresponding author.
E-mail addresses: junleyu1996@163.com (J. Yu), zhouwenhui@hdu.edu.cn (W. Zhou), avianhao@gmail.com (Z. Shen), ywmiao@hznu.edu.cn (Y. Miao).

https://doi.org/10.1016/j.eswa.2025.128591
Received 11 January 2025; Received in revised form 29 May 2025; Accepted 11 June 2025
Available online 25 June 2025

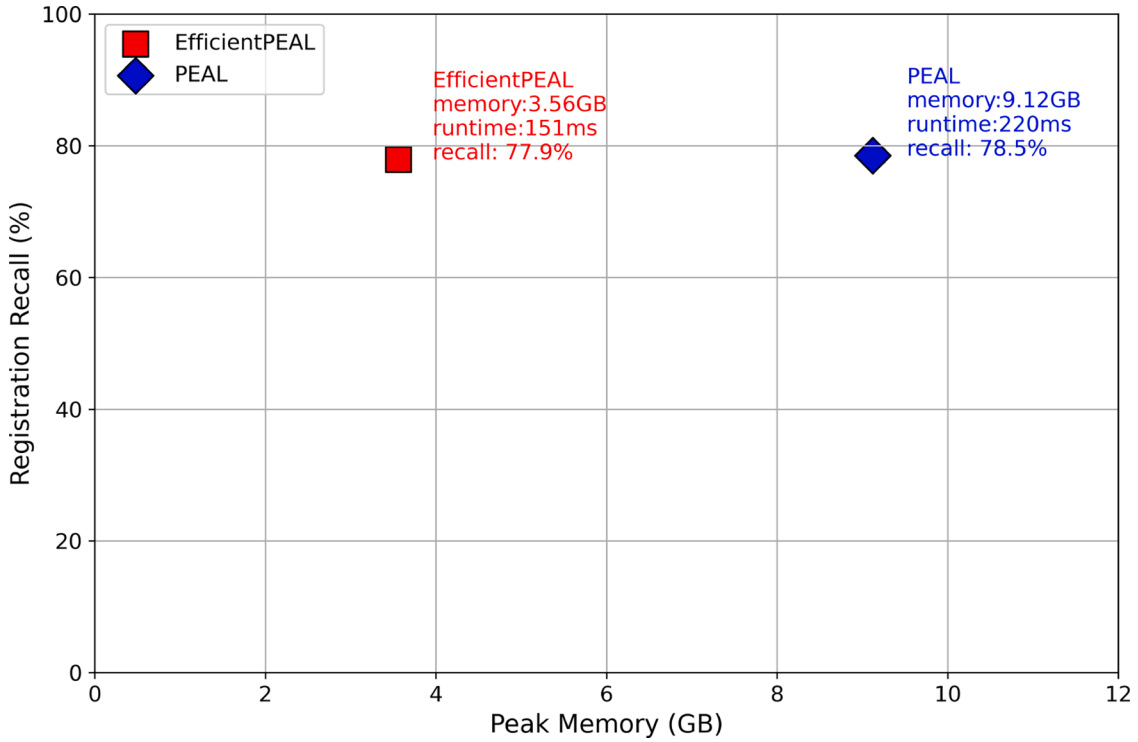

Fig. 1. Comparison of registration performance and resource consumption between EfficientPEAL and PEAL. EfficientPEAL achieves a comparable registration recall (77.9%) while significantly reducing peak memory usage (3.56GB vs. 9.12GB) and accelerating inference time (151ms vs. 220ms), demonstrating its superior efficiency and lightweight design.

learned superpoint features ambiguous and leads to numerous outlier matches.

To address this chicken-and-egg problem, we have proposed PEAL (Yu et al., 2023b) which predicts overlapping regions first, and then performs explicit one-way attention with the putative overlapping points to relieve the feature ambiguities of self-attention. On the coarse scale, the one-way attention module is plugged after self-attention, it explicitly models feature correlations with the localized area, thus avoiding interference from the numerous superpoints in non-overlapping areas. However, PEAL (Yu et al., 2023b) still suffers from high computational and memory costs due to its use of both global self-attention and one-way attention. These dual attention modules introduce latency and resource demands that limit its deployment in time-sensitive or resource-constrained applications. As shown in Fig. 1, although PEAL achieves strong accuracy, it incurs high peak memory usage (9.12 GB) and inference time (220 ms). Subsequent works (Chen et al., 2023; Jin, Armeni, Pollefeys, & Barath, 2024) have improved upon PEAL, but these methods focus primarily on denoising using diffusion models to enhance matching quality, without addressing aspects such as model lightweighting or acceleration.

Intuitively, when humans attempt to match the overlapping regions of two point clouds, they often begin by focusing on a small portion of the overlapping area. This initial focus allows them to develop a basic understanding of the overlapping objects or elements before expanding their observation to the entire region. Starting from this small portion, humans can gradually expand their observation scope by establishing connections between other areas and this small portion, ultimately encompassing the entire overlapping region. Therefore, we argue that once the local overlapping regions are identified, the global feature ambiguities can be resolved by learning correlations exclusively within these regions, eliminating the need for global feature associations through self-attention.

Motivated by the above observations, we propose **E**fficientPEAL to mitigate the memory footprint and computational cost of the PEAL (Yu et al., 2023b). Firstly, we follow PEAL to divide the superpoints into anchor ones (the superpoints lying in the putative overlapping region) and non-anchor ones (the superpoints located in the putative non-overlapping region). Then we propose a local geometric self-attention module to unify self-attention with one-way attention, which can establish the correlations with this small portion of putative overlapping (anchor) superpoints, thus alleviating the interference of non-anchor superpoints and significantly reducing the computational cost of PEAL. As shown in Fig. 1, compared to PEAL, EfficientPEAL reduces the peak memory usage by 60.9% (from 9.12GB to 3.56GB) and accelerates the inference time by 31.4% (from 220ms to 151ms), while achieving comparable registration accuracy (77.9% vs. 78.5%). These results demonstrate that EfficientPEAL delivers a well-balanced trade-off between accuracy and efficiency, making it suitable for real-time and resource-constrained applications.

The main technical contributions can be summarized as follows,

- We present a lite model EfficientPEAL. By reformulating self-attention, we unify the standard geometric self-attention and one-way attention into a single attention module, called local geometric self-attention. EfficientPEAL significantly reduces the complexity of PEAL, while maintaining performance.
- EfficientPEAL can integrate various overlap priors, such as 2D prior, 3D prior, hybrid 2D/3D prior and random prior. It can efficiently enhance the registration performance on the basis of existing methods.
- We further evaluate our method on the ScanNet and KITTI datasets, demonstrating its generality and effectiveness across different datasets.

The structure of this paper is organized as follows: Section 2 reviews some related work, including the current development and representative algorithms of point cloud registration methods. Section 3 presents the overall framework and key techniques of the proposed registration method in detail. Section 4 provides extensive experimental evaluations on multiple datasets, along with comparative analyses against existing

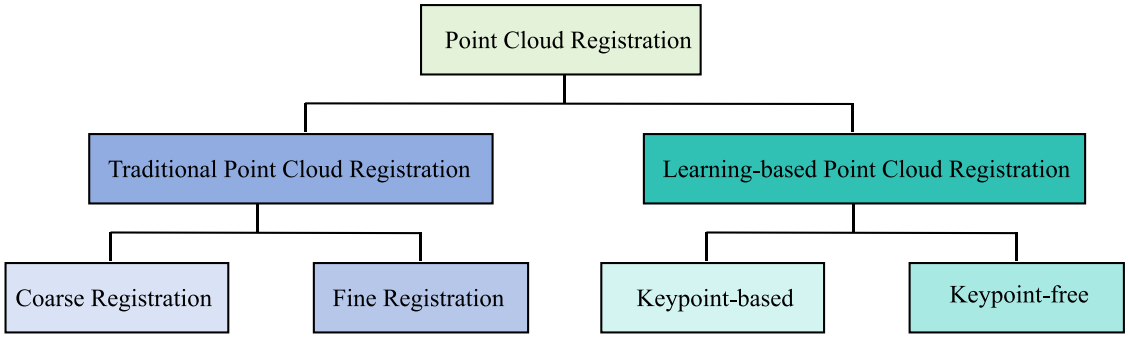

Fig. 2. An overview of the taxonomy of point cloud registration methods.

methods. Section 5 concludes the paper and discusses potential directions for future research.

2. Related work

This section reviews the related work pertinent to our study. We categorize point cloud registration methods into two main groups: traditional approaches and learning-based methods. Traditional registration techniques typically consist of two sequential stages: coarse registration, which provides an initial alignment, followed by fine registration to refine the transformation. Learning-based registration methods can be further divided into keypoint-based and keypoint-free approaches. A summary of the classification of point cloud registration methods is illustrated in Fig. 2. In addition, we also review relevant literature in the areas of multimodal learning and Transformer-based network architectures, which provide valuable insights and methodological inspiration for our work.

Traditional 3D registration methods. Traditional point cloud registration algorithms can be categorized into coarse registration and fine registration. Coarse registration methods (Guo, Sohel, Bennamoun, Lu, & Wan, 2013; Johnson & Hebert, 1999; Rusu, Blodow, & Beetz, 2009; Rusu, Blodow, Marton, & Beetz, 2008) aim to roughly align two point clouds from their initial positions, providing an initial estimate for subsequent fine registration. Fine registration performs more accurate alignment based on coarse registration. Among various registration methods, the Iterative Closest Point (ICP) algorithm (Besl & McKay, 1992; Chen & Medioni, 1992; Rusinkiewicz & Levoy, 2001), is the most well-known and efficient algorithm for aligning two 2D or 3D point sets under Euclidean (rigid) transformations. Given an initial transformation, ICP iteratively performs two steps on the two point sets to ensure convergence towards a locally optimal alignment: (1) finding the closest points as correspondences under the current transformation, (2) determining the optimal transformation that minimizes the distances between corresponding points using Singular Value Decomposition (SVD). However, ICP and its variants are prone to converging to local optima, and they are sensitive to noise, outliers, and partial overlap, with their success highly dependent on good initialization.

Learning-based point cloud registration methods. Recent learning-based point cloud registration methods are predominantly dominated by deep learning approaches based on correspondences. Correspondence-based methods (Choy, Park, & Koltun, 2019; Deng, Birdal, & Ilic, 2018; Gojcic, Zhou, Wegner, & Wieser, 2019) firstly establish correspondences between learned keypoints and then recover the transformation with a robust pose estimator such as RANSAC (Fischler & Bolles, 1981) or other RANSAC-free estimators (Bai et al., 2021; Choy, Dong, & Koltun, 2020). According to the way of extracting correspondences, these methods can be further divided into two categories, including keypoint-based methods and keypoint-free methods. The first category aims to detect more repeatable key points (Bai et al., 2020; Huang et al., 2021) and learn more powerful descriptors for key points (Ao et al., 2022; Ao, Hu, Yang, Markham, & Guo, 2021). The second category (Qin et al., 2022; Yu et al., 2021, 2023b), however, retrieves correspondences without key point detection by considering all possible matches. This method follows a detection-free approach and utilizes geometric information to improve the accuracy of correspondences. Our method focuses on further enhancing the feature discriminativeness of attention-based methods, it can be combined with both keypoint-based and keypoint-free methods.

Multi-modal learning. Multi-modal learning is currently a hot research area (Hou, Dai, & Nießner, 2019; Hou, Xie, Graham, Dai, & Nießner, 2021), and 2D-3D joint learning is a typical multi-modal learning branch. Pri3d (Hou et al., 2021) employs an implicit pretrain-and-finetune strategy to combine 2D and 3D knowledge for 2D downstream tasks. 3D-SIS (Hou et al., 2019) and RevalNet (Hou, Dai, & Nießner, 2020) propose an implicit fusion of color signal for 3D instance segmentation and detection tasks. ImVoteNet (Qi, Chen, Litany, & Guibas, 2020) explicitly uses 2D color input to enhance the performance of 3D object detection. BYOC (El Banani & Johnson, 2021), PCR-CG (Zhang, Yu, Huang, Zhou, & Hou, 2022) and (El Banani et al., 2021; Zhou, Ren, Yu, Qu, & Dai, 2024) proposes to explicitly fuse image features with point clouds to improve the accuracy of point cloud registration. In this paper, we employ the explicit fashion to leverage the 2D prior.

Transformers. The Transformer models (Devlin, Chang, Lee, & Toutanova, 2018; Vaswani et al., 2017) utilize a novel attention mechanism, employing multiple layers of self and cross multi-head attention to facilitate information exchange between input and output, demonstrating the superior performance of feature representation for NLP and vision tasks. Focal self-attention (Yang et al., 2021) is to incorporate fine-grained local and coarse-grained global interactions to capture both short and long-range visual dependencies. Deformable DETR (Zhu et al., 2020) uses deformable attention, which solely attends to a small set of key sampling points around a reference to aggregate multi-scale image features. GeoTransformer (Qin et al., 2022) utilizes geometric self-attention to extract transformation-invariant geometric features by encoding geometric structure. Masked-attention (Cheng, Misra, Schwing, Kirillov, & Girdhar, 2022) introduces the masked attention mechanism, which allows the model to better focus on object boundaries and details during segmentation, thereby improving segmentation accuracy. In this paper, we adopt an explicit attention learning fashion to learn discriminative features for low-overlap point clouds.

3. Method

We define point clouds $\mathcal{P} = \{\mathbf{p}_{x_i} \in \mathbb{R}^3 \mid i = 1, \dots, N\}$ and $\mathcal{Q} = \{\mathbf{q}_{y_i} \in \mathbb{R}^3 \mid i = 1, \dots, M\}$, as "source" and "target", respectively. The purpose of point cloud registration is to recover the unknown rigid transformation SE(3), which consists of a rotation $\mathbf{R} \in \mathrm{SO}(3)$ and a translation $\mathbf{T} \in \mathbb{R}^3$ for aligning $\mathcal{P}$ and $\mathcal{Q}$.

The pipeline of our method is illustrated in Fig. 3. Firstly, we follow PEAL (Yu et al., 2023b) to predict the overlap prior. Given the overlap

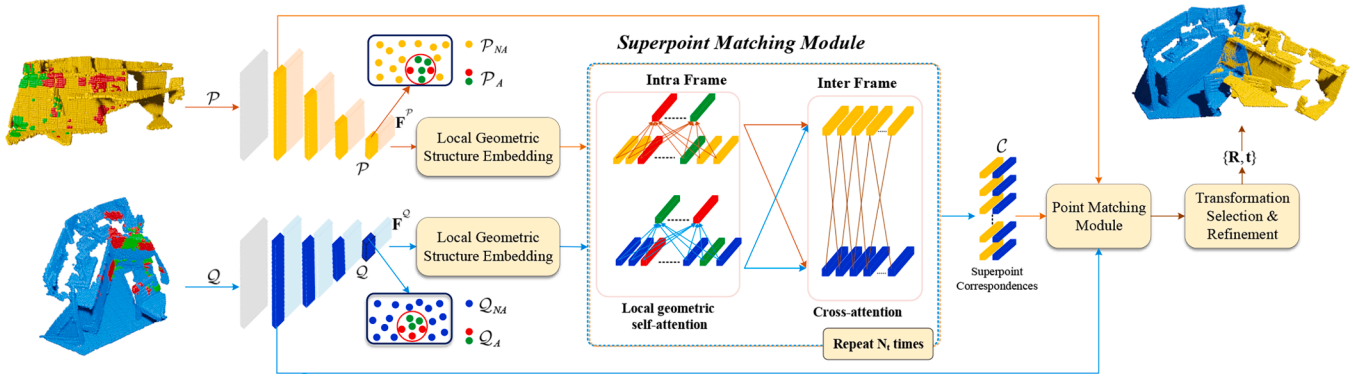

Fig. 3. The comprehensive framework of our method. The backbone KPConv-FPN downsamples the input point clouds and learns features in multiple resolution levels. The geometry structure embedding encodes a local geometry structure. The local geometric self-attention learns one-way correlations with the anchor superpoints (selected according to the given overlap prior, painted in red or green) to encode intra-frame geometric consistency. The discriminative features are then exchanged between two point clouds using a feature-based cross-attention. The resulting correspondences are then sent to the point matching module to calculate the transformation.

prior, we then employ a coarse-to-fine technique to extract correspondences. We build our method upon GeoTransformer (Qin et al., 2022), which utilizes KPConv-FPN (Thomas et al., 2019) to simultaneously downsample the input point clouds and extract point-wise features. The first and the last (coarsest) level of downsampled points correspond to the dense points and the superpoints, respectively. The superpoints are denoted by $\hat{\mathcal{P}}$ and $\hat{\mathcal{Q}}$, with their associated learned features denoted as $\hat{\mathbf{F}}^{\mathcal{P}}$ and $\hat{\mathbf{F}}^{\mathcal{Q}}$.

In $\hat{\mathcal{P}}$ and $\hat{\mathcal{Q}}$, the superpoints lying in the putative overlapping region are identified as anchor superpoints and denoted as $\hat{\mathcal{P}}_A$ and $\hat{\mathcal{Q}}_A$, respectively. The remaining superpoints are considered non-anchor superpoints and are denoted as $\hat{\mathcal{P}}_{NA}$ and $\hat{\mathcal{Q}}_{NA}$, respectively. Intra-frame attention modules, described in Sections 3.1 and 3.2, are employed to learn superpoint features. The two types of our explicit intra-frame attention modules are illustrated in Fig. 4. Furthermore, the inter-frame cross-attention is performed to facilitate feature exchange between the two point clouds. Then we follow GeoTransformer (Qin et al., 2022) to extract the superpoint correspondences, which are then propagated to the point matching module to obtain dense point correspondences for the calculation of rigid transformation.

3.1. Preliminaries

3.1.1. Revisiting self-attention mechanism

The self-attention mechanism (Vaswani et al., 2017) employs multi-head attention to jointly attend to information across diverse representation subspaces and positions.

$$\begin{aligned}\mathrm{MHAttn}(\mathbf{Q},\mathbf{K},\mathbf{V}) &= \mathrm{Concat}\left(\mathrm{head}_1,\ldots,\mathrm{head}_h\right)\mathbf{W}^O\\ \mathrm{head}_i &= \mathrm{Attn}\left(\mathbf{Q}\mathbf{W}_i^Q,\mathbf{K}\mathbf{W}_i^K,\mathbf{V}\mathbf{W}_i^V\right),\end{aligned} \tag{1}$$

where MHAttn denotes the multihead attention operation, and Attn denotes the attention operation. **Q**, **K**, and **V** represent queries, keys, and values, respectively. Additionally, $W_i^Q \in \mathbb{R}^{d_{\text{model}}\times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}}\times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}}\times d_v}$ and $W^O \in \mathbb{R}^{hd_v\times d_{\text{model}}}$. The computation of each attention head is described as:

$$\mathrm{Attn}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_t}}\right)\mathbf{V}. \tag{2}$$

In the self-attention layer of the point cloud registration task, the queries, keys, and values are typically set to the corresponding

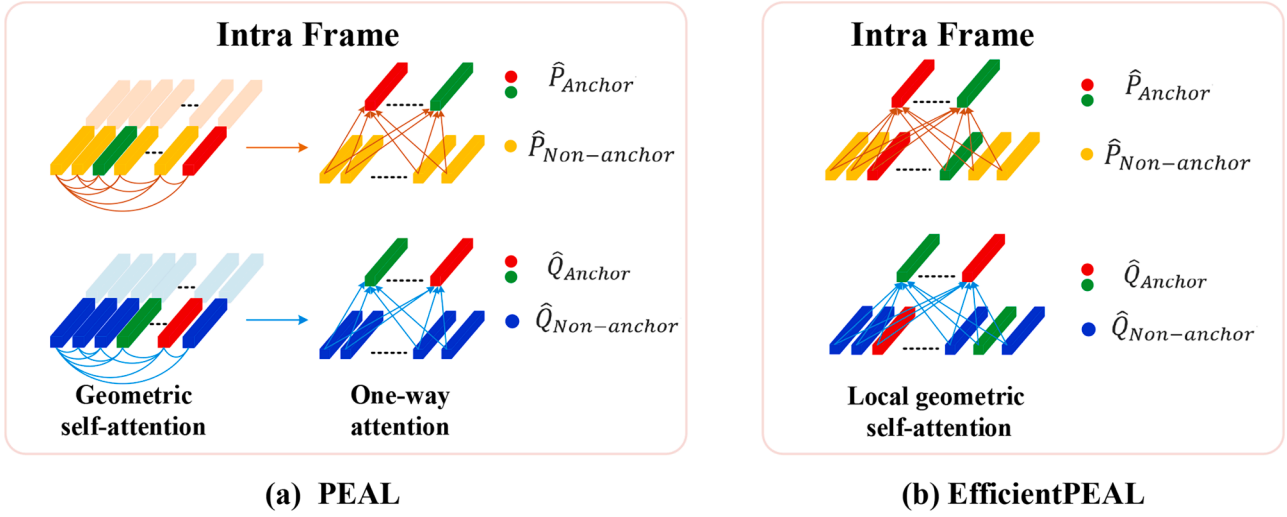

Fig. 4. The intra-frame attention module of standard PEAL and EfficientPEAL, EfficientPEAL indicates the lite model with local geometric self-attention. PEAL learns self-attention first, then models one-way feature correlations between anchor and non-anchor superpoints. While EfficientPEAL learns local geometric self-attention with the anchor superpoints to substitute self-attention.

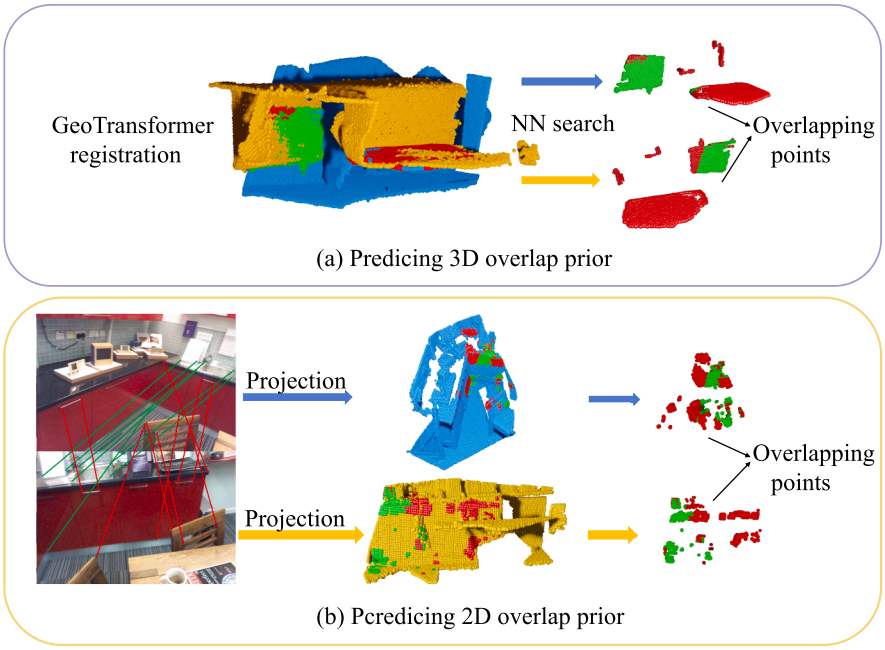

Fig. 5. Illustration of 3D and 2D overlap prior prediction used for anchor superpoint selection. (a) The 3D overlap prior is obtained by performing coarse registration using GeoTransformer, followed by extracting superpoints within the predicted overlapping region. (b) The 2D overlap prior is generated by projecting matched pixels from paired RGB images back into the 3D space to estimate coarse overlapping regions in the point clouds.

features extracted from the same point cloud. Specifically, in the context of the source point cloud, the self-attention mechanism is applied as MHAttn($\hat{\mathbf{F}}^{\mathcal{P}}, \hat{\mathbf{F}}^{\mathcal{P}}, \hat{\mathbf{F}}^{\mathcal{P}}$), and similarly for the target point cloud. This design choice enables points within each point cloud to attend to other informative regions within the same point cloud.

3.1.2. Revisiting PEAL

The registration performance of existing correspondence-based methods (Huang et al., 2021; Qin et al., 2022; Yu et al., 2021) heavily relies on discriminative superpoint features. Some recent methods, such as GeoTransformer (Qin et al., 2022), propose encoding global geometric structure to learn transformation-invariant features via the attention mechanism. However, utilizing self-attention for learning intra-frame point-wise features may introduce ambiguities. Because self-attention learns the global correlations, in which the incorrect correlations with a large range of non-overlapping superpoints introduce ambiguous ones. For the point cloud registration task, the correlations with the overlapping superpoints are the key to encoding inter-frame geometric consistency.

In our preliminary work PEAL (Yu et al., 2023b), we obtain the 3D overlap prior by applying GeoTransformer to predict a coarse transformation, followed by a nearest neighbor search within a distance threshold to identify overlapping regions, as illustrated in Fig. 5 (a) . For the 2D overlap prior, we follow PCR-CG (Zhang et al., 2022) by using SuperGlue (Sarlin, DeTone, Malisiewicz, & Rabinovich, 2020) to extract 2D image correspondences, expand them with bounding boxes, and project the matched pixels into 3D space, as shown in Fig. 5 (b) . Superpoints whose local patches contain prior points are then defined as anchor superpoints. PEAL divides the points into the anchor and non-anchor regions, which allows for a higher overlap ratio in the anchor region. Then it utilizes an explicit one-way attention module followed by self-attention to avoid introducing ambiguous geometric correlations from non-anchor superpoints. Since the complexity of one-way attention depends on the number of queries (input points) and keys (anchor superpoints), a higher number of anchor superpoints leads to lower computational efficiency of the attention module.

PEAL achieves superpoint matching by utilizing three attention modules, including a geometric self-attention module (Qin et al., 2022) to learn intra-frame point-wise geometric features, an explicit one-way attention module to enhance inter-frame local geometric consistency, and a feature-based cross-attention module to perform feature exchange between “source” and “target” point clouds. These three attention modules are interleaved for N_t times to extract hybrid features $\hat{\mathbf{H}}^{\mathcal{P}}$ and $\hat{\mathbf{H}}^{\mathcal{Q}}$ for reliable superpoint matching.

Explicit one-way attention. Followed by self-attention, then PEAL (Yu et al., 2023b) utilizes a one-way attention module to explicitly learn the intra-frame correlation with the anchor superpoints. Given anchor superpoints attention feature matrices $\mathbf{X}^{\hat{P}_A}$ and non-anchor superpoints attention feature matrices $\mathbf{X}^{\hat{P}_{NA}}$, the non-anchor superpoints attention features $\mathbf{Z}^{\hat{P}_{NA}}$ can be updated by the features of anchor superpoints $\mathbf{X}^{\hat{P}_A}$, which is computed as:

$$\mathbf{z}_n^{\hat{P}_{NA}} = \sum_{m=1}^{|\hat{P}_A|} a_{n,m}\left(\mathbf{x}_m^{\hat{P}_A}\mathbf{W}^V\right). \tag{3}$$

Similar to geometric self-attention, $a_{n,m}$ represents a row-wise softmax on the attention score $e_{n,m}$, which is the feature correlation between the $\mathbf{X}^{\hat{P}_{NA}}$ and $\mathbf{X}^{\hat{P}_A}$:

$$e_{n,m} = \frac{\left(\mathbf{x}_n^{\hat{P}_{NA}}\mathbf{W}^Q\right)\left(\mathbf{x}_m^{\hat{P}_A}\mathbf{W}^K\right)^T}{\sqrt{d_t}}. \tag{4}$$

The attention features of $\mathbf{X}^{\hat{Q}_{NA}}$ are updated same as $\mathbf{X}^{\hat{P}_{NA}}$, while $\mathbf{X}^{\hat{Q}_A}$ and $\mathbf{X}^{\hat{P}_A}$ remain unchanged.

3.2. Local geometric self-attention

When aligning two partially overlapping 3D point clouds, humans typically begin by identifying a small, recognizable region shared by both scans—such as overlapping tables or chairs—and then progressively expand their attention to adjacent areas by leveraging spatial continuity and structural context. This localized and incremental matching strategy naturally avoids distraction from unrelated regions and enables more accurate alignment. Similarly, in partially overlapping point cloud

scenarios, only a small subset of superpoints—those located within the actual overlapping regions—contribute meaningfully to the matching process. However, conventional self-attention mechanisms indiscriminately model global correlations across all superpoints, without differentiating between overlapping and non-overlapping regions. This often results in redundant computation and interference from irrelevant areas, which significantly hampers feature discriminability, especially under low-overlap conditions. To address these limitations, we propose a local geometric self-attention mechanism. By explicitly constraining the attention scope, this module models feature correlations only among anchor superpoints that are predicted to lie within the overlapping regions, thereby effectively suppressing the influence of non-overlapping areas. To illustrate this, let's consider the example of a downsampled point cloud $\hat{\mathcal{P}}$ and its corresponding anchor superpoint cloud $\hat{\mathcal{P}}_A$. While the queries are set to the point cloud as usual, the keys and values are set to the anchor superpoint cloud. This allows us to formulate self-attention as $\mathrm{MHAttn}(\mathbf{Q}=\hat{\mathbf{F}}^{\mathcal{P}},\mathbf{K}=\hat{\mathbf{F}}^{\mathcal{P}},\mathbf{V}=\hat{\mathbf{F}}^{\mathcal{P}})$ as formulated as:

$$\mathrm{MHAttn}(\mathbf{Q}=\hat{\mathbf{F}}^{\mathcal{P}},\mathbf{K}=\hat{\mathbf{F}}^{\mathcal{P}_A}(\mathbf{K}_A),\mathbf{V}=\hat{\mathbf{F}}^{\mathcal{P}_A}(\mathbf{V}_A)). \tag{5}$$

Therefore, the computation of the attention scores can be formulated as:

$$e_{i,m}=\frac{\left(\mathbf{x}_i^{\hat{\mathcal{P}}}\mathbf{W}^{Q}\right)\left(\mathbf{x}_m^{\hat{\mathcal{P}}_A}\mathbf{W}^{K_A}+\mathbf{r}_{i,m}\mathbf{W}^{R_A}\right)^{T}}{\sqrt{d_t}}, \tag{6}$$

where $\mathbf{r}_{i,m}$ refers to the local geometric structure embedding. Then we perform the softmax function to the attention score: $a_{i,m}=\mathrm{softmax}\left(e_{i,m}\right)$. Finally, the output feature matrix $\mathbf{Z}\in\mathbb{R}^{|\hat{\mathcal{P}}|\times d_t}$ is obtained by performing a weighted sum of all projected input features:

$$\mathbf{z}_i=\sum_{m=1}^{|\hat{\mathcal{P}}_A|}a_{i,m}\left(\mathbf{x}_m^{\hat{\mathcal{P}}_A}\mathbf{W}^{V_A}\right). \tag{7}$$

Local geometric structure embedding. As shown in Fig. 6, the standard geometric structure embedding (Qin et al., 2022) computes global pairwise distance embeddings and triplet angle embeddings to model the geometric structure of the point cloud, providing invariance to rigid transformations. However, in point cloud registration tasks, especially when only partial overlap exists between frames, we argue that global geometric encoding may introduce redundancy or even noise. Instead, encoding geometric consistency across frames can be achieved more effectively by solely focusing on the geometric structures within the overlapping regions. Local distance and angular embeddings capture the geometric structure information of overlapping points in the point cloud, reducing ambiguity in feature description and constructing rotation-invariant feature representations, thereby enhancing the discrimination of features. Given superpoint clouds $\hat{\mathcal{P}}$ and $\hat{\mathcal{Q}}$, as well as explicit putative overlapping (anchor) superpoints $\left(\hat{\mathbf{p}}_{x_i},\hat{\mathbf{q}}_{y_j}\right)$ obtained from existing 2D/3D matching methods, we modify the standard geometric embedding to a local geometric embedding. The computation of the local geometric embedding is described as follows:

(1) *Local distance emebedding.* In the standard *Pair-wise Distance Embedding*, the distance embedding is computed as:

$$\rho_{i,j}=\left\|\hat{\mathbf{p}}_i-\hat{\mathbf{p}}_j\right\|_2,\hat{\mathbf{p}}_i\in\hat{\mathcal{P}},\hat{\mathbf{p}}_j\in\hat{\mathcal{P}}. \tag{8}$$

Here, we compute the distance between superpoints $\hat{\mathcal{P}}$ and anchor superpoints $\hat{\mathcal{P}}_A$. This approach focuses on capturing a more discriminative geometric structure, which aims to avoid introducing ambiguous geometric structures from non-anchor superpoints.

$$\rho_{i,m}=\left\|\hat{\mathbf{p}}_i-\hat{\mathbf{p}}_m\right\|_2,\quad \hat{\mathbf{p}}_i\in\hat{\mathcal{P}},\hat{\mathbf{p}}_m\in\hat{\mathcal{P}}_A. \tag{9}$$

Then, the distance embedding $\mathbf{r}_{i,n}^{D}$ between $\hat{\mathbf{p}}_i$ and $\hat{\mathbf{p}}_n$ is computed by applying a sinusoidal function (Vaswani et al., 2017) on $\rho_{i,n}/\sigma_d$. Here, same as (Qin et al., 2022), σ_d is a hyper-parameter used to tune the sensitivity on distance variations:

$$\begin{cases} r_{i,m,2k}^{D}=\sin\left(\frac{d_{i,m}/\sigma_d}{10000^{2k/d_t}}\right)\\ r_{i,m,2k+1}^{D}=\cos\left(\frac{d_{i,m}/\sigma_d}{10000^{2k/d_t}}\right).\end{cases} \tag{10}$$

(2) *Local angular emebedding.* We select the K nearest neighbors $\mathcal{K}_x$ of $\hat{\mathbf{p}}_m$. Given $\hat{\mathbf{p}}_x\in\mathcal{K}_m$, similar to *Local distance embedding* in Eq. 9, we compute the explicit local angle structure of superpoints as:

$$\alpha_{i,m}^{x}=\angle\left(\Delta_{x,i},\Delta_{m,i}\right),\quad \hat{\mathbf{p}}_i\in\hat{\mathcal{P}},\hat{\mathbf{p}}_m\in\hat{\mathcal{P}}_A. \tag{11}$$

where $\Delta_{i,m}=\hat{\mathbf{p}}_i-\hat{\mathbf{p}}_m$. The triplet-wise angular embedding $\mathbf{r}_{i,m,k}^{A}$ is then computed with a sinusoidal function on $\alpha_{i,m}^{k}/\sigma_a$, with σ_a controlling the sensitivity on angular variations:

$$\begin{cases} r_{i,m,y,2l}^{A}=\sin\left(\frac{\alpha_{i,m}^{y}/\sigma_a}{10000^{2l/d_t}}\right)\\ r_{i,m,y,2l+1}^{A}=\cos\left(\frac{\alpha_{i,m}^{y}/\sigma_a}{10000^{2l/d_t}}\right).\end{cases} \tag{12}$$

Finally, we compute the geometric structure embedding $\mathbf{r}_{i,m}$ by aggregating the pair-wise distance embedding and the triplet-wise angular embedding using projection matrices $\mathbf{W}^{D}$ and $\mathbf{W}^{A}$:

$$\mathbf{r}_{i,m}=\mathbf{r}_{i,m}^{D}\mathbf{W}^{D}+\max_y\left\{\mathbf{r}_{i,m,y}^{A}\mathbf{W}^{A}\right\}, \tag{13}$$

where $\mathbf{W}^{D}$ and $\mathbf{W}^{A}$ are projection matrices of size $d_t\times d_t$ for the distance and angular embeddings, respectively. Taking the point cloud $\hat{\mathcal{P}}$ as an example, the detailed procedure of the local geometric self-attention

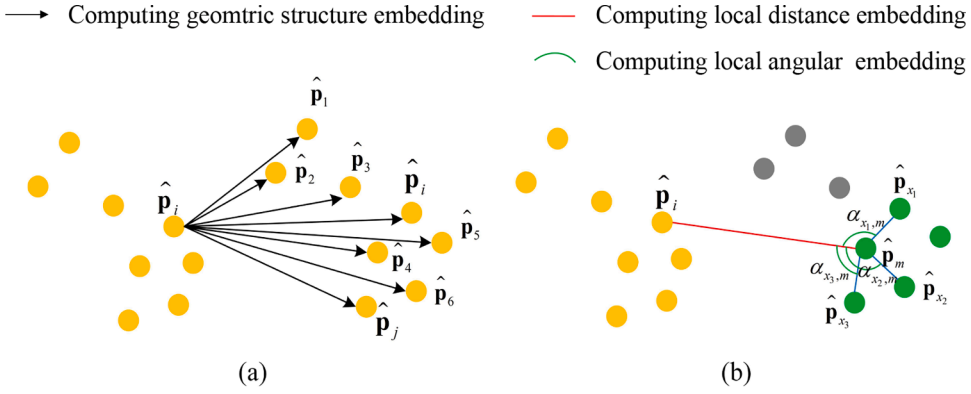

Fig. 6. The illustration shows the standard geometric structure embedding on the left and the local geometric structure embedding on the right. In the standard setting, global pairwise geometric structures are computed across all points. In contrast, the local geometric structure embedding focuses solely on the geometric relationships within the overlapping regions (highlighted in green in Fig.(b)), thereby reducing interference from non-overlapping areas. In Fig.(b), the red lines indicate the distances computed between a query point and the overlapping points, while the green arcs represent the triplet angles formed between the query point and its neighboring overlapping points.

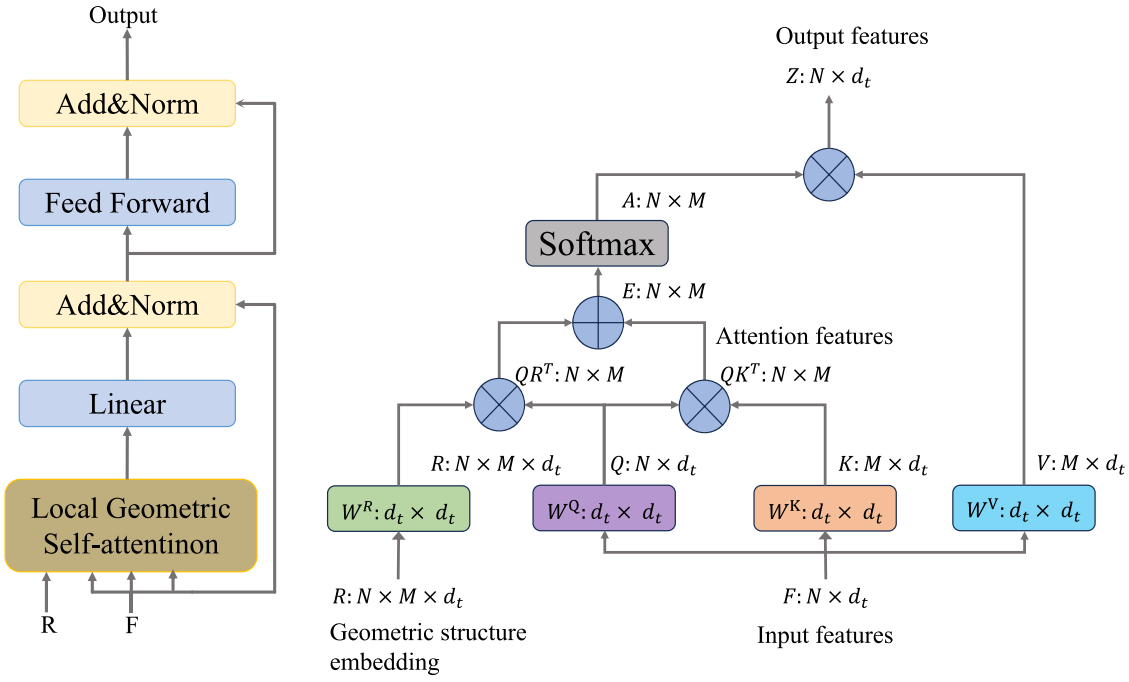

Fig. 7. Left: The structure of local geometric self-attention. Right: The computation of the local geometric self-attention mechanism. N represents the size of input superpoints, while M represents the size of input anchor superpoints.

Algorithm 1 Local geometric self-attention.

Input: Superpoints $\hat{P}$, anchor superpoints $\hat{P}_A$; feature matrix $\mathbf{F}_{\hat{P}}$ (superpoints), $\mathbf{F}_{\hat{P}_A}$ (anchor superpoints); neighbor size K
Output: Updated features $\mathbf{Z} \in \mathbb{R}^{|\hat{P}| \times d_t}$
Step 1: Compute local geometric structure embedding
for each $\hat{p}_i \in \hat{P}$ and $\hat{p}_m \in \hat{P}_A$ **do**
  Compute local distance: $\rho_{i,m} = \|\hat{p}_i - \hat{p}_m\|_2$
  Compute sinusoidal distance embedding: $\mathbf{r}^D_{i,m}$
  Find K neighbors of $\hat{p}_m$ in $\hat{P}_A$
  for each neighbor $\hat{p}_x$ of $\hat{p}_m$ **do**
    Compute angle: $\alpha^x_{i,m} = \angle(\hat{p}_x - \hat{p}_m, \hat{p}_i - \hat{p}_m)$
    Compute angular embedding: $\mathbf{r}^{A,x}_{i,m}$
  end for
  $\mathbf{r}_{i,m} = \mathbf{r}^D_{i,m}\mathbf{W}^D + \max_x(\mathbf{r}^{A,x}_{i,m}\mathbf{W}^A)$
end for

Step 2: Compute local geometric self-attention
$\mathbf{Q} = \mathbf{F}_{\hat{P}}\mathbf{W}^Q$
$\mathbf{K} = \mathbf{F}_{\hat{P}_A}\mathbf{W}^K + \mathbf{R}\mathbf{W}^R$
$\mathbf{V} = \mathbf{F}_{\hat{P}_A}\mathbf{W}^V$
$\mathbf{A} = \text{softmax}(\mathbf{Q}\mathbf{K}^\top / \sqrt{d_t})$
$\mathbf{Z} = \mathbf{A}\mathbf{V}$
return Z

module is summarized in Algorithm 1. The computation process of the local geometric self-attention is illustrated in Fig. 7.

Next, an inter-frame feature-based cross-attention module is used to facilitate the exchange of features between the input point clouds (Qin et al., 2022). The resultant superpoint features are then sent to the superpoint matching module for reasoning precise superpoint correspondences. Finally, we perform point matching (Qin et al., 2022) to extract the dense point correspondences and local-to-global registration to obtain the estimated transformation.

Loss functions. We build our method upon GeoTransformer (Qin et al., 2022), and inherent their losses as well. The definition of overlap-aware circle loss, as well as point matching loss, are identical to GeoTransformer.

4. Experimental results

In this section, we conduct extensive experiments to evaluate the effectiveness of our approach. We first evaluate our method on indoor 3DMatch (Zeng et al., 2017) and 3DLoMatch (Huang et al., 2021) benchmarks (Section 4.1) using both 2D prior and 3D prior. To further evaluate the effectiveness of using 2D prior, we evaluate our approach on the largest RGB-D dataset ScanNet (Dai et al., 2017), where the model is trained on the 3DMatch RGB-D dataset (Section 4.2). Finally, we conduct ablation studies and evaluate our method on KITTI dataset (Geiger, Lenz, & Urtasun, 2012) in Section 4.3. In the experimental tables, **boldfaced** numbers highlight the best and the second best are underlined.

4.1. 3DMatch & 3DLoMatch

Dataset. 3DMatch (Zeng et al., 2017) is composed of 62 scenes, among which 46 are used for training, 8 for validation, and 8 for testing. Each scene is associated with RGB-D data. We evaluate our approach using the preprocessed training point clouds provided by Huang et al. (2021), and test it on both the 3DMatch and 3DLoMatch protocols. The point cloud pairs on 3DMatch have an overlap greater than 0.3, while those in 3DLoMatch exhibit lower overlap ranging from 0.1 to 0.3.

We collect the corresponding RGB-D data (Zhang et al., 2022) for both benchmarks, where each point cloud is generated by fusing 50 consecutive depth frames. Since the RGB and depth images are paired, each point cloud is also associated with 50 RGB frames.

Experiments setup. The 3D overlap prior is acquired by performing a nearest neighbor search under the estimated transformation, with a threshold of 0.0375m to determine the predicted overlap region. To obtain 2D overlap prior, considering the computation cost of inferencing 2D correspondences, we use three frames of images for both the source and target point clouds. Correspondences are extracted from nine pairs of images using Superglue (Sarlin et al., 2020), with an image resolution of [640,480]. The XYZ coordinates of each point cloud are projected

Table 1
Evaluation results on 3DMatch and 3DLoMatch.

	3DMatch					3DLoMatch				
Samples	5000	2500	1000	500	250	5000	2500	1000	500	250
Feature Matching Recall(%) ↑										
PerfectMatch (Gojcic et al., 2019)	95.0	94.3	92.9	90.1	82.9	63.6	61.7	53.6	45.2	34.2
FCGF (Choy et al., 2019)	97.4	97.3	97.0	96.7	96.6	76.6	75.4	74.2	71.7	67.3
D3Feat (Bai et al., 2020)	95.6	95.4	94.5	94.1	93.1	67.3	66.7	67.0	66.7	66.5
SpinNet (Ao et al., 2021)	97.6	97.2	96.8	95.5	94.3	75.3	74.9	72.5	70.0	63.6
Predator (Huang et al., 2021)	96.6	96.6	96.5	96.3	96.5	78.6	77.4	76.3	75.7	75.3
YOHO (Wang, Liu, Dong, & Wang, 2022)	98.2	97.6	97.5	97.7	96.0	79.4	78.1	76.3	73.8	69.1
CoFiNet (Yu et al., 2021)	98.1	98.3	98.1	98.2	98.3	83.1	83.5	83.3	83.1	82.6
GeoTransformer (Qin et al., 2022)	97.9	97.9	97.9	97.9	97.6	88.3	88.6	88.8	88.6	88.3
RoITr (Yu et al., 2023a)	98.0	98.0	97.9	98.0	97.9	**89.6**	**89.6**	**89.5**	**89.4**	**89.3**
PEAL-3D(Yu et al., 2023b)	**98.5**	**98.5**	**98.4**	98.4	**98.4**	88.7	89.0	88.8	88.6	88.7
EfficientPEAL-3D	98.2	98.1	98.3	**98.5**	98.3	86.4	86.8	87.4	87.7	87.4
Inlier Ratio (%) ↑										
PerfectMatch (Gojcic et al., 2019)	36.0	32.5	26.4	21.5	16.4	11.4	10.1	8.0	6.4	4.8
FCGF (Choy et al., 2019)	56.8	54.1	48.7	42.5	34.1	21.4	20.0	17.2	14.8	11.6
D3Feat (Bai et al., 2020)	39.0	38.8	40.4	41.5	41.8	13.2	13.1	14.0	14.6	15.0
SpinNet (Ao et al., 2021)	47.5	44.7	39.4	33.9	27.6	20.5	19.0	16.3	13.8	11.1
Predator (Huang et al., 2021)	58.0	58.4	57.1	54.1	49.3	26.7	28.1	28.3	27.5	25.8
YOHO (Wang et al., 2022)	64.4	60.7	55.7	46.4	41.2	25.9	23.3	22.6	18.2	15.0
CoFiNet (Yu et al., 2021)	49.8	51.2	51.9	52.2	52.2	24.4	25.9	26.7	26.8	26.9
GeoTransformer (Qin et al., 2022)	71.9	75.2	76.0	82.2	85.1	43.5	45.3	46.2	52.9	57.7
RoITr (Yu et al., 2023a)	**82.6**	**82.8**	**93.0**	83.0	83.0	**54.3**	**54.6**	55.1	55.2	55.3
PEAL-3D(Yu et al., 2023b)	73.9	80.5	85.5	**87.4**	**88.6**	47.9	53.6	**60.0**	**62.7**	**64.6**
EfficientPEAL-3D	73.1	79.8	84.6	86.5	87.7	47.1	52.8	59.2	61.8	63.6
Registration Recall(%) ↑										
PerfectMatch (Gojcic et al., 2019)	78.4	76.2	71.4	67.6	50.8	33.0	29.0	23.3	17.0	11.0
FCGF (Choy et al., 2019)	85.1	84.7	83.3	81.6	71.4	40.1	41.7	38.2	35.4	26.8
D3Feat (Bai et al., 2020)	81.6	84.5	83.4	82.4	77.9	37.2	42.7	46.9	43.8	39.1
SpinNet (Ao et al., 2021)	88.6	86.6	85.5	83.5	70.2	59.8	54.9	48.3	39.8	26.8
Predator (Huang et al., 2021)	89.0	89.9	90.6	88.5	86.6	59.8	61.2	62.4	60.8	58.1
YOHO (Wang et al., 2022)	90.8	90.3	89.1	88.6	84.5	65.2	65.5	63.2	56.5	48.0
CoFiNet (Yu et al., 2021)	89.3	88.9	88.4	87.4	87.0	67.5	66.2	64.2	63.1	61.0
Lepard (Li & Harada, 2022)			93.5					69.0		
GeoTransformer (Qin et al., 2022)	92.0	91.8	91.8	91.4	91.2	75.0	74.8	74.2	74.1	73.5
RoITr (Yu et al., 2023a)	91.9	91.7	91.8	91.4	91.0	74.7	74.8	74.8	74.2	73.6
PEAL-3D(Yu et al., 2023b)	94.6	94.2	93.9	93.7	93.2	79.5	**79.5**	**79.1**	**78.4**	**77.6**
SemReg (Fung et al., 2025)			**94.7**					75.9		
EfficientPEAL-3D	94.0	93.2	93.4	93.2	92.7	79.0	78.4	78.4	78.1	77.3

onto their associated image planes to establish the pixel-point correspondence relationship, with a depth threshold of 0.2m. During training,we use a single GPU (RTX3090) and set the batch size to one due to the high memory demands of point cloud registration tasks and to maintain consistency with prior work such as GeoTransformer (Qin et al., 2022) and PEAL (Yu et al., 2023b). This setup allows fair comparison and ensures accessibility for researchers with limited hardware resources. The model is trained for 20 epochs using the Adam optimizer with a learning rate of 1e-4, and tested with six iterative updates for transformation refinement. The remaining experimental setup follows the same configuration as GeoTransformer (Qin et al., 2022).

Metrics. We compare the results on five metrics as prior works (Qin et al., 2022; Yu et al., 2023b), namely Registration Recall (RR), Feature Matching Recall (FMR), Inlier ratio (IR), Relative Rotation Error (RRE) as well as Relative Translation Error (RTE), the definitions of these metrics are as follows: (1) Inlier Ratio (IR) measures the fraction of putative correspondences whose residuals are below a certain threshold (i.e., 0.1m) under the ground-truth transformation, (2) Feature Matching Recall (FMR) represents the fraction of point cloud pairs whose inlier ratio is above a certain threshold (i.e., 5%), with less than 10 cm residual under the ground truth transformation, even if the transformation can not be recovered from those matches. In practical applications, there is often a trade-off between FMR and IR: increasing FMR retains more potential matches and improves coverage, but may introduce more outliers and reduce IR; on the other hand, overly emphasizing IR may discard critical correspondences, compromising the robustness of pose estimation. In pose estimation tasks, a higher IR contributes to more accurate estimation, but overly low FMR may still lead to failure. In the point cloud registration task, Registration Recall is the main metric we compare on and is most reliable. (3) Registration Recall (RR) represents the fraction of point cloud pairs whose transformation error is smaller than a certain threshold (i.e., RMSE $<$ 0.2m), (4) Relative Translation (RTE) and Rotation Errors (RRE) measure the deviations from the ground truth pose. More specifically, the RTE is computed by the differences between two frames by L1 norm; RRE is the drifted degrees between two frames registered by the predicted transformation. Please refer to supplementary materials for detailed explanations and equation definitions.

Main results. Following Bai et al. (2020) and Huang et al. (2021), we report the results with different numbers of correspondences using the RANSAC estimator in Table 1. Compared with PEAL (Yu et al., 2023b), EfficientPEAL achieves comparable *Feature Matching Recall* on both 3DMatch and 3DLoMatch. For *Inlier Ratio*, Our method improves by no less than 2.6% on 3DLoMatch and 1.2% on 3DMatch compared to the baseline GeoTransformer (Qin et al., 2022). The improvement is more prominent with fewer sampled correspondences, for instance, EfficientPEAL-3D achieves 59+% *Inlier Ratio* on 1000, 500, and 250 sampled correspondences on 3DLoMatch. For *Registration Recall*, our method outperforms GeoTransformer (Qin et al., 2022) by 4.0% on 3DLoMatch.

Table 2
Registration results without RANSAC on 3DMatch (3DM) and 3DLoMatch (3DLM).

Models	Estimator	Samples	RR(%)	
			3DM	3DLM
CoFiNet (Yu et al., 2021)	LGR	all	87.6	64.8
REGTR (Yew & Lee, 2022)	RANSAC-free	all	92.0	64.8
GeoTransformer (Qin et al., 2022)	LGR	all	91.5	74.0
PEAL-3D(Yu et al., 2023b)	LGR	all	94.5	79.2
EfficientPEAL-3D	LGR	all	93.4	78.8
PEAL-2D(Yu et al., 2023b)	LGR	all	**95.2**	**81.2**
EfficientPEAL-2D	LGR	all	94.5	80.3

Table 3
Evaluation of transformer complexity, GPU comsumption and runtime on 3DLoMatch, we don't use the iterative updating module in this experiment.

Models	Comp.	RR(%)	GPU (GB)	Infer.(s)
PEAL-3D(Yu et al., 2023b)	$O(N^2) + O(k_1N^2)$	**78.5**	9.12	0.220
EfficientPEAL-3D	$O(k_2N^2)$	77.9	**3.56**	**0.151**

Table 4
Relative Rotation Errors (RRE) and Relative Translation Errors (RTE) on 3DMatch and 3DLoMatch benchmarks.

	Estimator	3DMatch		3DLoMatch	
		RRE (°)	RTE (m)	RRE (°)	RTE (m)
Predator (Huang et al., 2021)	RANSAC-*50k*	2.029	0.064	3.048	0.093
CoFiNet (Yu et al., 2021)	RANSAC-*50k*	2.002	0.064	3.271	0.090
PCR-CG Zhang et al. (2022)	RANSAC-*50k*	1.993	0.061	3.002	0.087
GeoTransformer (Qin et al., 2022)	RANSAC-free	1.772	0.061	2.849	0.088
REGTR (Yew & Lee, 2022)	RANSAC-free	**1.567**	**0.049**	2.827	**0.077**
PEAL-3D(Yu et al., 2023b)	RANSAC-free	1.749	0.062	**2.711**	0.085
EfficientPEAL-3D	RANSAC-free	1.745	0.061	2.789	0.086

RANSAC-free estimator. We compare the registration results of the RANSAC-free estimator in Table 2. We separate our models into EfficientPEAL-2D and EfficientPEAL-3D according to the method of acquiring overlap prior. Both of them achieve comparable performance compared with PEAL (Yu et al., 2023b) on 3DMatch and 3DLoMatch. The improvement over the baseline is more prominent than using the RANSAC estimator. When 2D information is introduced, our method can achieve higher performance. Specifically, EfficientPEAL-2D improves previous best (REGTR Yew & Lee, 2022 on 3DMatch, GeoTransformer Qin et al., 2022 in 3DLoMatch) by 2.5% on 3DMatch and 6.3% on 3DLoMatch. Similarly, EfficientPEAL-3D outperforms the baseline by 1.9% on 3DMatch and 4.8% on 3DLoMatch, showcasing significant improvements compared to the baseline method. In conclusion, both EfficientPEAL-3D and EfficientPEAL-2D demonstrate noteworthy advancements in registration accuracy compared to GeoTransformer.

Complexity and runtime. The computational complexity of the standard self-attention and one-way attention used in PEAL (Yu et al., 2023b) is approximately $O(|\hat{P}|d_t^2 + |\hat{P}|^2 d_t^2) + O(|\hat{P}||\hat{P}_{\mathcal{A}}|d_t^2)$, which may become a limitation in terms of scalability and efficiency when dealing with a large number of superpoints. We reformulate the self-attention module by focusing on modeling the one-way correlation with the anchor superpoints. This modification reduces the computation complexity to $O(|\hat{P}||\hat{P}_{\mathcal{A}}|d_t^2)$.

We compare the model complexity with the baseline methods using LGR estimator in Table 3. For simplicity, we approximate the complexity of geometric self-attention as $O(N^2)$. The computational complexity of the proposed method is evaluated by comparing the inference time and peak GPU memory occupancy (GB). The complexity of local geometric attention is denoted as $O(k_2N^2)$, where k_2 represents the ratio of anchor superpoints to input superpoints. In this study, we compute the average ratio between the number of anchor superpoints and input superpoints on the 3DLoMatch dataset, resulting in $k_1 = 0.18$ and $k_2 = 0.29$. As shown in the table, EfficientPEAL-3D achieves comparable registration performance with PEAL while reducing the complexity of the Transformer by 75%. Additionally, EfficientPEAL-3D demonstrates significant speed improvements in terms of inference time when compared to PEAL-3D (Yu et al., 2023b). Given the practical application, we provide the peak GPU memory occupancy (GB) in this table. On GPU memory occupancy, EfficientPEAL-3D is only about 39% of PEAL (Yu et al., 2023b). Experimental results demonstrate that EfficientPEAL achieves comparable accuracy to the standard PEAL while significantly reducing the computation complexity and time. By explicitly considering local geometric consistency, the local geometric self-attention module effectively addresses the issue of global feature ambiguities present in self-attention.

RRE and RTE. We then compare the RRE and RTE with the recent state-of-the-arts in Table 4. As shown in this table, we achieve the second best in RRE, RTE on 3DMatch, and RTE on 3DLoMatch, and the best in RRE on 3DLoMatch.

Qualitative results. We visualize the registration results of EfficientPEAL, PEAL (Yu et al., 2023b) and GeoTransformer (Qin et al., 2022) in Fig. 8. Our approach, which incorporates explicit attention learn-

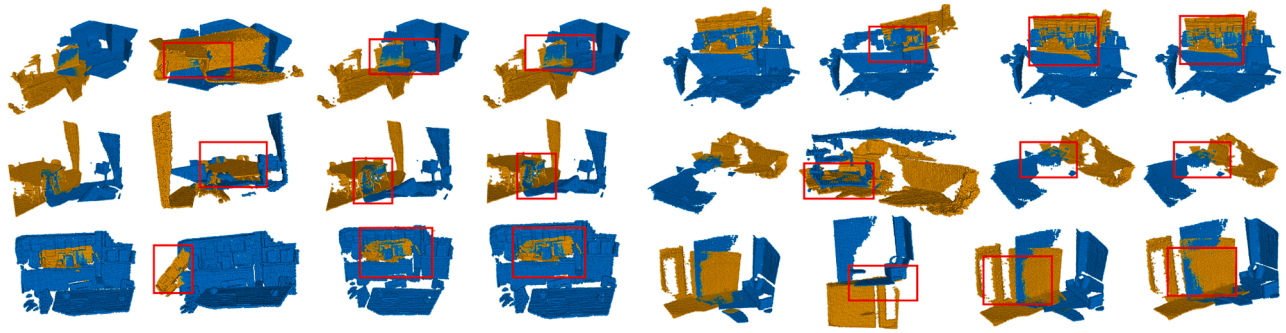

Fig. 8. Comparison of the registration results on 3DLoMatch. Benefitting from the promising overlap ratio in the anchor region, both EfficientPEAL-3D and EfficientPEAL-2D effectively recognize the overlapping region and successfully register the point clouds. The registration performance of EfficientPEAL is comparable to that of PEAL, while the baseline GeoTransformer fails to register these scenarios.

Table 5
Pairwise registration accuracies on the ScanNet dataset. We report the rotation accuracy with different angles (i.e., 5°, 10°, and 45°), and the translation accuracy with different lengths (i.e., 5cm, 10cm, and 25cm).

Methods	Train set	Frame interval	Rotation Accuracy ↑			Error↓		Translation Accuracy ↑			Error↓	
			5°	10°	45°	Mean	Med	5cm	10cm	25cm	Mean	Med
ICP (Besl & Mckay, 1992)	-	20	31.7	55.6	99.6	10.4	8.8	7.5	19.4	74.6	22.4	20.0
FPFH (Rusu et al., 2009)	-	20	34.1	64.0	90.3	20.6	7.2	8.8	26.7	66.8	42.6	18.6
SIFT (Lowe, 2004) + RANSAC	-	20	55.2	75.7	89.2	-	-	17.7	44.5	79.8	-	-
SuperPoint (DeTone, Malisiewicz, & Rabinovich, 2018) + RANSAC	-	20	65.5	86.9	96.6	-	-	21.2	51.7	88.0	-	-
FCGF (Choy et al., 2019) + RANSAC	-	20	70.2	87.7	96.2	-	-	27.5	58.3	82.9	-	-
BYOC (El Banani & Johnson, 2021)	3DMatch	20	66.5	85.2	98.8	2.7	0.8	83.8	93.8	97.6	5.8	2.0
UR& R (El Banani et al., 2021)	3DMatch	20	87.6	93.7	98.8	3.8	1.1	67.5	83.8	94.6	8.5	3.0
GeoTransformer (Qin et al., 2022)	3DMatch	20	95.4	97.9	98.8	3.4	1.2	79.5	92.0	97.0	7.3	2.6
EfficientPEAL-2D	3DMatch	20	**97.8**	**99.6**	**99.9**	**1.2**	**1.0**	**85.0**	**97.4**	**99.6**	**3.0**	**2.0**
BYOC (El Banani & Johnson, 2021)	3DMatch	50	53.2	68.0	94.5	16.1	6.9	27.5	42.2	65.0	33.5	18.9
GeoTransformer (Qin et al., 2022)	3DMatch	50	77.5	84.3	89.6	17.6	2.1	48.8	69.9	80.9	33.1	5.2
EfficientPEAL-2D	3DMatch	50	**85.1**	**91.3**	**94.8**	**9.7**	**1.8**	**57.2**	**78.8**	**89.3**	**17.1**	**4.3**

ing, demonstrates a notable enhancement in registration performance for low-overlap scenarios. Both of our proposed models exhibit strong performance compared to the baseline method.

4.2. ScanNet

Datasets. We follow UR&R (El Banani et al., 2021) to evaluate our methods on the largest indoor RGB-D dataset ScanNet (Dai et al., 2017). As in El Banani et al. (2021), we train our models on 3DMatch (Zeng et al., 2017) RGB-D dataset and evaluate them on ScanNet (Dai et al., 2017). The ScanNet dataset provided RGB-D videos with annotated poses for 1,513 scenes, while the RGB-D dataset of 3DMatch contains 101 scenes. We use the same setting as UR&R to split training/validation/test data for both 3DMatch and ScanNet datasets.

Training details. We train our models using the 3D Match dataset and evaluate their performance on ScanNet. For training, we randomly select 20,000 pairs from the 3DMatch dataset. The model is trained for 200,000 iterations using a learning rate of 1e-4. The training process is conducted on a machine equipped with a single NVIDIA 3090 GPU. We employ the Adam Optimizer with an epsilon value of 1e-4 and a momentum of 0.9.

Metrics. We use the evaluation metrics including the rotation error and translation error as in El Banani et al. (2021); El Banani and Johnson (2021). Furthermore, we also report the registration accuracy, which refers to the accuracy within three different rotation angle thresholds, and translation accuracy within three length thresholds.

Experimental results. We report the registration results in Table 5, it is clear that the supervised approaches (EfficientPEAL-2D and GeoTransformer) outperform all the other methods by a large margin, both of the two methods achieve a significant improvement in the 5° rotation accuracy and 5cm translation accuracy settings. Notably, our proposed method demonstrates a 2.4% enhancement on the 5° rotation accuracy and improves by 5.5% on 5cm translation accuracy, showing robust improvements compared to the baseline method.

To further evaluate the effectiveness of our methods, we propose conducting experiments in a more challenging setting: low-overlap RGB-D point cloud registration. We sample the RGB-D image pairs using a more sparse frame interval (eg. from 20 to 50), resulting in a lower overlapping region. We compare our method with the baseline methods and report the results at the bottom of Table 5. As shown in Table 5, when the frame interval increases from 20 to 50, all the baseline methods exhibit a significant performance drop. This is mainly due to the dramatic reduction in the overlap ratio between point clouds as the frame interval grows. In such cases, existing methods are severely affected by interference from a large range of non-overlapping regions, making it difficult to establish robust, consistent feature descriptions across frames. Consequently, a large number of mismatches occur in the extracted correspondences, leading to degraded registration accuracy. EfficientPEAL-2D first employs a highly efficient and robust image matching algorithm to predict correspondences between RGB images. These predicted matches are then projected onto point clouds through a projection module, yielding estimated overlapping points. Based on these predicted overlapping points, EfficientPEAL-2D constructs local geometric self-attention at the superpoint level. By explicitly restricting the attention scope to interactions between input points and the anchor superpoints within predicted overlapping regions, our approach effectively suppresses the interference from non-overlapping areas. Furthermore, benefiting from localized attention modeling and reliable 2D overlap priors, EfficientPEAL-2D significantly enhances both the geometric consistency and discriminability of the learned features. At the same time, it substantially reduces computational overhead, providing a strong foundation for building an efficient and scalable point cloud registration framework. As a result, the improvement is more prominent with lower overlap. Our method outperforms GeoTransformer (Qin et al., 2022) by 7.6% and 8.4% in terms of 5° rotation accuracy and 5cm accuracy, respectively.

Table 6
Comparison of computation complexity and 5° rotation accuracy on ScanNet, prior time represents the time of predicting overlap prior.

Models	Comp.	GPU (GB)	Acc. (%)	Prior (s)	Infer. (s)	Total (s)
GeoTransformer (Qin et al., 2022)	$O(N^2)$	9.46	95.4	-	0.172	0.172
EfficientPEAL-2D	$O(k_1 N^2)$	4.54	97.8	0.05	0.125	0.175

To assess the computational complexity of our proposed method, we compare the inference time and present the results in Table 6. We calculate the average ratio between the size of anchor superpoints and the size of input superpoints on the ScanNet dataset, with $k_1 = 0.19$. As indicated in the table, our proposed method, EfficientPEAL-2D , exhibits a 2.4% improvement in registration rotation accuracy compared to GeoTransformer (Qin et al., 2022). Additionally, it effectively reduces the complexity of self-attention by more than 80%. The memory occupancy of our method is only approximately 48% of that required by GeoTransformer.

In conclusion, the main advantages of EfficientPEAL-2D can be summarized in two aspects: First, it doesn't rely on accurate 2D pixel-to-pixel (point-to-point) correspondences, but rather utilizes a coarse patch-wise overlap prior. Second, PEAL-2D significantly reduces the computational complexity of the Transformer and reduces the memory footprint.

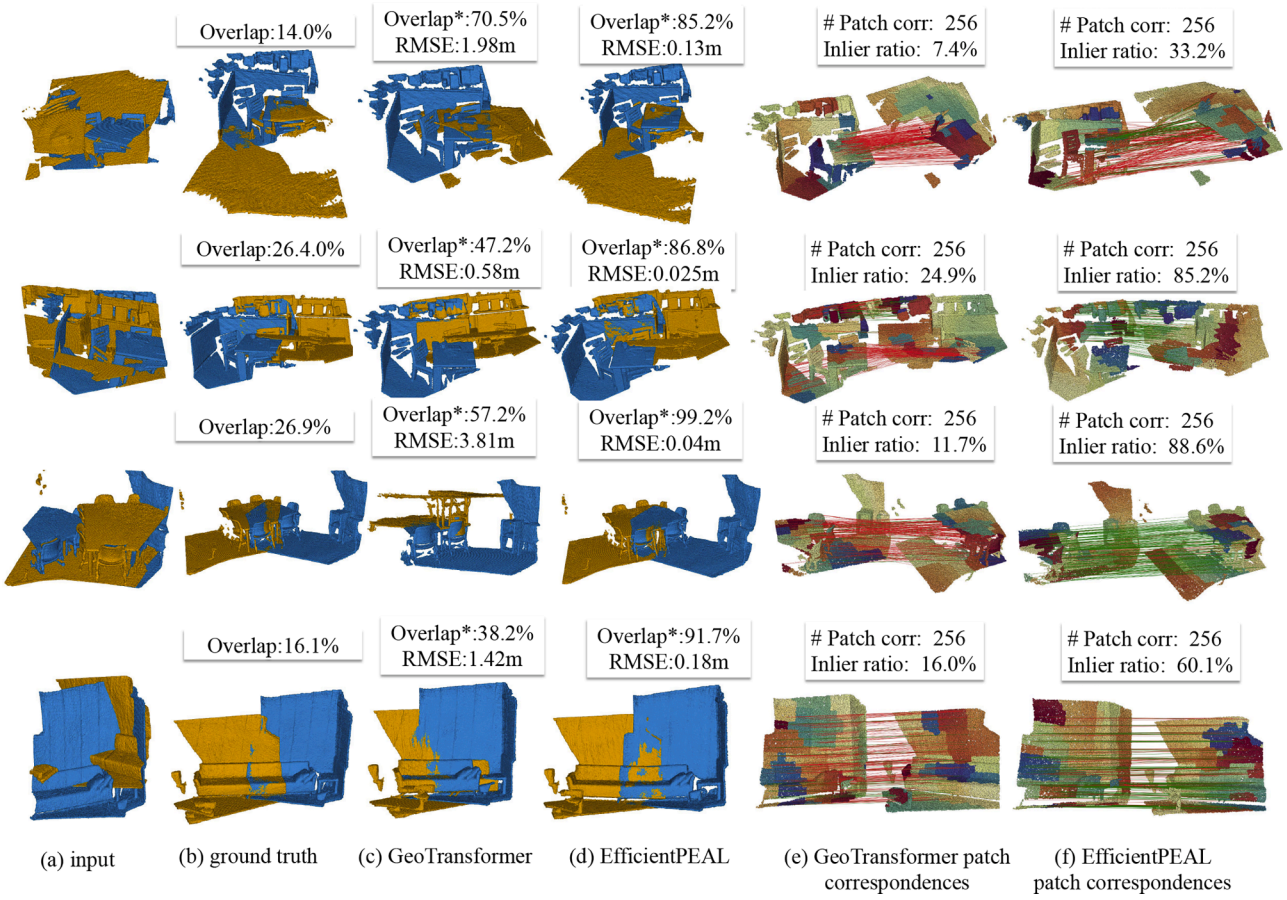

Fig. 9. Registration results of the GeoTransformer and EfficientPEAL-3D on 3DLoMatch. The overlapping region estimated by GeoTransformer is used as our 3D overlap prior (anchor region). Here, in the 2^{nd} column, "Overlap" represents the overlap ratio of input point clouds, and in 3^{rd} and 4^{th} columns, "Overlap*" represents the overlap ratio of anchor region. Our approach infers much more inlier superpoint matches benefitting from the improved overlap ratio in the anchor region, significantly enhancing the registration performance. We use T-SNE to visualize superpoint (patch) features.

4.3. Ablation studies

In this section, we ablate the different setups for the proposed modules in our method. In practice, we observe that the local-to-global registration (LGR) (Qin et al., 2022) estimator demonstrates greater stability and faster performance compared to the RANSAC estimator, while achieving comparable results. Therefore, we conduct our ablation experiments based on the LGR estimator.

How does 3D prior work? We first analyze the overlap ratio of GeoTransformer estimated overlapping region (3D prior or anchor region) to observe these *partial-aligned* scenarios. We compute the overlap ratio of 3D prior in cases where GeoTransformer fails to register the two point clouds. The distribution of overlap ratios is shown in Table 7, we discover that over 36% of the 3D prior exhibit an overlap ratio of 0.5+, while the original input overlap ratio is all ≤ 0.3 (Huang et al., 2021). This enhancement in the overlap ratio provides the potential to improve subsequent registration, which is the primary reason why EfficientPEAL-3D (EfficientPEAL-3D) outperforms GeoTransformer by a significant margin. We show some *partial-aligned* scenes in the 3^{rd} (c) column in Fig. 9, here, "Overlap*" represents the overlap ratio of anchor region.

Comparision with 2D-based method. Then we compare our method with another point cloud registration method, PCR-CG (Zhang et al., 2022), which also utilizes 2D prior. PCR-CG projects the deep features learned from color images into the geometry representation, showing consistent improvements when combined with different 3D point cloud registration methods. However, projecting deep features from images can be expensive and lead to information losses due to indirectness. In Table 8, it can be observed that regardless of whether combined with Predator or GeoTransformer, our method outperforms PCR-CG comprehensively in terms of registration recall, memory occupancy, and inference time, demonstrating its superiority.

Table 7
The overlap ratio of 3D prior when GeoTransformer fails on 3DLoMatch.

	Overlap ratio*			
Prior	0-0.1	0.1-0.3	0.3-0.5	0.5-1
3D prior	40.7%	12.9%	10.2%	36.1%

Table 8
Comparison with PCR-CG on 3DLoMatch.

Models	Comp.	GPU (GB)	RR(%)	Infer.(s)
Predator (Huang et al., 2021) + PCR-CG	$O(N^2) + O(M)$	12.3	66.3	0.563
Predator (Huang et al., 2021) + EfficientPEAL-2D	$O(k_2N^2)$	**7.26**	**72.3**	**0.422**
GeoTransformer (Qin et al., 2022) + PCR-CG	$O(N^2) + O(M)$	9.58	76.5	0.357
GeoTransformer (Qin et al., 2022) + EfficientPEAL-2D	$O(k_2N^2)$	**3.31**	**77.9**	**0.152**

Table 9
Ablation experiments of EfficientPEAL(hybrid prior) about the points threshold on 3DLoMatch.

Models	Points threshold	Registration Recall(%)
GeoTransformer (Qin et al., 2022)	-	74.0
EfficientPEAL-2D	0	80.3
EfficientPEAL(hybrid prior)	100	82.7
EfficientPEAL(hybrid prior)	200	**83.2**
EfficientPEAL(hybrid prior)	300	83.1

Hybrid prior. Although the 2D prior brings significant performance improvements, it may not always be reliable in scenarios with large viewpoint or illumination variations. To address this challenge, we introduce a hybrid 2D-3D prior. The intuitive idea is to use the 2D prior when it is deemed reliable; otherwise, we fall back on the 3D prior. The reliability of the 2D prior can be naively estimated based on the number of 2D matching pairs produced by SuperGlue, a higher number indicates more accurate 2D matching. Since these 2D correspondences are ultimately projected onto the 3D point cloud to form predicted overlapping points, we introduce a threshold parameter, denoted as δ (e.g., the number of corresponding overlapping points $> \delta$), to determine whether to use the 2D prior. If the number of points falls below the threshold, we replace the 2D prior with the 3D prior. In our experiments, we evaluate the impact of this threshold on registration performance, as shown in Table 9. Notably, with the hybrid prior, our method achieves an impressive registration recall of 83.2%, outperforming GeoTransformer by a significant margin of **9.2%**.

Quantitative results. We visualize the side by side comparison for EfficientPEAL and GeoTransformer (Qin et al., 2022) in Fig. 9. Our explicit attention-learning fashion significantly improves feature representation and helps to infer superpoint (patch) matches in extreme low-overlap scenarios.

KITTI dataset. Despite designing for low-overlap registration, our method also demonstrates effectiveness for high-overlap point clouds. We conduct experiments on the KITTI dataset (Geiger et al., 2012) and report the results in Table 10. As shown in the table, we discover that the performance is getting saturated. Considering GeoTransformer (Qin et al., 2022) as the current state-of-the-art method in both registration performance and inference speed, our primary focus with EfficientPEAL is achieving faster registration and using less memory footprint. By randomly selecting a certain ratio of input superpoints as a prior, our method EfficientPEAL(random prior) substantially reduces the complexity of self-attention while maintaining comparable performance to GeoTransformer. Moreover, we further investigate the impact of different proportions of randomly sampled priors on both efficiency and registration performance, as shown in Table 11. The results demonstrate that combining random anchor priors with the local geometric attention mechanism significantly improves efficiency. Notably, increasing the anchor ratio from 0.1 to 0.5 does not yield a substantial improvement in registration accuracy, while the inference time increases noticeably. This indicates that an appropriate anchor ratio can effectively balance accuracy and efficiency. In contrast, further increasing the number of anchors introduces unnecessary computational overhead without significant accuracy gains.

Table 10
Registration results on KITTI odometry.

Models	Estimators	RTE (cm)	RRE(°)	RR(%)
SpinNet (Ao et al., 2021)	RANSAC-50K	9.9	0.47	99.1
Predator (Huang et al., 2021)	RANSAC-50K	6.8	0.27	99.8
CoFiNet (Yu et al., 2021)	RANSAC-50K	8.2	0.41	99.8
GeoTransformer (Qin et al., 2022)	LGR	6.8	0.24	99.8
EfficientPEAL(random prior)	LGR	6.8	0.24	99.8

Table 11
Comparison of computation complexity on KITTI odometry. EfficientPEAL(random prior) shows a clear improvement in inference time.

Models	Ratio	GPU (GB)	Infer. (s)	RTE (cm)	RRE (°)	RR (%)
GeoTransformer (Qin et al., 2022)	-	10.42	0.139	6.8	0.24	99.8
EfficientPEAL(random prior)	0.1	6.35	0.129	6.8	0.24	99.8
EfficientPEAL(random prior)	0.3	6.45	0.131	6.8	0.24	99.8
EfficientPEAL(random prior)	0.5	6.94	0.133	6.8	0.24	99.8

Further discussion about EfficientPEAL . EfficientPEAL mitigates the computational cost of processing high-resolution point clouds to some extent by modeling local geometric self-attention. However, since the full EfficientPEAL architecture combines a KPConv-based convolutional network with attention mechanisms, both components may become computational bottlenecks when applied to large-scale or higher-resolution point clouds. To address the potential bottleneck posed by the KPConv network, we plan to explore the following enhancement: constructing a local reference frame for each query point within the KPConv framework to avoid global feature extraction and reduce computational cost. For the local geometric attention module, we will investigate more efficient alternatives, such as linear attention or fast attention mechanisms. We believe these strategies will significantly enhance the scalability of EfficientPEAL for large-scale, high-resolution point cloud tasks, and we will further explore and optimize them in future work.

5. Conclusion

We present EfficientPEAL, an efficient learning model for enhancing the pose registration of point clouds. It starts with the existing 2D/3D matching methods to obtain overlap prior. Then it leverages a local geometric self-attention module, which explicitly captures local geometric feature correlations between the input and putative overlapping region. Our approach significantly reduces the GPU memory usage and inference time of PEAL, while achieving comparable performance. Consistent and significant improvements across multiple test datasets effectively validate the efficacy of the proposed method. EfficientPEAL demonstrates robust and efficient point cloud registration under challenging conditions, such as low overlap and weak geometric regions. In comparison to PEAL, EfficientPEAL exhibits greater suitability for deployment in robotics and augmented reality applications. Its well-balanced trade-off between computational efficiency and accuracy positions it as a compelling solution for SLAM, object localization, and scene reconstruction tasks, particularly in dynamic or resource-constrained environments.

CRediT authorship contribution statement

Junle Yu: Conceptualization, Methodology, Software, Investigation, Data curation, Writing - original draft; **Wenhui Zhou:** Conceptualization, Methodology, Validation, Project administration; **Zhehao Shen:** Visualization, Investigation, Data curation, Software; **Yongwei Miao:** Conceptualization, Writing - review & editing, Project administration, Supervision.

Data availability

Data will be made available on request.

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.

Acknowledgement

We would like to thank the anonymous reviewers for their helpful and valuable comments. This work was partially supported by the Joint Funds of the National Natural Science Foundation of China under Grant No. U24A20249, and the Zhejiang Provincial Natural Science Foundation of China under Grant No. LZ23F020002, LTY22F020001, LY21F010007.

References

Ao, S., Guo, Y., Hu, Q., Yang, B., Markham, A., & Chen, Z. (2022). You only train once: Learning general and distinctive 3d local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(3), 3949–3967.

Ao, S., Hu, Q., Yang, B., Markham, A., & Guo, Y. (2021). Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11753–11762).

Aoki, Y., Goforth, H., Srivatsan, R. A., & Lucey, S. (2019). Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7163–7172).

Bai, X., Luo, Z., Zhou, L., Chen, H., Li, L., Hu, Z., Fu, H., & Tai, C.-L. (2021). PointDSC: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15859–15869).

Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., & Tai, C.-L. (2020). 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* (pp. 6359–6367).

Besl, P. J., & Mckay, H. D. (1992). A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *14*(2), 239–256.

Besl, P. J., & McKay, N. D. (1992). Method for registration of 3-D shapes. In *Sensor fusion IV: Control paradigms and data structures* (pp. 586–606). Spie (*vol. 1611*).

Chen, Y., & Medioni, G. (1992). Object modelling by registration of multiple range images. *Image and Vision Computing*, *10*(3), 145–155.

Chen, Z., Ren, Y., Zhang, T., Dang, Z., Tao, W., Süsstrunk, S., & Salzmann, M. (2023). Diffusionpcr: Diffusion models for robust multi-step point cloud registration. *arXiv preprint arXiv:2312.03053*.

Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., & Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1290–1299).

Choy, C., Dong, W., & Koltun, V. (2020). Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2514–2523).

Choy, C., Park, J., & Koltun, V. (2019). Fully convolutional geometric features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8958–8966).

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., & Nießner, M. (2017). Scannet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5828–5839).

Deng, H., Birdal, T., & Ilic, S. (2018). Ppf-foldnet: Unsupervised learning of rotation invariant 3D local descriptors. In *Proceedings of European conference on computer vision* (pp. 602–618).

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018). Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshop* (pp. 224–236).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

El Banani, M., Gao, L., & Johnson, J. (2021a). UnsupervisedR&R: Unsupervised point cloud registration via differentiable rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7129–7139).

El Banani, M., & Johnson, J. (2021). Bootstrap your own correspondences. In *Proceedings of international conference on computer vision* (pp. 6433–6442).

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381–395.

Fung, S., Lu, X., de Silva, E. D., Pan, W., Liu, X., & Li, H. (2025). Semreg: Semantics constrained point cloud registration. In *Proceedings of European conference on computer vision* (pp. 293–310).

Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3354–3361).

Gojcic, Z., Zhou, C., Wegner, J. D., & Wieser, A. (2019). The perfect match: 3D point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5545–5554).

Guo, Y., Sohel, F., Bennamoun, M., Lu, M., & Wan, J. (2013). Rotational projection statistics for 3d local surface description and object recognition. *International Journal of Computer Vision*, *105*, 63–86.

Hou, J., Dai, A., & Nießner, M. (2019). 3D-SIS: 3D semantic instance segmentation of RGB-D Scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4421-4430).

Hou, J., Dai, A., & Nießner, M. (2020). RevealNet: Seeing behind objects in RGB-D scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2098–2107).

Hou, J., Xie, S., Graham, B., Dai, A., & Nießner, M. (2021). Pri3d: Can 3D priors help 2d representation learning? *arXiv preprint arXiv:2104.11225*.

Huang, S., Gojcic, Z., Usvyatsov, M., Wieser, A., & Schindler, K. (2021). Predator: Registration of 3D point clouds with low overlap. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4267–4276).

Jin, S., Armeni, I., Pollefeys, M., & Barath, D. (2024). Multiway point cloud mosaicking with diffusion and global optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 20838–20849).

Johnson, A. E., & Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *21*(5), 433–449.

Li, Y., & Harada, T. (2022). Lepard: Learning partial point cloud matching in rigid and deformable scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5554–5564).

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Qi, C. R., Chen, X., Litany, O., & Guibas, L. J. (2020). Imvotenet: Boosting 3D object detection in point clouds with image votes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4404–4413).

Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Ilic, S., Hu, D., & Xu, K. (2023). Geotransformer: Fast and robust point cloud registration with geometric transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(8), 9806–9821.

Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., & Xu, K. (2022). Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11143–11152).

Rusinkiewicz, S., & Levoy, M. (2001). Efficient variants of the ICP algorithm. In *Proceedings third international conference on 3-d digital imaging and modeling* (pp. 145–152).

Rusu, R. B., Blodow, N., & Beetz, M. (2009). Fast point feature histograms (FPFH) for 3d registration. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 3212–3217).

Rusu, R. B., Blodow, N., Marton, Z. C., & Beetz, M. (2008). Aligning point cloud views using persistent feature histograms. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (pp. 3384–3391).

Sarlin, P.-E., DeTone, D., Malisiewicz, T., & Rabinovich, A. (2020). Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4938–4947).

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., & Guibas, L. J. (2019). KPConv: Flexible and deformable convolution for point clouds. In *Proceedings of the international conference on computer vision* (pp. 6411–6420).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems (NeurIPS)* (pp. 6000–6010).

Wang, H., Liu, Y., Dong, Z., & Wang, W. (2022). You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In *Proceedings of the ACM International conference on multimedia* (pp. 1630–1641).

Yang, J., Li, C., Zhang, P., Dai, X., Xiao, B., Yuan, L., & Gao, J. (2021). Focal self-attention for local-global interactions in vision transformers. *arXiv preprint arXiv:2107.00641*.

Yew, Z. J., & Lee, G. H. (2022). Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6677–6686).

Yu, H., Li, F., Saleh, M., Busam, B., & Ilic, S. (2021). Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. In *Advances in neural information processing systems (NeurIPS)*, *34*, 23872–23884.

Yu, H., Qin, Z., Hou, J., Saleh, M., Li, D., Busam, B., & Ilic, S. (2023a). Rotation-invariant transformer for point cloud matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5384–5393).

Yu, J., Ren, L., Zhang, Y., Zhou, W., Lin, L., & Dai, G. (2023b). 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* (pp.17702–17711).

Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., & Funkhouser, T. (2017). 3DMatch: Learning local geometric descriptors from RGB-D reconstructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1802–1811).

Zhang, Y., Yu, J., Huang, X., Zhou, W., & Hou, J. (2022). Pcr-cg: Point cloud registration via deep explicit color and geometry. In *Proceedings of European conference on computer vision* (pp. 443–459).

Zhou, W., Ren, L., Yu, J., Qu, N., & Dai, G. (2024). Boosting RGB-d point cloud registration via explicit position-aware geometric embedding. *IEEE Robotics and Automation Letters*, *9*(6), 5839–5846.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., & Dai, J. (2020). Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*.