

Equi-GSPR: Equivariant $SE(3)$ Graph Network Model for Sparse Point Cloud Registration

Xueyang Kang^{1,2,3}, Zhaoliang Luan^{2,4}, Kourosh Khoshelham³, and
Bing Wang²*

¹ Faculty of Electrical Engineering, KU Leuven

² Spatial Intelligence Group, The Hong Kong Polytechnic University

³ Faculty of Engineering and IT, The University of Melbourne

⁴ IoTUS Lab, Queen Mary University of London

alex.kang@kuleuven.com, z.luan@qmul.ac.uk, k.khoshelham@unimelb.edu.au,
bingwang@polyu.edu.hk

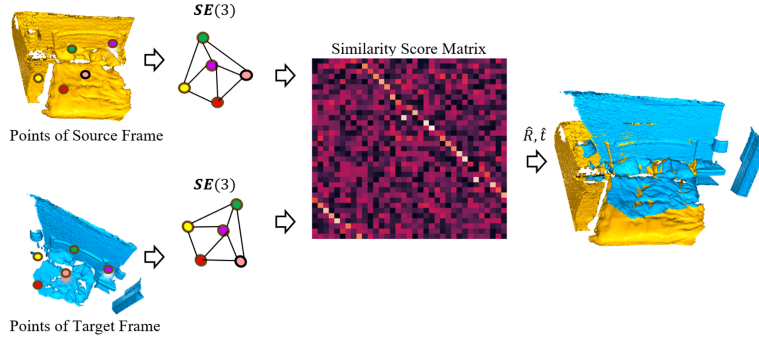


Fig. 1: The registration model converts the sparse point descriptors of the source and target frames into an equivariant graph feature representation, respectively. Then the $SE(3)$ equivariant graph features are used for the similarity score calculation. The matched features are then decoded into the relative transform to align the two scans.

Abstract. Point cloud registration is a foundational task for 3D alignment and reconstruction applications. While both traditional and learning-based registration approaches have succeeded, leveraging the intrinsic symmetry of point cloud data, including rotation equivariance, has received insufficient attention. This prohibits the model from learning effectively, resulting in a requirement for more training data and increased model complexity. To address these challenges, we propose a graph neural network model embedded with a local Spherical Euclidean 3D equivariance property through $SE(3)$ message passing based propagation. Our model is composed mainly of a descriptor module, equivariant graph layers, match similarity, and the final regression layers. Such modular design enables us to utilize sparsely sampled input points and initialize the descriptor by self-trained or pre-trained geometric feature descriptors easily. Experiments conducted on the 3DMatch and KITTI datasets exhibit the compelling and robust performance of our model compared to state-of-the-art approaches, while the model complexity remains relatively low at the same time.

* Corresponding author

Keywords: Equivariance · $\mathbf{SE}(3)$ · Graph Network Model · Point Cloud Registration · Feature Descriptor · Similarity.

1 Introduction

The registration of point clouds typically involves formulating robust geometric feature descriptors and a subsequent complex matching process to predict feature correspondences [8]. However, these correspondences established from raw point cloud often exhibit a high outlier-to-inlier ratio, leading to significant registration errors or complete failures. To enhance the robustness of registration processes, PointDSC [2] explicitly calculates local feature spatial consistency and evaluates pairwise 3D geometric feature descriptor similarity across two frames [10, 46] to eliminate outliers from the alignment optimization process. Other approaches like Deep Global Registration (DGR) [9] treat correspondence prediction as a classification problem, utilizing concatenated coordinates of input point cloud pairs and employing a differentiable optimizer for pose refinement. Despite the effectiveness of these models on public datasets, their training requires accurate correspondence supervision, necessitating a complex point-to-point search process that is particularly vulnerable to numerous outliers.

Geometric feature descriptors, derived from keypoint neighborhoods of a specified range, often overlook the underlying geometric topology of the data, such as the global connectivity among points. This oversight results in feature descriptors lacking $\mathbf{SE}(3)$ rotation equivariance, thereby impeding efficient and robust learning of rotation-equivariant and invariant features. The recently introduced RoReg [44] model employs a rotation-guided detector to enhance rotation coherence matching and integrates it with RANSAC for pose estimation. However, it suffers from high computational demand and reduced processing speed. This highlights the need for more efficient rotation-equivariant model architectures to significantly enhance registration performance.

To address these challenges, we introduce a novel approach that leverages a graph convolution-based model to jointly learn $\mathbf{SE}(3)$ equivariant features, starting with feature descriptors extracted from sparsely sampled points across two frames. Our proposed $\mathbf{SE}(3)$ equivariant graph network model, aimed at sparse point cloud registration, is depicted in Fig. 1. Unlike Transformer and CNN-based models, our graph architecture captures both the topology and geometric features of point clouds, similar to other proposed geometric descriptors [40, 53], facilitating the learning of fine-grained rigid rotation-equivariant feature representations for more robust and coherent point cloud registration through data symmetry. The primary contributions of our study are the following:

- Introduction of an equivariant graph model to facilitate neighbor feature aggregation and $\mathbf{SE}(3)$ equivariant coordinate embedding from either learned geometric descriptors for point cloud registration.
- Implementation of a novel matching approach within the implicit feature space, based on similarity evaluation and Low-Rank Feature Transformation

- (LRFT), eliminating the need for explicit point correspondence supervision and exhausting search.
- Development of a specific matrix rank-based regularizer to enable the model to automatically identify and mitigate the impact of correspondence outliers, enhancing the robustness of the registration process.

2 Related Work

The concept of equivariant properties can be embedded in the layers of Convolutional Neural Networks (CNNs) to depict $\mathbf{SO}(2)$ group characteristics, as initially proposed by Cohen [11], and later extended to encompass arbitrary continuous input [16]. Another area of study focuses on equivariance representation by employing steerable kernel filters [12, 27, 41, 47–49] for equivariance learning. For more intricate tasks involving $\mathbf{SO}(3)$, techniques such as Vector Neurons [13] and Tensor Field Networks [42] can be viewed as implementations of the capsule network model [50], transitioning from scalar values to vectors. Following the introduction of the Transformer model, Lie-group-based Transformer models [18, 19, 26] have been developed to capture equivariance through attention mechanisms. Moreover, to address the intricate equivariance inherent in the input data, equivariant(n) graph neural networks [14, 28, 38] are utilized to learn equivariant features for dynamic and complex issues. As $\mathbf{SE}(3)$ features are maintained through message passing [5] within the graph model, these equivariant models exhibit considerable potential in addressing many longstanding challenges, such as predicting molecule structures [39] or quantum structures [21], as well as particle dynamic flow physics [4, 29]. Some studies have explored the application of equivariant models in various point cloud tasks, including 3D detection [40], 3D point classification [53], point cloud-based place recognition [30], 3D shape point registration [7, 55], and 3D shape reconstruction [6]. These applications underscore the learning efficiency gained from the leveraging of the intrinsic symmetries in input data.

Despite the prevalence of equivariant models in the microscopic realm [18, 39], the application of such models for tasks like multi-view 3D reconstruction [23] or other intricate 3D challenges remains under-explored. A fundamental component of 3D reconstruction involves point cloud registration. Many conventional methods employ linear-algebra-based optimization techniques for iterative point cloud registration, such as point-to-plane registration [33, 34], LOAM [45], and its variation F-LOAM [45]. In recent years, there has been a rise in deep learning models for registration purposes, relying on representative feature descriptors [10, 46] or precise correspondence establishment between descriptors, like deep global registration [9]. To enhance registration accuracy, some studies focus on developing more resilient descriptors like rotation-equivariant descriptors [3, 43, 44] for subsequent correspondence matching or incorporating $\mathbf{SO}(2)$ rotation equivariance into the registration framework using cylindrical convolution, as in Spinnet [1]. Other research works concentrate on optimizing correspondence search explicitly, for instance, Stickypillars utilizes optimal transport

for matching, and PointDSC [2] employs spectral matching to eliminate outliers from raw correspondences. In contrast to conventional models, works in [36, 54] excel in rapid registration performance. Moreover, some end-to-end models improve overall performance from feature descriptor learning to feature association in a differentiable way, like 3D RegNet [32], or correspondence-free registration aimed at streamlining the 3D point cloud registration [56]. Notably, Banani et al. [15] introduce an unsupervised model for point cloud registration using differentiable rendering, while Predator [25] demonstrates registration capabilities even in applications with low input point cloud overlap and numerous outliers.

3 Method

Our registration process begins by extracting feature descriptors from downsampled point clouds. Equivariance is integrated into the features through equivariant graph convolution layers. Subsequently, the number of features in the pairwise graph is aggregated to a smaller number via Low-Rank based constraint. Finally, the similarity between the pairwise features of the two frames is calculated for relative transform prediction. A detailed illustration of the model is shown in Fig. 2. The input to our model consists of N points $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in N \times \mathbb{R}^3$ from the source frame, and N points $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in N \times \mathbb{R}^3$ from the target frame, where $x_i \in \mathbb{R}^3$ and $y_j \in \mathbb{R}^3$ form a correspondence (i, j) . It is important to note, for ease of subsequent similarity search, that the coordinates of points in each frame are rearranged in descending order based on the ray length $\|\mathbf{r}(t)\|^2$ from the point position to the sensor frame center \mathbf{o}_s . For numerical stability during training, the source scan is normalized to a canonical frame, and the target scan is transformed relative to the source frame, allowing the model to predict the relative transformation from source to target.

3.1 Feature Descriptor

We incorporate geometric details of nearby interest points into our graph model using feature descriptors. To extract these descriptors, we can reuse available pre-trained point-based descriptors or train a shallow Multi-Layer Perceptron (MLP) module with l_1 layers prior to the model in an end-to-end manner, inspired by PointNet++ [35]. The feature representation of point i in the next layer $l + 1$ is calculated by averaging the output of the mapping function $h(\cdot)$ applied to the relative positional coordinates of point i and neighboring point $k \in \mathcal{N}(i)$ (n points), along with the hidden feature $h_k^{l_1}$ from the prior layer l_1 .

$$\mathbf{h}_i^{l_1+1} = \frac{1}{n} \sum_{k \in \mathcal{N}(i)} f_h(\mathbf{h}_k^{l_1}, \mathbf{x}_k - \mathbf{x}_i). \quad (1)$$

3.2 Equivariant Graph Network Model

By utilizing the equivariant graph representation (l_2 layers) as introduced by Satorras et al. [38], we can enhance the receptive field and representation for

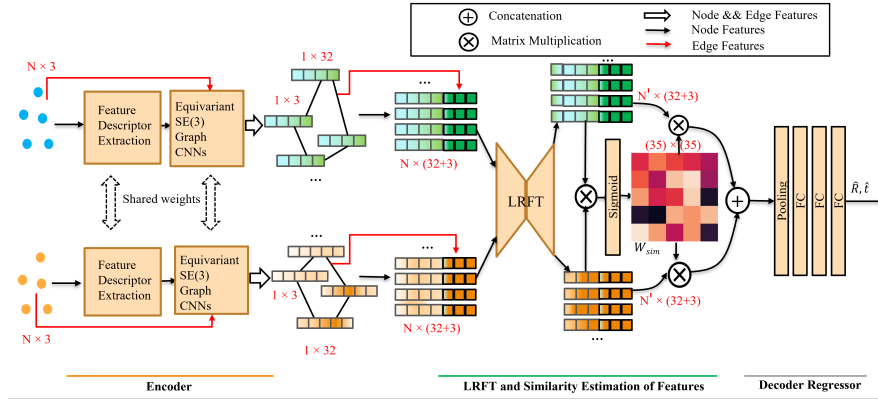


Fig. 2: The registration model consists of an encoder, a feature match block, and a decoder. Pointwise feature descriptors are extracted from the source and target scan points, passed through equivariant graph layers, and combined with coordinate embeddings to form a row-major order matrix. Next, the feature matrices from the source and target frames are compressed using MLPs-based Low-Rank Feature Transformation (LRFT). The aggregated features are used to create a similarity map through dot product of feature descriptors. In the decoder, features are weighted by similarity scores, then concatenated, and processed through pooling and fully connected layers to predict relative translation t_j^i and quaternion q_j^i .

feature descriptors of interest points by incorporating **SE**(3) equivariant properties via graph feature aggregation. Our method leverages graph-based message passing techniques [22] to propagate **SE**(3) equi-features. For the construction of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices \mathcal{V} and \mathcal{E} edges. The individual hidden point descriptors $\mathbf{h}_i^{l_2} \in \mathbb{R}^{32}$ and the point coordinate embedding $\mathbf{x}_i^{l_2} \in \mathbb{R}^3$ at layer l_2 are treated as the node and edge features, respectively. The graph convolutional layer updates the edge equi-message $\mathbf{m}_{ik} \in \mathbb{R}^{3 \times 3}$, node hidden feature $\mathbf{h}_i^{l_2} \in \mathbb{R}^{32}$, and coordinate embedding $\mathbf{x}_i^{l_2} \in \mathbb{R}^3$ at each equivariant layer.

$$\mathbf{m}_{ik} = \phi_m(\mathbf{h}_i^{l_2}, \mathbf{h}_k^{l_2}, \|\mathbf{x}_k^{l_2} - \mathbf{x}_i^{l_2}\|^{\frac{1}{2}}), \quad (2)$$

$$\mathbf{x}_i^{l_2+1} = \mathbf{x}_i^{l_2} + C \sum_{k \in \mathcal{N}(i)} \exp(\mathbf{x}_k^{l_2} - \mathbf{x}_i^{l_2}) \phi_x(\text{proj}_{\mathcal{F}_{ik}} \mathbf{m}_{ik}), \quad (3)$$

$$\mathbf{h}_i^{l_2+1} = \phi_h(\mathbf{h}_i^{l_2}, \sum_{k \in \mathcal{N}(i)} (\text{proj}_{\mathcal{F}_{ik}} \mathbf{m}_{ik})), \quad (4)$$

where ϕ_m , ϕ_x , and ϕ_h represent 1D convolutional layers for the message, coordinate embedding, and hidden feature update, respectively. The normalizing factor C is applied to the exponentially weighted sum of mapped equi-message in Eq. (3). Additionally, a neighbouring search of $\mathbf{x}_i^{l_2}$ within a specific radius is conducted to find the $\mathcal{N}(i)$ neighbouring feature descriptors for edge establishments, and this is used to prevent information overflow by confining the exchange of information within a local context, thereby reducing the complexity of the graph feature adjacency matrix from $O(n^2)$ to approximately $O(n)$. In

Eq. (3), the projection of \mathbf{m}_{ij} onto a locally equivariant frame ($\text{proj}_{\mathcal{F}_{ik}}(\cdot)$) helps to preserve the $\mathbf{SO}(3)$ feature invariance. The frame \mathcal{F}_{ik} is constructed using pairwise coordinate embeddings as outlined in ClofNet [14],

$$\mathcal{F}_{ik} = (\mathbf{a}_{ik}, \mathbf{b}_{ik}, \mathbf{c}_{ik}), \quad (5)$$

$$= \left(\frac{\mathbf{x}_i^l - \mathbf{x}_k^l}{\|\mathbf{x}_i^l - \mathbf{x}_k^l\|}, \frac{\mathbf{x}_i^l \times \mathbf{x}_k^l}{\|\mathbf{x}_i^l \times \mathbf{x}_k^l\|}, \frac{\mathbf{x}_i^l - \mathbf{x}_k^l}{\|\mathbf{x}_i^l - \mathbf{x}_k^l\|} \times \frac{\mathbf{x}_i^l \times \mathbf{x}_k^l}{\|\mathbf{x}_i^l \times \mathbf{x}_k^l\|} \right), \quad (6)$$

Consequently, the projection of \mathbf{m}_{ik} into $\hat{\mathbf{m}}_{ik}$ is formulated into the linear combination of axes of the local equi-frame scaled by the coefficients $(x_{ik}^{\mathbf{a}}, x_{ik}^{\mathbf{b}}, x_{ik}^{\mathbf{c}})$,

$$\text{proj}_{\mathcal{F}_{ik}} \mathbf{m}_{ik} = \hat{\mathbf{m}}_{ik} = x_{ik}^{\mathbf{a}} \mathbf{a}_{ik} + x_{ik}^{\mathbf{b}} \mathbf{b}_{ik} + x_{ik}^{\mathbf{c}} \mathbf{c}_{ik}. \quad (7)$$

The projection of edge message m_{ij} in Eq. (3) is performed in the local equi-frame (within bracket of Eq. (6)), to obtain projected message $\hat{\mathbf{m}}_{ik}$, while the scalar coefficients in Eq. (7) remain $\mathbf{SO}(3)$ invariant. Consequently, the sum of equi-projected message in Eq. (4) is still a vector-based sum, maintaining the equivariance upon integration into the hidden layer ϕ_h .

3.3 Low-Rank Feature Transformation

Inspired by LoRA of language model [24], our approach diverges by not requiring pre-trained weights for fine-tuning. We employ two stacked linear forward layers with low-rank constraints in the middle of model (Fig. 3a) to map feature descriptors to aggregated descriptors. We name it as Low-Rank Feature Transformation (LRFT). This design enhances similarity match reliability and computational efficiency by performing matches on aggregated descriptors with integrated neighboring information. Specifically, The motivation for employing LRFT is twofold: 1) Theoretically, low-rank constraints in linear layers capture essential feature correlations within descriptors, as demonstrated by the matrix low-rank theorem (refer to Appendix Sec.1.2), leading to more reliable similarity matches; 2) Practically, our LRFT improves computational efficiency by aggregating feature descriptors prior to similarity matching, and It also enhances low-rank learning efficiency by training parameters during the forward pass, eliminating the need for pre-trained weights.

After the final layer ((l_2^*) th) of the equivariant graph module, the output graph features consist of both node and edge features. During this stage, we preserve each graph node feature $\mathbf{h}_i^{l_2^*}, i \in N$ from the source frame and feature $\mathbf{h}_j^{l_2^*}, j \in N$ from the target frame of the last graph layer. Next, the node feature is combined with the mean coordinate embeddings $\hat{\mathbf{x}}_i^{l_2^*} = \frac{1}{n} \sum_{k \in \mathcal{N}(i)} \mathbf{x}_k^{l_2^*}$, where $\mathbf{x}_k^{l_2^*} \in \mathbb{R}^3$ is obtained from the edge embeddings (Eq. (3)) connected to the node feature $\mathbf{x}_i^{l_2^*}$. Consequently, matrices for the source frame $\mathbf{H}_{src} \in \mathbb{R}^{N \times 35}$ and target frame $\mathbf{H}_{tar} \in \mathbb{R}^{N \times 35}$ are created by stacking of node features $(\mathbf{h}_i^{l_2^*}, \hat{\mathbf{x}}_i^{l_2^*})$ and coordinate embeddings along the column respectively. Before computing

the feature similarity, the Low-Rank Feature Transformation (LRFT) technique is applied to compress the features into $\hat{\mathbf{H}}_{src} \in \mathbb{R}^{N' \times 35}$ and $\hat{\mathbf{H}}_{tar} \in \mathbb{R}^{N' \times 35}$ utilizing parameters of \mathbf{A} and \mathbf{B} mapping layers.

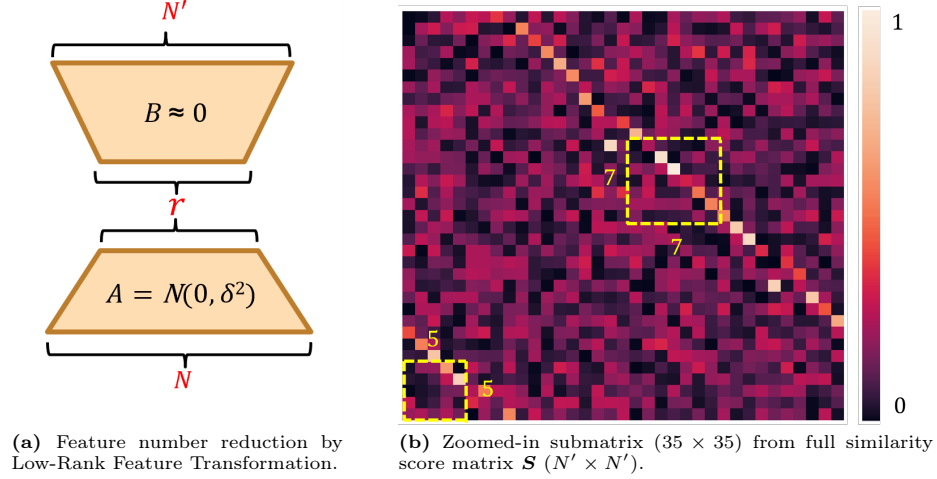


Fig. 3: Reducing the feature number through low-rank using MLP layers (a), and examining the similarity score matrix with submatrices for rank verification at bottom right (5×5) and center (7×7) of yellow dashed region as illustrated in subfigure (b).

$$\hat{\mathbf{H}}_{src}, \hat{\mathbf{H}}_{tar} = (\mathbf{A}\mathbf{B})^T(\mathbf{H}_{src}, \mathbf{H}_{tar}), \quad (8)$$

where \mathbf{A} is a $N \times r$ matrix and \mathbf{B} is $r \times N'$ matrix, and $\text{rank } r \ll \min(N, N')$, so that the number of node features are compressed into N' dimension after LRFT module mapping via multiplication $\mathbf{A}\mathbf{B}$, as depicted in the left of Fig. 3a. \mathbf{A} is initialized from Gaussian distribution with standard deviation $\delta = \sqrt{r}$, while \mathbf{B} is initialized with small constant close to zero.

The LRFT layers extract spatial context from neighboring feature descriptors through linear mapping, efficiently aggregating local information for decoder.

3.4 Similarity Calculation

Subsequently, we calculate the feature similarity score matrix (refer to Fig. 3b) for feature correspondence establishments using the compressed number of features after the LRFT module. This is achieved by computing the dot product $\langle \cdot \rangle$ of features as an element. Prior to the multiplication, the respective element features \mathbf{h}_i from $\hat{\mathbf{H}}_{src} \in \mathbb{R}^{N' \times 35}$ and \mathbf{h}_j from $\hat{\mathbf{H}}_{tar} \in \mathbb{R}^{N' \times 35}$ are first normalized to $\hat{\mathbf{h}}_i$ and $\hat{\mathbf{h}}_j$.

$$\mathbf{S}_{ij} = \langle \hat{\mathbf{h}}_i \cdot \hat{\mathbf{h}}_j \rangle, \quad (9)$$

which forms a square similarity matrix $\mathbf{S} \in \mathbb{R}^{N' \times N'}$, then subsequently normalized along each row to produce $\hat{\mathbf{S}} \in \mathbb{R}^{N' \times N'}$. We compute the determinant of $\hat{\mathbf{S}}$ by calculating trace to indicate whether the matrix rank has deficiency, which may arise from the presence of ambiguous correspondences. Accordingly, as per the match assignment rule, each row of the similarity matrix $\hat{\mathbf{S}}$ should contain a singular value close to one, depicted as a light-colored square in Fig. 3b. Subsequently, the similarity matrix is employed to project the feature matrices $\hat{\mathbf{H}}_{src}$ and $\hat{\mathbf{H}}_{tar}$ through multiplication by $\hat{\mathbf{S}}^T \hat{\mathbf{H}}_{src} \in \mathbb{R}^{N' \times 35}$ and $\hat{\mathbf{S}} \hat{\mathbf{H}}_{tar} \in \mathbb{R}^{N' \times 35}$, respectively. The resulting matrices are concatenated to facilitate subsequent pooling and mapping through fully-connected layers. Furthermore, a regularizer is used to enforce the rank of $\hat{\mathbf{S}}$ close to 35, ensuring that a submatrix $\hat{\mathbf{S}}'$ with rank 35 out of $\hat{\mathbf{S}}$ can be found,

$$\mathcal{L}_{Reg} = \left| (\text{Trace}(\hat{\mathbf{S}}^T \hat{\mathbf{S}}))^{\frac{1}{2}} - 35 \right|. \quad (10)$$

Additionally, to eliminate outlier feature correspondences, each matched pair element $\hat{\mathbf{S}}_{ij}$ undergoes verification through a submatrix full-rank check. This involves evaluating the determinant of a 7×7 submatrix centered at the feature element $\hat{\mathbf{S}}_{ij}$ or a 5×5 submatrix at the border element of the $\hat{\mathbf{S}}$ matrix (highlighted by the red dashed box in Fig. 3b). This verification process ensures local consistency in feature similarity matches, aiding in the identification of globally consistent and reliable match pattern search from the similarity matrix. Following verification, the final valid rank of the similarity matrix $\hat{\mathbf{S}}$ is established as $r \leq 35$, with any invalid assignment row $\hat{\mathbf{S}}_i$ zeroed out to mask the corresponding feature for subsequent computations. The upper limit of 35 for rank r is derived from the **Theorem: Rank**(\mathbf{AB}) $\leq \min(\mathbf{Rank}(\mathbf{A}), \mathbf{Rank}(\mathbf{B}))$. Please refer to the supplementary part for detailed proof the rank theorem.

3.5 Training Loss

The final layer predicts translation $\hat{\mathbf{t}}$ and rotation matrix $\hat{\mathbf{R}}$ in quaternion form. The ground-truth translation and rotation are denoted by \mathbf{t}^* and \mathbf{R}^* respectively.

$$\mathcal{L}_{total} = \mathcal{L}_{rot} + \mathcal{L}_{trans} + \beta \mathcal{L}_{Reg}, \quad (11)$$

β for regularizer is set to 0.05. The translation error (TE) and rotation error (RE) losses can be formulated as follows,

$$\mathcal{L}_{rot}(\hat{\mathbf{R}}) = \arccos \frac{\text{Trace}(\hat{\mathbf{R}}^T \mathbf{R}^*) - 1}{2}, \quad (12)$$

$$\mathcal{L}_{trans}(\hat{\mathbf{t}}) = \|\hat{\mathbf{t}} - \mathbf{t}^*\|^2. \quad (13)$$

The rotation error term \mathcal{L}_{rot} and translation error term \mathcal{L}_{trans} are measured in radians and meters, respectively. Given that the predicted transform is relative, from source to target frame, the scale of these transforms is typically in normal scale to avoid numerical stability issues in training.

4 Experiments

We evaluate the performance of the proposed model for point cloud registration in both indoor and outdoor environments. For indoor scenes, we utilize 3DMatch introduced by Zeng et al. [52]. The raw point clouds are uniformly downsampled to 1024 points through a voxel filter. For the outdoor evaluation, we select the KITTI dataset [20] with the same dataset split from the creators and follow the same downsampling process as in Choy *et al.*, [10]. We report both the qualitative and quantitative results of the proposed model. Furthermore, we offer an in-depth analysis of the effect of varying parameter configurations and the contribution of each component to enhancing the model’s performance. The computational efficiency of each model is presented in the metric table below.

Implementation Details. The key parameters of our model include the dimension of graph-relevant features and LRFT layers. Initially, the extracted feature descriptor dimension is 32 for subsequent graph learning. A critical aspect is the number of nearest neighbors for each node feature in the graph, set to 16 for constructing the graph using ball query for 3DMatch (ball radius at $0.3m$), while for KITTI, we employ kNN, selecting the nearest 16 points of query point to form a graph with 1024 nodes and 1024×16 edges. In graph learning, the node feature dimension is 32, and the edge embedding feature is 3, comprising coefficients projected onto the locally constructed coordinate frame as shown in Eq. (7). We use 4 equi-graph layers throughout the tests. The LRFT module consists of 3 parameters: input dimension N , internal rank r , and N' . Our model adopts a configuration of $1024/(32 + 3)/128$, where rank r is the sum of graph node feature dimension (32) and coordinate embedding dimension 3. The submatrix determinant check for similarity score matrix $\hat{\mathbf{S}}$ is 5×5 along the borders and 7×7 within the matrix. A performance analysis comparing different parameter configurations is presented in the subsequent ablation section. All training and inference tasks are conducted on a single RTX 3090 GPU.

Evaluation Metrics. We employ the average Relative Error (RE) and Translation Error (TE) metrics from PointDSC [2] to assess the accuracy of predicted pose errors in successful registration. Additionally, we incorporate Registration Recall (RR) and *F1 score* as performance evaluation measures. To evaluate these metrics, we establish potential corresponding point pairs $(\mathbf{x}_i, \mathbf{y}_j) \in \Omega$ using input points from two frames, following the correspondence establishment approach outlined in PointDSC [2]. We apply the predicted transformation to the source frame point \mathbf{x}_i , recording a pairwise registration success only when the average Root Mean Square Error (RMSE) falls below a predefined threshold τ . The registration recall value δ is calculated as:

$$\delta = \sqrt{\frac{1}{\mathcal{N}(\Omega)} \sum_{(\mathbf{x}_i, \mathbf{y}_j) \in \Omega} \mathbb{1}[\|\hat{\mathbf{R}}\mathbf{x}_i + \hat{\mathbf{t}} - \mathbf{y}_j\|^2 < \tau]}, \quad (14)$$

where $\mathcal{N}(\Omega)$ denotes the total number of ground truth correspondences in set Ω . The symbol $\mathbb{1}$ functions as an indicator for condition satisfaction. Removing the conditional check within the indicator brackets on the equation’s right side transforms it into a standard Root Mean Square Error (RMSE),

$\sqrt{\frac{1}{\mathcal{N}(\Omega)} \sum_{(\mathbf{x}i, \mathbf{y}j) \in \Omega} |\hat{\mathbf{R}}\mathbf{x}i + \hat{\mathbf{t}} - \mathbf{y}j|^2}$. This RMSE metric is utilized in ablation experiments for parameter analysis. The *F1 score* is defined as $2 \cdot \frac{Precision \times Recall}{Precision + Recall}$.

Baseline Methods 1) For the 3DMatch [52] benchmark, we compare our model with vanilla RANSAC implementations using various iterations and optimization refinements. We also include Go-ICP [51] and Super4PCS [31], which operate on raw points. Among learning-based methods, we compare with DGR [9] and PointDSC [2] combined with FCGF descriptors [10]. Additionally, we select D3Feat [3], SpinNet [1], and RoReg [44], which incorporate rotation invariance or equivariance. These learning methods do not support descriptor replacement, denoted by *. 2) For KITTI sequences [20], we implement the hand-crafted feature descriptor FPHF [37] due to performance saturation issues with FCGF [10] descriptors, as noted in PointDSC [2]. RoReg [44] is replaced with the registration model from the FCGF paper [10] (denoted as FCGF-Reg) due to public code limitations for KITTI dataset.

4.1 Indoor Fragments/Scans Registration

Point clouds are initially downsampled using a 5cm voxel size to generate 1024 sampled points. Registration success is evaluated using thresholds of 30cm for translational error (TE) and 15° for rotational error (RE). The correspondence distance threshold τ in Eq. (14) is set at 10cm. Comparative results between our proposed model and baseline approaches are presented in Tab. 1. Our model outperforms all comparison methods, despite slightly slower latency compared to RANSAC with 1k iterations. RoReg and SpinNet, ranking second and third, demonstrate minimal registration errors and maximal registration scores, highlighting the advantages of incorporating rotation features. While our model can integrate the FCGF descriptor, we present evaluation results using the PointNet++ learning descriptor for end-to-end training.

4.2 Outdoor Scenes Registration

The input point cloud from the KITTI sequences [20] is downsampled using a voxel size of 30 cm to generate 1024 sparse points for the experiments. We set the registration thresholds at 60cm for Translation Error and 5° for Rotation Error. To measure Registration Recall (RR), we establish a threshold τ of 60cm. Tab. 2 presents the quantitative results for comparison. Our proposed model demonstrates plausible performance compared to other methods, exhibiting minimal rotation and translation errors, and achieving the highest registration recall rate of 94.60% when compared to RoReg, the second-best model. However, RoReg has a remarkable weakness in its real-time performance, registering in over 30

Table 1: Evaluation results of registration methods on 3DMatch show non-learning-based approaches at the top and deep-learning registration models below. The learning-based feature descriptor FCGF [10] is tested with various learning baseline approaches. The symbol * indicates the original model implementation on 3D Match due to the lack of support for this feature descriptor replacement.

	RE($^{\circ}$) \downarrow	TE(cm) \downarrow	RR(%) \uparrow	F1(%) \uparrow	Time(s) \downarrow
RANSAC-1k [17]	3.16	9.67	86.57	76.62	0.08
RANSAC-10k	2.69	8.25	90.70	80.76	0.58
RANSAC-100k	2.49	7.54	91.50	81.43	5.50
RANSAC-100k + refine	2.17	6.76	92.30	81.43	5.51
Go-ICP [51]	5.38	14.70	22.95	20.08	771.0
Super4PCS [31]	5.25	14.10	21.6	19.86	4.55
DGR [9] \uparrow	2.40	7.48	91.30	89.76	1.36
D3Feat* [3]	2.57	8.16	89.79	87.40	0.14
SpinNet* [1]	1.93	6.24	93.74	92.07	2.84
PointDSC [2]	2.06	6.55	93.28	89.35	0.09
RoReg* [44] \downarrow	1.84	6.28	93.70	91.60	2226
Ours	1.67	5.68	94.60	94.35	0.12

Table 2: Registration methods for evaluation on the KITTI dataset [20] involve testing the hand-crafted FPHF descriptor [37] in conjunction with different learning strategies. The symbol * indicates the lack of support for replacing the feature descriptor, as per the original implementations on KITTI.

	RE($^{\circ}$) \downarrow	TE(cm) \downarrow	RR(%) \uparrow	F1(%) \uparrow	Time(s) \downarrow
RANSAC-1k [17]	2.51	38.23	11.89	14.13	0.20
RANSAC-10k	1.90	37.17	48.65	42.35	1.23
RANSAC-100k	1.32	25.88	74.37	73.13	13.7
RANSAC-100k + refine	1.28	18.42	77.20	74.07	15.65
Go-ICP [51]	5.62	42.15	9.63	12.93	802
Super4PCS [31]	4.83	32.27	21.04	23.72	6.29
FCGF-Reg* [10] \downarrow	1.95	18.51	70.86	68.90	0.09
DGR [9]	1.45	14.6	76.62	73.84	0.86
D3Feat* [3]	2.07	18.92	70.06	65.31	0.23
SpinNet* [1]	1.08	10.75	82.83	80.91	3.46
PointDSC* [2]	1.63	12.31	74.41	70.08	0.31
Ours	0.92	8.74	83.83	85.09	0.14

minutes. Additionally, we visually display the registration sample outcomes of our proposed model on 3DMatch and KITTI below. For a more comprehensive visual comparison to baselines, please refer to the supplementary section.

To verify the proposed model performance under the different numbers of sampled input points, we also implement the sparse tests in Tab. 3, by comparing with FCGF registration, D3Feat with or without PointDSC combination, and SpinNet. Our model has a consistent performance on 3DMatch over the com-

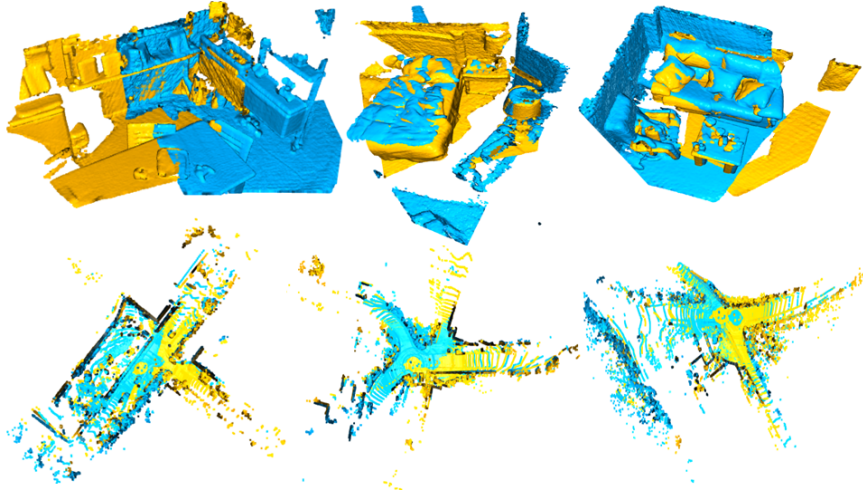


Fig. 4: The visual registration results of the proposed model on 3DMatch [52] and KITTI [20] are illustrated in the registration samples. Points from the target frame are represented in blue, whereas points converted from the source frame to the target frame by the predicted transform are visualized in yellow.

parison methods. In addition, using more points can boost the proposed model registration performance with a marginal gain. Considering a good trade-off between accuracy and computational efficiency, 1024 is chosen as the input point number in our final implemented model, because there is only a small performance gap within 1% compared to the 2048 and 4096 number of points.

Table 3: RR results on 3DMatch with a different number of sampled points.

#Sampled Points	4096	2048	1024	512	256	Average
FCGF-Reg [17]	91.7	90.3	89.5	85.7	80.5	87.5
D3Feat [3]	91.9	90.4	89.8	86.0	82.5	88.1
D3Feat [3]+PointDSC [2]	92.1	92.5	90.8	87.4	83.6	89.3
SpinNet [1]	93.8	93.6	93.7	89.5	85.7	91.3
Ours	95.3	94.8	94.6	91.3	88.5	92.9

4.3 Ablation Study

Firstly, we present a table (Tab. 4) displaying different configurations and combinations of module blocks in the proposed model. Table analysis reveals significant accuracy enhancements with our model’s learning-based descriptor over pre-trained FCGF and FPHF descriptors, as shown in rows 1 and 2, across multiple metrics. The descriptor learning layers (row 3) and equivariant graph CNN layers (row 4) are key to performance improvements. Replacing the equivariant with standard graph CNN layers (row 5) impairs rotation convergence, while omitting the LRFT module (row 6) marginally reduces performance. Ball query graph initialization (rows 8) outperforms KNN search (row 7) in efficiency and

Table 4: An ablation study was conducted on our model design, using 3DMatch [52] as the test dataset. The study involved comparing various descriptor combinations, exploring different combinations of layers for equi-feature learning, analyzing different methods of graph construction, and examining the impact of regularizers.

		RE($^{\circ}$) \downarrow	TE(cm) \downarrow	RR(%) \uparrow	F1(%) \uparrow	Time(s) \downarrow
1.	FCGF Descriptor + Ours	1.62	6.24	93.87	94.28	0.15
2.	FPHF Descriptor + Ours	1.83	6.49	83.62	73.06	0.12
3.	w/o Feature Descriptor Layers	10.26	9.02	61.39	60.04	0.08
4.	w/o Equi-graph Layers	9.64	8.37	62.45	60.03	0.06
5.	Replacing by normal GCNN	8.32	5.94	68.52	67.54	0.10
6.	w/o LRFT layers	2.76	6.47	83.09	81.78	0.09
7.	KNN Graph Construction	1.72	5.31	92.37	93.74	0.16
8.	Ball Query Graph Construction	1.67	5.68	94.60	94.35	0.12
9.	w/o Rank Regularizer	6.41	7.92	76.45	78.09	0.10
10.	w/o Sub-matrix Rank Verification	2.58	6.93	87.76	88.36	0.07
11.	Descriptor Layers + GCNN	8.32	5.94	68.52	67.54	0.10
12.	w/o Descriptor Layers + Equi-GCNN	10.26	9.02	61.39	60.04	0.08
13.	SpinNet + Equi-GCNN	2.93	5.97	82.16	83.74	3.62
14.	Ours	1.67	5.68	94.60	94.35	0.12

real-time outcomes on 3DMatch. Additionally, integrating rank regularizers (row 9) and sub-matrix rank verification (row 10) enhances model performance. Although equivariant layers introduce a slight computational delay (around 60ms), the overall performance gain is significant. Lastly, experiments of row 12-13 show that directly applying E-GCNN layers on the input or replacing the descriptor layers with equivariant structures such as SpinNet does not yield performance on par with our proposed method. Additionally, replacing E-GCNN layers with normal GCNN (row 11) results in significant performance degradation. These findings highlight the tailored effectiveness of our equi-approach.

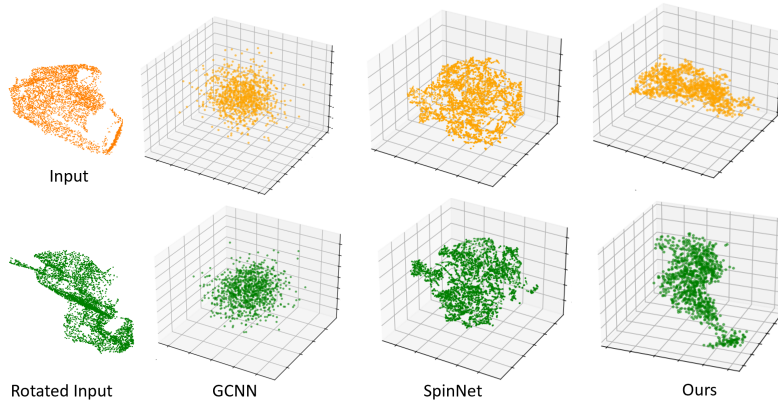


Fig. 5: The t-SNE comparisons of equi-features outputs.

For a more intuitive understanding of our model capability of learning equi-features, we provide t-SNE plots below for different encoder network outputs,

showing that our Equi-Graph CNN produces features that are equivariant to rotated input. In comparison, features extracted by the conventional Graph CNN (GCNN) and SpinNet ($\text{SO}(2)$ equivariant) as baselines do not exhibit rotation equivariance. To further elucidate the visual relationship between the rank of the feature similarity score matrix $\hat{\mathbf{S}}$ and point correspondences with or without equivariant features, please refer to the supplementary section.

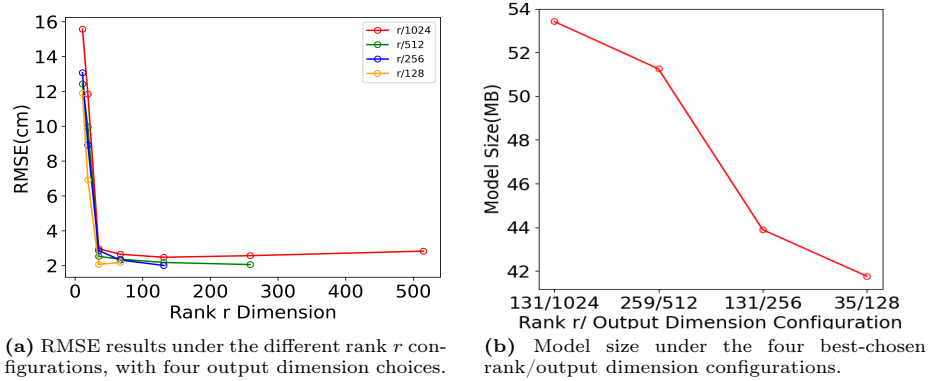


Fig. 6: The left is the RMSE results plot of four output feature sizes of the LRFT module under the various rank dimensions, and the model size of the best r /output dimension configuration of each curve in the left plot is presented at right.

Moreover, we provide the parameter configuration comparison of LRFT layers as well. It can be observed that varying the LRFT output feature number shows that increasing the middle-rank dimension enhances accuracy, yet beyond 200 ranks, error climbs a bit as depicted by the red curve. The relationship among the input feature number N , rank r , and the output number N' is defined as $r \ll N' < N$, indicating that N' should be at least twice the size of r , as evidenced by the termination position of curves in Fig. 6. Additionally, the optimal model size with the lowest RMSE is achieved with the 35/128 (r/N') LRFT configuration, corresponding to the yellow curve on the left.

5 Conclusion

We introduce an end-to-end model that leverages pre-trained feature descriptors or learns directly from raw scan points across two frames, incorporating equivariance embedding through graph layers, Low-Rank Feature Transformation and similarity score computation. Validation in both indoor and outdoor datasets confirms the superior performance of our proposed model. Ablation studies further substantiate the model design. Notably, the model’s latency demonstrates its potential applicability in visual odometry. Future work could explore generalizing this framework to be input-order permutation invariant through graph attention layers or pooling, potentially integrating additional sensor modalities to address dynamic challenges.

Acknowledgements

This research work was jointly supported by the Young Scientists Fund of the National Natural Science Foundation of China (42301520), the Research Grants Council of Hong Kong (25206524), the Platform Project of UAV Research Centre (P0049516), the Seed Project of Smart Cities Research (P0051028), the Research Cultivation Program of ShenZhen University (2023JCT003), and the Pearl River Talent (2021JC02G046). Furthermore, we appreciate the University of Melbourne to provide the Graduate Research Scholarship for Xueyang to pursue his joint PhD program.

References

1. 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. pp. 11753–11762 (2021)
2. Bai, X., Luo, Z., Zhou, L., Chen, H., Li, L., Hu, Z., Fu, H., Tai, C.L.: 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 (2021)
3. 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. pp. 6359–6367 (2020)
4. Bogatskiy, A., Anderson, B., Offermann, J., Roussi, M., Miller, D., Kondor, R.: Lorentz group equivariant neural network for particle physics. In: International Conference on Machine Learning. pp. 992–1002. PMLR (2020)
5. Brandstetter, J., Hesselink, R., van der Pol, E., Bekkers, E.J., Welling, M.: Geometric and physical quantities improve e (3) equivariant message passing, 2021. URL <https://arxiv.org/abs/2110.02905>
6. Chatzipantazis, E., Pertigkiozoglou, S., Dobriban, E., Daniilidis, K.: Se(3)-equivariant attention networks for shape reconstruction in function space. arXiv preprint arXiv:2204.02394 (2022)
7. Chen, H., Liu, S., Chen, W., Li, H., Hill, R.: Equivariant point network for 3d point cloud analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14514–14523 (2021)
8. Cheng, Y., Huang, Z., Quan, S., Cao, X., Zhang, S., Yang, J.: Sampling locally, hypothesis globally: accurate 3d point cloud registration with a ransac variant. *Visual Intelligence* **1**(1), 20 (2023)
9. Choy, C., Dong, W., Koltun, V.: Deep global registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2514–2523 (2020)
10. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8958–8966 (2019)
11. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: International conference on machine learning. pp. 2990–2999. PMLR (2016)
12. Cohen, T.S., Geiger, M., Köhler, J., Welling, M.: Spherical cnns. arXiv preprint arXiv:1801.10130 (2018)

13. Deng, C., Litany, O., Duan, Y., Poulenard, A., Tagliasacchi, A., Guibas, L.J.: Vector neurons: A general framework for so (3)-equivariant networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12200–12209 (2021)
14. Du, W., Zhang, H., Du, Y., Meng, Q., Chen, W., Zheng, N., Shao, B., Liu, T.Y.: Se(3) equivariant graph neural networks with complete local frames. In: International Conference on Machine Learning. pp. 5583–5608. PMLR (2022)
15. El Banani, M., Gao, L., Johnson, J.: Unsupervised point cloud registration via differentiable rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7129–7139 (2021)
16. Finzi, M., Stanton, S., Izmailov, P., Wilson, A.G.: Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data (2020)
17. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* **24**(6), 381–395 (1981)
18. Fuchs, F., Worrall, D., Fischer, V., Welling, M.: Se(3)-transformers: 3d rotation equivariant attention networks. *Advances in neural information processing systems* **33**, 1970–1981 (2020)
19. Fuchs, F.B., Wagstaff, E., Dauparas, J., Posner, I.: Iterative se (3)-transformers. In: Geometric Science of Information: 5th International Conference, GSI 2021, Paris, France, July 21–23, 2021, Proceedings 5. pp. 585–595. Springer (2021)
20. 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. pp. 3354–3361. IEEE (2012)
21. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: International conference on machine learning. pp. 1263–1272. PMLR (2017)
22. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Message passing neural networks. *Machine learning meets quantum physics* pp. 199–214 (2020)
23. Gojcic, Z., Zhou, C., Wegner, J.D., Guibas, L.J., Birdal, T.: Learning multiview 3d point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1759–1769 (2020)
24. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021)
25. 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 (CVPR). pp. 4267–4276 (June 2021)
26. Hutchinson, M., Lan, C.L., Zaidi, S., Dupont, E., Teh, Y.W., Kim, H.: Lietransformer: Equivariant self-attention for lie groups (2020)
27. Jenner, E., Weiler, M.: Steerable partial differential operators for equivariant neural networks. In: International Conference on Learning Representations (2022), <https://openreview.net/forum?id=N9W24a4zU>
28. Keriven, N., Peyré, G.: Universal invariant and equivariant graph neural networks. *Advances in Neural Information Processing Systems* **32** (2019)
29. Köhler, J., Klein, L., Noé, F.: Equivariant flows: exact likelihood generative learning for symmetric densities. In: International conference on machine learning. pp. 5361–5370. PMLR (2020)

30. Lin, C.E., Song, J., Zhang, R., Zhu, M., Ghaffari, M.: Se (3)-equivariant point cloud-based place recognition. In: Conference on Robot Learning. pp. 1520–1530. PMLR (2023)
31. Mellado, N., Aiger, D., Mitra, N.J.: Super 4pcs fast global pointcloud registration via smart indexing. In: Computer graphics forum. vol. 33, pp. 205–215. Wiley Online Library (2014)
32. Pais, G.D., Ramalingam, S., Govindu, V.M., Nascimento, J.C., Chellappa, R., Miraldo, P.: 3dregnet: A deep neural network for 3d point registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7193–7203 (2020)
33. Park, J., Zhou, Q.Y., Koltun, V.: Colored point cloud registration revisited. In: Proceedings of the IEEE international conference on computer vision. pp. 143–152 (2017)
34. Park, S.Y., Subbarao, M.: An accurate and fast point-to-plane registration technique. Pattern Recognition Letters **24**(16), 2967–2976 (2003)
35. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
36. 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. pp. 11143–11152 (2022)
37. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: 2009 IEEE international conference on robotics and automation. pp. 3212–3217. IEEE (2009)
38. Satorras, V.G., Hoogeboom, E., Welling, M.: E(n) equivariant graph neural networks (2021)
39. Schütt, K.T., Sauceda, H.E., Kindermans, P.J., Tkatchenko, A., Müller, K.R.: Schnet—a deep learning architecture for molecules and materials. The Journal of Chemical Physics **148**(24) (2018)
40. Shi, W., Rajkumar, R.: Point-gnn: Graph neural network for 3d object detection in a point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1711–1719 (2020)
41. Sosnovik, I., Szmaja, M., Smeulders, A.: Scale-tooltool steerable networks. arXiv preprint arXiv:1910.11093 (2019)
42. Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., Riley, P.: Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. arXiv preprint arXiv:1802.08219 (2018)
43. 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. pp. 1630–1641 (2022)
44. 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 Transactions on Pattern Analysis and Machine Intelligence (2023)
45. Wang, H., Wang, C., Chen, C.L., Xie, L.: F-loam: Fast lidar odometry and mapping. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4390–4396. IEEE (2021)
46. Wang, Y., Solomon, J.M.: Deep closest point: Learning representations for point cloud registration. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3523–3532 (2019)

47. Weiler, M., Cesa, G.: General E(2)-Equivariant Steerable CNNs. In: Conference on Neural Information Processing Systems (NeurIPS) (2019)
48. Weiler, M., Geiger, M., Welling, M., Boomsma, W., Cohen, T.S.: 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems* **31** (2018)
49. Weiler, M., Hamprecht, F.A., Storath, M.: Learning steerable filters for rotation equivariant cnns. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 849–858 (2018)
50. Xinyi, Z., Chen, L.: Capsule graph neural network. In: International conference on learning representations (2018)
51. Yang, J., Li, H., Campbell, D., Jia, Y.: Go-icp: A globally optimal solution to 3d icp point-set registration. *IEEE transactions on pattern analysis and machine intelligence* **38**(11), 2241–2254 (2015)
52. 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. pp. 1802–1811 (2017)
53. Zhang, Y., Rabbat, M.: A graph-cnn for 3d point cloud classification. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6279–6283. IEEE (2018)
54. Zhou, Q.Y., Park, J., Koltun, V.: Fast global registration. In: Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II 14. pp. 766–782. Springer (2016)
55. Zhu, M., Ghaffari, M., Clark, W.A., Peng, H.: E2pn: Efficient se (3)-equivariant point network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1223–1232 (2023)
56. Zhu, M., Ghaffari, M., Peng, H.: Correspondence-free point cloud registration with so (3)-equivariant implicit shape representations. In: Conference on Robot Learning. pp. 1412–1422. PMLR (2022)