Correspondence-Free Point Cloud Registration with SO(3)-Equivariant Implicit Shape Representations

Anonymous Author(s)
Affiliation
Address
email

Abstract: This paper proposes a correspondence-free method for point cloud rotational registration. We learn an embedding for each point cloud in a feature space that preserves the SO(3)-equivariance property, enabled by recent developments in equivariant neural networks. The proposed shape registration method achieves three major advantages through combining equivariant feature learning with implicit shape models. First, the necessity of data association is removed because of the permutation-invariant property in network architectures similar to PointNet. Second, the registration in feature space can be solved in closed-form using Horn's method due to the SO(3)-equivariance property. Third, the registration is robust to noise in the point cloud because of implicit shape learning. The experimental results show superior performance compared with existing correspondence-free deep registration methods.

Keywords: Point cloud registration, implicit shape model, equivariant neural network, representation learning

1 Introduction

Point clouds are a major form of 3D information representation in perception with numerous applications today, including intelligent robots and self-driving cars. The registration of two different point clouds capturing the same underlying object or scene refers to estimating the relative transformation that aligns them. Point cloud registration enables information aggregation across several observations, which is essential for many tasks such as mapping, localization, shape reconstruction, and object tracking. Given perfect point-to-point pairings between the point-cloud-pair, the relative transformation can be solved in closed-form using Horn's method [1].

However, such a pairing is not available or corrupted in real-world problems, posing a significant challenge for the point cloud registration task, namely data association. In the deep learning era, the strong feature learning capacity of neural networks enables better description and discrimination of points, improving their matching. Overall, these methods can be categorized as correspondence-based registration.

An alternative approach, called correspondence-free registration, is to completely circumvent the matching of points by learning a global representation for the entire point cloud and estimating the transformation by aligning their representations instead of the original points. Similar to correspondence-based registration, correspondence-free registration can also be with or without the involvement of deep learning. The main obstacle in this type of approach is that the mapping between the input Euclidean space and the output feature space realized by neural networks is nonlinear and obscure. Thus it is common to rely on iterative optimization with local linearization like Gauss-Newton [2]. The performance of these approaches deteriorates when the initial pose difference gets larger.

In this paper, we innovatively employ equivariant neural networks and implicit shape learning into correspondence-free registration by constructing a feature space with two major advantages. First, thanks to the equivariance property, the feature space preserves the same rotation operation as the Euclidean input space, allowing us to solve the feature-space registration in closed form using Horn's

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

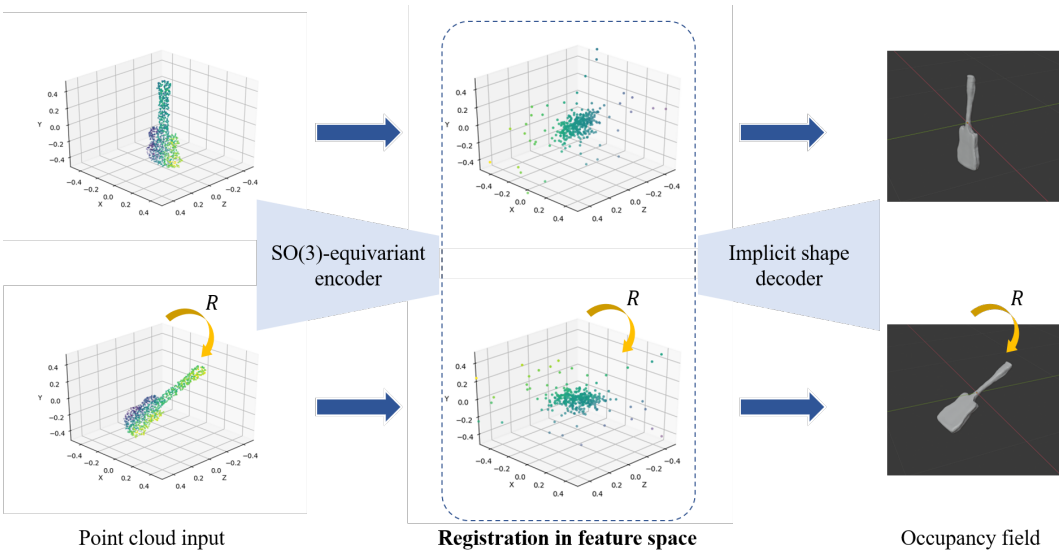

Figure 1: Overview of the correspondence-free rotational registration network.

method. As a result, the accuracy is independent of the initial pose difference. Second, the implicit shape learning encourages the network to learn features describing the underlying geometry rather than specific points sampled on the geometry, encouraging features to be robust to noise. In this work, we focus on the rotational registration task.

In particular, this work has the following contributions.

1. We propose a correspondence-free point cloud registration method via $\mathrm{SO}(3)$-equivariance representation learning and closed-form pose estimation.
2. The proposed method achieves accurate registration results independent of initial pose differences with robustness against noisy points.
3. Open source software will be available (after the final decision).

2 Preliminaries on Equivariance and Vector Neurons

Symmetry (or equivariance) in a neural network is an important ingredient for efficient learning and generalization. It also allows us to build a clear connection between the changes in the input space and the output space. For example, convolutional neural networks are translation-equivariant. It makes a network generalize to image shifting and enables huge parameter saving. However, other symmetries are less explored. For example, most point cloud processing networks (e.g., PointNet [3], DGCNN [4]) are not equivariant to rotations.

We leverage the recently proposed Vector Neuron layers [5] to enable rotation-equivariant feature learning for point clouds. For the sake of completeness, we briefly introduce the preliminary on equivariance and the design of layers.

A function $f : X \to X$ is equivariant to a set of transformations G, if for any $g \in G$, f and g commutes, i.e., $g \cdot f(x) = f(g \cdot x), \forall x \in X$. For example, applying a translation on a 2D image and then going through a convolution layer is identical to processing the original image with a convolution layer and then shifting the output feature map. Therefore convolution layers are translation-equivariant.

Vector Neurons extend the equvariance property to $\mathrm{SO}(3)$ rotations for point cloud neural networks. The key idea is to augment the scalar feature in each feature dimension to a vector in $\mathbb{R}^3$ and redesign the linear, nonlinear, pooling, and normalization layers accordingly. In a regular network, the feature matrix with feature dimension C corresponding to a set of N points is $V \in \mathbb{R}^{N \times C}$. In Vector Neuron networks, it is $V \in \mathbb{R}^{N \times C \times 3}$. The mapping between layers can be written as $f : \mathbb{R}^{N \times C_l \times 3} \to$

$\mathbb{R}^{N \times C_{l+1} \times 3}$, where l is the layer index. Following this design, the representation of $\mathrm{SO}(3)$ rotations in feature space is straight-forward: $g(R) \cdot V := VR$, where $g(R)$ denotes the rotation operation in the feature space, parameterized by the 3-by-3 rotation matrix $R \in \mathrm{SO}(3)$. In the following, we ignore the first dimension N of V for simplicity.

The linear layer for Vector Neurons is similar to a traditional MLP: $f_{\text{lin}}(V) = WV$, where $W \in \mathbb{R}^{C_{l+1} \times C_l}$. It is easy to see that such a mapping is $\mathrm{SO}(3)$-equivariant:

$$g(R) \cdot f_{\text{lin}}(x) = WVR = f_{\text{lin}}(g(R) \cdot V) \tag{1}$$

Designing nonlinearities for Vector Neuron networks is less trivial. A naive ReLu on all elements in the feature matrix will break the equivariance. Instead, a *vectorized* version of ReLu is designed as the truncation of vectors along a hyperplane. Specifically, given an input feature matrix V, predict a canonical direction $k = UV \in \mathbb{R}^{1 \times 3}$ where $U \in \mathbb{R}^{1 \times C}$. Then the ReLU is defined as:

$$v' = \begin{cases} v, & \text{if } \langle v, k \rangle \geq 0 \\ v - \langle v, \frac{k}{\|k\|} \rangle \frac{k}{\|k\|}, & \text{otherwise} \end{cases} \tag{2}$$

for each row $v \in \mathbb{R}^{1 \times 3}$ of V. This design preserves $\mathrm{SO}(3)$ equivariance because k and V will rotate together, and $\langle VR, kR \rangle = \langle V, k \rangle$, meaning that rotations do not change the activation.

For the design of pooling layers and normalization layers of Vector Neurons, please refer to the original paper for details [5].

3 Methodology

The overall idea of correspondence-free registration is to transform the alignment of points in input space to the alignment of global features where point-wise matching is not needed. In this work, we employ a $\mathrm{SO}(3)$-equivariant neural network to construct a feature space preserving the same rotation representations as to the input space, thus simplifying solving the rotation for feature alignment. Furthermore, deep implicit representation learning improves the robustness of the features against noise and allows registration between two different scans of the same geometry.

3.1 $\mathrm{SO}(3)$-equivariant point cloud global feature extractor

We choose PointNet as the feature extractor backbone. The permutation invariance property implies that two point clouds correspond to the same feature embeddings if they only differ in the permutation of points. We replace the layers in PointNet with the corresponding Vector Neuron version, similar to the work of Deng et al. [5]. Denote the feature extraction network as f, a point cloud $P \in \mathbb{R}^{N \times 3}$, and its rotated and permuted copy $P' = MPR$, where $M \in \mathbb{R}^{N \times N}$ is a permutation matrix and $R \in \mathrm{SO}(3)$ is a 3-by-3 rotation matrix. Then we have

$$Q' = f(P') = f(MPR) = f(MP)R = f(P)R = QR \tag{3}$$

where $Q := f(P) \in \mathbb{R}^{C \times 3}$, guaranteed by the $\mathrm{SO}(3)$-equivariance and permutation-invariance properties, which is essential for feature-space registration.

3.2 Deep implicit representation learning

Being able to register two rotated copies of the same point cloud is not enough for practical use. In real application, the measurements have noise, and the individual points captured by different scans generally do not correspond to the same physical points. To our knowledge, we are among the first to address the need for registration across noisy and different scans of point clouds in deep correspondence-free registration methods.

Our solution is to repurpose the global feature (for registration ultimately) for implicit shape representation. Following Occupancy Network, we build an encoder-decoder network. The encoder is the aforementioned $\mathrm{SO}(3)$-equivariant feature extractor, and the decoder takes the feature and a queried position as input, predicting the occupancy value of that position. In this way, the encoder is encouraged to predict identical features if the input point clouds correspond to the same geometry even if they scan different points. Gaussian noise is also added to the input during training to improve the robustness. In order to output an identical implicit function field, we adopted the same invariant layer designed by Deng et al. [5] to convert the $\mathrm{SO}(3)$-equivariant features to be $\mathrm{SO}(3)$-invariant before feeding them into the decoder.

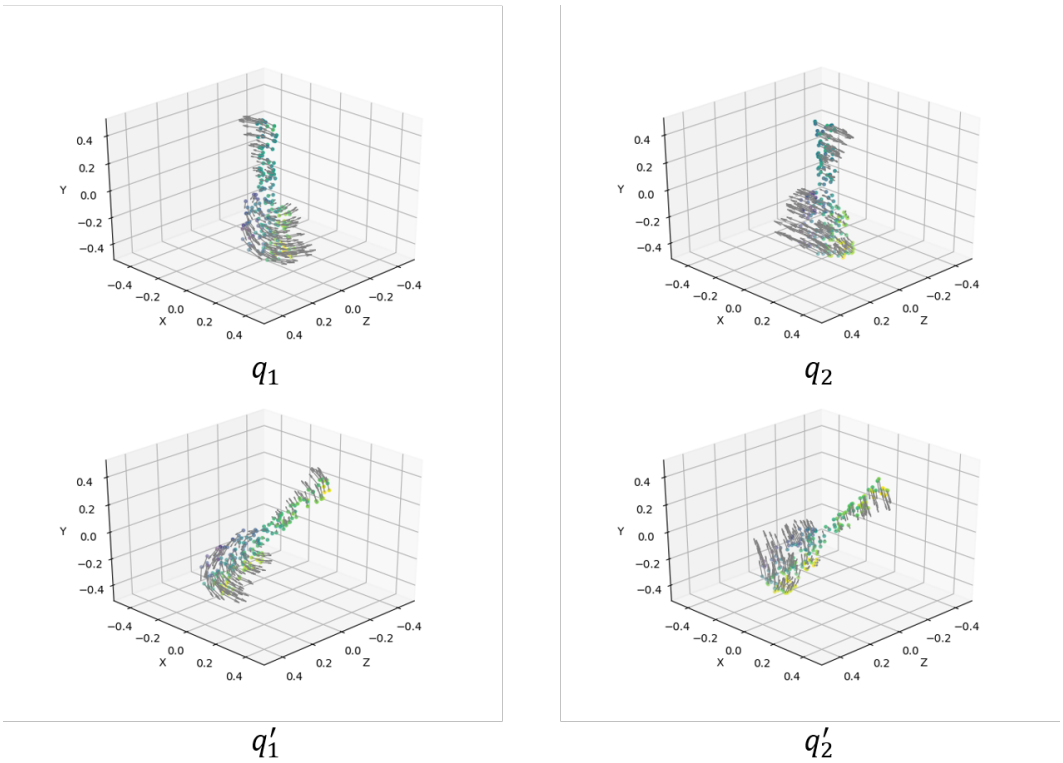

Figure 2: Visualization of two feature vectors as vector fields applied on the original Euclidean space. q_1 and q_2 are two rows of Q, and $q_1^{'}$ and $q_2^{'}$ are two rows of Q'. The generated vector fields are identical up to an rotation.

3.3 Point cloud registration through feature alignment

With the above network design, we can further relax the connection between the two point clouds P' and P in (3). They do not have to be permutations of each other, as long as they are sampled on the same geometry. However, the equation only holds approximately in this case since the implicit shape model is not trained perfectly.

One can see that (3) is exactly in the form of orthogonal Procrustes problem. The learned features Q and Q' can be regarded as two *pseudo* point clouds and each row of them are *automatically matched*. This problem has a standard closed-form solution. First calculate the cross covariance matrix $H = Q^{\mathsf{T}}Q'$. Then conduct Singular Value Decomposition (SVD) on H: $H = USV^{\mathsf{T}}$. The optimal rotation matrix $\hat{R}$ is given as:

$$\hat{R} = V\Lambda U^{\mathsf{T}} \tag{4}$$

where $\Lambda = \text{diag}(1, 1, \det(VU^{\mathsf{T}})) \in \mathbb{R}^{3\times 3}$.

Not only can each row $q \in \mathbb{R}^3$ of the feature matrix Q be regarded as a point in Euclidean space, q can also be interpreted as an element in $so(3)$, and visualized as a vector field applied on the Euclidean space where the input point clouds live. Each feature generates such a vector field. Because of the rotation-equivariance property, the vector field generated from the feature of the rotated point cloud is identical to the rotated vector field generated from the feature of the original point cloud. See Fig. 2 for an example.

4 Experiments

Consistent with existing literature, we conduct experiments on the ModelNet40 dataset [6]. It is composed of 12311 CAD models from 40 categories of objects. Following the official split, 9843 models are used for training, and 2468 models are used for testing. The models are preprocessed

Table 1: Rotational registration error given different random initial rotations. Tested using ModelNet40 official test set. The *max initial rotation angle* refers to the maximum rotation angle allowed in the generation of initial random rotations during testing. The rotation axis is fully randomized. The last column (180 degrees) corresponds to unconstrained initial rotations. All values are in degrees.

Max initial rotation angle	0	30	60	90	120	150	180
Methods	Rotation error after registration						
PCR-Net	7.077	9.498	27.379	68.223	109.608	129.289	133.486
FMR	0.000	0.469	5.292	21.947	42.256	66.385	79.432
Ours	0.019	0.018	0.019	0.020	0.021	0.018	0.021

Table 2: Rotational registration error of noisy point clouds.

Max initial rotation angle	0	30	60	90	120	150	180
Methods	Rotation error after registration						
PCR-Net	7.212	9.295	27.397	69.956	108.425	125.547	132.396
FMR	1.564	2.012	5.784	19.950	45.728	67.653	82.048
Ours	2.949	3.019	3.172	2.912	3.628	3.198	4.351

using the tool provided by Stutz and Geiger [7] to obtain watertight and simplified models that are centered and scaled to a unit cube.

During training, 1000 points are randomly sampled on each mesh model as the input point cloud. The occupancy value is queried at 2048 points in the unit cube to compose the loss function. The sampling strategy of the querying points is consistent with the Occupancy Network [8]. The dimension of the learned global feature is set as $C = 342$, corresponding to $342 \times 3 = 1026$ dimensions for a traditional feature vector, close to the dimension chosen by previous literature including PointNetLK [9], PCRNet [10], and Feature-Metric Registration (FMR) [11]. The network is trained on a single Tesla V100 GPU with batch size 24.

4.1 Registration with rotated copies of point clouds

We first tested the registration of two points clouds that are only different by rotation and permutation. Both point clouds have 1000 points sampled from the mesh model. The initial rotations are generated randomly using axis-angle representation to specify the maximum magnitude of the initial rotations. We compare with two representative correspondence-free deep registration methods as baselines: PCR-Net [10] and FMR [11]. The experiments on the baseline methods are based on their official open-source implementation and pretrained weights. The results are shown in Table 1. We can see that both PCR-Net and FMR are sensitive to the initial rotation angle and only work properly when the initial rotation is small because they rely on iterative refinements. In comparison, the registration accuracy of our method is independent of the initial rotation angle, and it provides almost perfect rotational registration.

4.2 Registration with rotated copies of point clouds corrupted with Gaussian noise

In practice, the pair of point clouds for registration will not be identical to each other. Therefore we test the case where the source and target point clouds are both corrupted by noise. We add a Gaussian noise with $\sigma = 0.01$ to all points before putting them through the networks. The rest of the setup is the same as in Sec. 4.1. As shown in Table 2, PCR-Net and FMR show a similar trend of increasing error when the initial angle gets larger. Our method, though showing a slightly larger registration error, still outperforms the baseline in most columns. Most importantly, the error remains consistent across different initialization. The result shows that our method not only works in an ideal situation where the point clouds are noise-free but also delivers accurate rotation estimation under noise corruption.

Table 3: Rotational registration error of point clouds with different densities.

Max initial rotation angle	0	30	60	90	120	150	180
Methods	Rotation error after registration						
PCR-Net	7.365	9.702	27.044	65.953	109.220	126.430	130.369
FMR	1.951	2.712	6.913	23.748	45.141	66.972	81.558
Ours	17.701	21.605	19.739	17.392	22.190	20.143	24.240

4.3 Registration with point clouds with different densities

To further test the robustness of the proposed method against variations in the point clouds, we tested the registration performance when given two point clouds of different densities. Here we sample 1000 and 500 points for the pair of point clouds, respectively. The registration accuracy is shown in Table 3. The trend of the three methods is similar to what is shown in Sec. 4.1 and Sec. 4.2. We do observe an increase of error in our method in this experiment. A possible reason is that at the beginning of our PointNet encoder is an edge-convolution layer to initialize the feature dimensions of each point, which is dependent on the neighborhood. Change of density may change the feature at the first layer, affecting the feature space registration. This issue is to be analyzed in more detail in future work. However, our result still outperforms the baselines when the initial rotation angle is larger than 90 degrees.

Overall, our proposed orientation registration method can estimate consistent results independent of initial rotations. It outperforms the baselines even under the presence of noise, showing the practicality of our proposed method. Under density variance, the accuracy of our method deteriorates, showing an improvement direction of our method. Nevertheless, it still outperforms the baseline methods under large initial rotations.

5 Related Work and Discussion

In this work, we connect ideas in several different fields, including sensor registration, implicit shape learning, and equivariant neural networks. It may benefit the understanding of our method provided a broader context of each of the fields. Here we first review point cloud registration methods with and without data association with a focus on deep learning-based approaches. Then we review literature about equivariant neural networks and deep implicit models briefly.

5.1 Correspondence-based point cloud registration

A major challenge here is to recover the corresponding pairs of points from a pair of point clouds. ICP simply matches the closest points together, solves the transformation, and iteratively rematches the closest points after aligning the two point clouds using the estimated transformation [12]. Point-to-line [13], point-to-plane [14], plane-to-plane [15], and Generalized-ICP [16] build local geometric structures to the loss formulation.

A challenge in the matching is to distinguish points corresponding to different underlying locations and to recognize points corresponding to the same location, requiring strong feature descriptor for points, on which deep learning approaches show expertise. Through metric learning, a feature descriptor can be learned such that matching points are close to each other in the feature space, while non-matching points are far away. Approaches following this idea include PPFNet [17], 3DSmoothNet [18], SpinNet [19], and FCGF [20]. Good point matching leads to accurate pose estimation when solving the orthogonal Procrustes problem. Therefore, the error in pose estimation can be used to supervise point matching and feature learning. DCP [21], RPM-Net [22], DGR [23], and 3DRegNet [24] are some of the methods leveraging ground truth pose to supervise point feature learning. In a point cloud, it is likely that a subset of them (e.g. corner points) have more salient features and are easier to be identified than others (e.g. points on a flat surface). They are called keypoints. Keypoints can be matched more reliably, resulting in better pose estimation. In this regard, neural networks are designed to better pick the keypoints (e.g., USIP [25] and SampleNet [26]). Keypoint selection and feature learning may also be considered jointly. Representative works include 3DFeat-Net [27], D3Feat [28], DeepICP [29], and PRNet [30].

The main remaining challenge is that such matching pairs may not exist in the input point clouds in the first place, because point clouds are sparse and a point may not be captured repeatedly by different scans. Soft matching (or many to many) [31, 32] is proposed to address this problem, but it is at best an approximation of the underlying true matching using sparse samples, which can deteriorate the performance when the sparsity increases.

5.2 Correspondence-free point cloud registration

Correspondence-free registration treats a point cloud as a whole rather than a collection of points, requiring a global representation for an entire point cloud. In this way, the limitation brought by the sparsity as mentioned in Sec. 5.1 is circumvented. CVO [33, 34] represents a point cloud as a function in a reproducing kernel Hilbert space, transforming the registration problem to maximizing the inner product of two functions. A series of deep-learning-based methods attempt to extract a global feature embedding for a point cloud and solve the registration by aligning the global features. Examples include PointNetLK [9], PCRNet [10], and Feature-Metric Registration [11]. A limitation of global-feature-based registration is that the nonlinearity of the feature extraction networks leaves us few clues about the structure and properties of the feature space. Therefore these methods rely on iterative local optimization such as the Gauss-Newton algorithm. Consequently, these methods require good initialization to achieve decent results. Our method differentiates from previous work in that the feature extraction network is $\mathrm{SO}(3)$ equivariant, enabling well-behaved optimization in the feature space (detailed in Sec. 3).

5.3 Group equivariant neural networks

Neural networks today mainly preserve symmetry against translation, limiting the capacity of networks to deal with broader transformations presented in data. Researchers have been working on group-equivariant neural networks to address this issue. For example, the $\mathrm{SO}(3)$ group is a classical Lie group and frequently appears in practice [35, 36]. One line of work forms kernels as some steerable functions so that the rotations in the input space can be transformed into rotations in the output space [37, 38, 39]. However, this strategy can limit the expressiveness of the feature extractor since the form of the kernels is constrained.

Another strategy is to *lift* the input space to a higher-dimensional space where the group is contained so that equivariance is obtained naturally [40, 41]. The simplest example is that the group of 2D translations is contained in $\mathbb{R}^2$; therefore, 2D convolution embeds equivariance to 2D translations. However, for each lifted input to K group elements, we need to integrate over the entire group for computing its convolution. While being mathematically sound, this approach involves the use of group $\exp$ and $\log$ maps (for mapping from and to the Lie algebra) and a Monte Carlo approximation to compute the convolution integral over a fixed set of group elements [41, Sections 4.2-4.4].

Cohen and Welling [42] studied the problem of learning the irreducible representations of commutative Lie groups. Kondor and Trivedi [40] studied the generalization of equivariance and convolution in neural networks to the action of compact groups. The latter also discussed group convolutions and their connection with Fourier analysis [35, 36]. In this context, the convolution theorem naturally extends to the group-valued functions and the representation theory [43].

A new design of $\mathrm{SO}(3)$ equivariant neural networks is proposed by Deng et al. [5]. This approach allows incorporating the $\mathrm{SO}(3)$ equivariance property into existing network architectures such as PointNet and DGCNN by replacing the linear, nonlinear, pooling, and normalization layers with their *vector neuron* version. Our equivariant feature learner builds on this simple and useful idea.

5.4 Deep implicit shape modeling

Implicit shape modeling represents a shape as a field on which a function such as an occupancy or signed distance function is defined. The surface is typically defined by the equation $f(x, y, z) = c$ for some constant c. In recent years, deep models are developed to model such implicit function fields, where a network is trained to predict the function value given a query input point coordinate. Some representative works include DeepSDF [44], OccNet [8], NeRF [45].

Implicit functions model a shape continuously, up to the infinite resolution, which offers opportunities in registration to circumvent the challenge in data association due to sparsity in point clouds.

DI-Fusion [46] leverages deep implicit models to do surface registration and mapping by optimizing the position of the query points to minimize the absolute SDF value. iNeRF [47] performs RGB-only registration through deep implicit models by minimizing the pixel residual between observed images and neural-rendered images. In this work, we also exploit the continuous nature of the deep implicit model but by using the learned latent feature instead of the decoded function field.

6 Conclusion

In this paper, we presented a correspondence-free rotational registration method for point clouds. The method is built upon the developments in equivariant neural networks and implicit shape representation learning. We construct a feature space where the rotational relations in Euclidean space are preserved, in which registration can be done efficiently. We circumvent the need for data association and solve the rotation in closed form, achieving low errors independent of initial rotation angles. Furthermore, we leverage implicit representation learning to enhance the network robustness against non-ideal cases where the points are subject to noise.

Nevertheless, the robustness of our method against density variations still has room for improvement. Furthermore, there are two open issues unsolved. First, generalizing the $\mathrm{SO}(3)$-equivariant neural network to deal with translation so that the registration of rotation and translation can be solved together. Secondly, generalizing the registration to larger-scale problems to handle outdoor LiDAR or stereo camera scans. These problems are left for future studies.

References

[1] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Am. A*, 5(7):1127–1135, Jul 1988. doi:10.1364/JOSAA.5.001127. URL http://josaa.osa.org/abstract.cfm?URI=josaa-5-7-1127.

[2] R. W. Wedderburn. Quasi-likelihood functions, generalized linear models, and the gauss—newton method. *Biometrika*, 61(3):439–447, 1974.

[3] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016.

[4] Y. Wang, Y. Sun, Z. Liu, S. Sarma, M. Bronstein, and J. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 38, 01 2018. doi:10.1145/3326362.

[5] C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. Guibas. Vector neurons: A general framework for SO(3)-equivariant networks. *arXiv preprint arXiv:2104.12229*, 2021.

[6] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[7] D. Stutz and A. Geiger. Learning 3d shape completion under weak supervision. *CoRR*, abs/1805.07290, 2018. URL http://arxiv.org/abs/1805.07290.

[8] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[9] Y. Aoki, H. Goforth, R. A. Srivatsan, and S. Lucey. PointNetLK: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2019.

[10] V. Sarode, X. Li, H. Goforth, Y. Aoki, R. A. Srivatsan, S. Lucey, and H. Choset. Pcrnet: Point cloud registration network using pointnet encoding. *ArXiv*, abs/1908.07906, 2019.

[11] X. Huang, G. Mei, and J. Zhang. Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11366–11374, 2020.

[12] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb 1992. ISSN 1939-3539. doi:10.1109/34.121791.

[13] A. Censi. An ICP variant using a point-to-line metric. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 19–25. IEEE, 2008.

[14] Y. Chen and G. G. Medioni. Object modeling by registration of multiple range images. *Image Vision Comput.*, 10(3):145–155, 1992.

[15] N. J. Mitra, N. Gelfand, H. Pottmann, and L. Guibas. Registration of point cloud data from a geometric optimization perspective. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, SGP '04, page 22–31, New York, NY, USA, 2004. Association for Computing Machinery. doi:10.1145/1057432.1057435.

[16] A. Segal, D. Haehnel, and S. Thrun. Generalized-ICP. In *Robotics: science and systems*, volume 2, page 435. Seattle, WA, 2009.

[17] H. Deng, T. Birdal, and S. Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 195–205, 2018.

[18] Z. Gojcic, C. Zhou, J. D. Wegner, and A. Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5545–5554, 2019.

[19] S. Ao, Q. Hu, B. Yang, A. Markham, and Y. Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[20] C. Choy, J. Park, and V. Koltun. Fully convolutional geometric features. In *ICCV*, 2019.

[21] Y. Wang and J. M. Solomon. Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3523–3532, 2019.

[22] Z. J. Yew and G. H. Lee. Rpm-net: Robust point matching using learned features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11824–11833, 2020.

[23] C. Choy, W. Dong, and V. Koltun. Deep global registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[24] G. D. Pais, S. Ramalingam, V. M. Govindu, J. C. Nascimento, R. Chellappa, and P. Miraldo. 3DRegNet: A deep neural network for 3d point registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7193–7203, 2020.

[25] J. Li and G. H. Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 361–370, 2019.

[26] I. Lang, A. Manor, and S. Avidan. SampleNet: Differentiable point cloud sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7578–7588, 2020.

[27] Z. J. Yew and G. H. Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *ECCV*, 2018.

[28] X. Bai, Z. Luo, L. Zhou, H. Fu, L. Quan, and C.-L. Tai. 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*, pages 6359–6367, 2020.

[29] W. Lu, G. Wan, Y. Zhou, X. Fu, P. Yuan, and S. Song. Deepicp: An end-to-end deep neural network for 3d point cloud registration, 05 2019.

[30] Y. Wang and J. M. Solomon. Prnet: Self-supervised learning for partial-to-partial registration. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8814–8826. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9085-prnet-self-supervised-learning-for-partial-to-partial-registration.pdf.

[31] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness. New algorithms for 2D and 3D point matching: Pose estimation and correspondence. *Pattern recognition*, 31(8):1019–1031, 1998.

[32] S. Granger and X. Pennec. Multi-scale EM-ICP: A fast and robust approach for surface registration. In *Proceedings of the European Conference on Computer Vision*, pages 418–432, 2002.

[33] M. Ghaffari, W. Clark, A. Bloch, R. M. Eustice, and J. W. Grizzle. Continuous direct sparse visual odometry from RGB-D images. In *Proceedings of the Robotics: Science and Systems Conference*, Freiburg, Germany, June 2019.

[34] R. Zhang, T.-Y. Lin, C. E. Lin, S. A. Parkison, W. Clark, J. W. Grizzle, R. M. Eustice, and M. Ghaffari. A new framework for registration of semantic point clouds from stereo and RGB-D cameras. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1–8, 2021.

[35] G. S. Chirikjian and A. B. Kyatkin. *Engineering applications of noncommutative harmonic analysis: with emphasis on rotation and motion groups*. CRC Press, 2001. doi:10.1201/9781420041767.

[36] G. S. Chirikjian. *Stochastic models, information theory, and Lie groups, volume 2: Analytic methods and modern applications*, volume 2. Springer Science & Business Media, 2009.

[37] T. S. Cohen and M. Welling. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016.

[38] C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018.

[39] F. B. Fuchs, D. E. Worrall, V. Fischer, and M. Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *arXiv preprint arXiv:2006.10503*, 2020.

[40] R. Kondor and S. Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International Conference on Machine Learning*, pages 2747–2755. PMLR, 2018.

[41] M. Finzi, S. Stanton, P. Izmailov, and A. G. Wilson. Generalizing convolutional neural networks for equivariance to Lie groups on arbitrary continuous data. In *International Conference on Machine Learning*, pages 3165–3176. PMLR, 2020.

[42] T. Cohen and M. Welling. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pages 1755–1763. PMLR, 2014.

[43] B. Hall. *Lie groups, Lie algebras, and representations: an elementary introduction*, volume 222. Springer, 2015.

[44] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[45] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020.

[46] J. Huang, S.-S. Huang, H. Song, and S.-M. Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. *arXiv preprint arXiv:2012.05551*, 2020.

[47] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T.-Y. Lin. inerf: Inverting neural radiance fields for pose estimation. *arXiv preprint arXiv:2012.05877*, 2020.