PCRP: UNSUPERVISED POINT CLOUD OBJECT RETRIEVAL AND POSE ESTIMATION

Pranav Kadam[1], *Qingyang Zhou*[1], *Shan Liu*[2], *C.-C. Jay Kuo*[1]

University of Southern California, Los Angeles, CA, USA[1]
Tencent Media Lab, Palo Alto, CA, USA[2]

ABSTRACT

An unsupervised point cloud object retrieval and pose estimation method, called PCRP, is proposed in this work. It is assumed that there exists a gallery point cloud set that contains point cloud objects with given pose orientation information. PCRP attempts to register the unknown point cloud object with those in the gallery set so as to achieve content-based object retrieval and pose estimation jointly, where the point cloud registration task is built upon an enhanced version of the unsupervised R-PointHop method. Experiments on the ModelNet40 dataset demonstrate the superior performance of PCRP in comparison with traditional and learning based methods.

***Index Terms*—** Unsupervised learning, pose estimation, object retrieval, successive subspace learning.

1. INTRODUCTION

Content-based point cloud object retrieval and category-level point cloud object pose estimation are two important tasks of point cloud processing. For the former, one can find similar objects from the gallery set, which can provide more information about the unknown object. For the latter, the goal is to estimate the 6-DOF pose of a 3D object comprising of rotation ($R \in SO(3)$) and translation ($t \in \mathbb{R}^3$) , with respect to a chosen reference. The pose information can facilitate downstream tasks such as object grasping, obstacle avoidance and path planning for robotics. In a typical scene understanding problem using data from range sensors or a depth camera, this problem would arise after a 3D detection algorithm has successfully localized and labeled the objects present in the point cloud scan. An unsupervised point cloud object retrieval and pose estimation method, called PCRP, is proposed in this work.

The R-PointHop method was recently proposed by Kadam *et al.*[1] for point cloud registration. Although being unsupervised, it offers competitive performance with respect to other supervised learning methods. In this paper, we extend R-PointHop to the context of point cloud object retrieval and pose estimation against a gallery point cloud set, which contains point cloud objects with known pose orientation information. Point cloud retrieval has been researched for quite some time in terms of retrieving a similar object from a database or aggregating local feature descriptors for recognizing places in the wild. Yet, retrieving objects with pose variations is less investigated. Here, we show how R-PointHop features can be reused to retrieve a similar point cloud object. Registration of two similar objects (potentially with partial overlap), which was the focus of R-PointHop, has been widely studied. Although being a related problem, estimating the pose of a single object addressed in this work is less explored. We analyze several bottlenecks of R-PointHop and propose modifications to enhance its performance for pose estimation.

This work was supported by Tencent Media Lab.

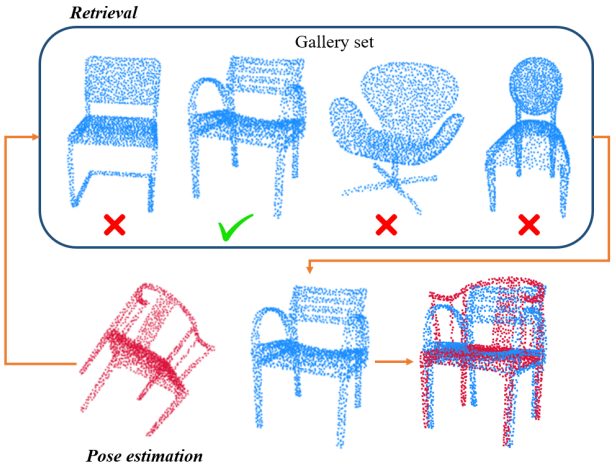

Fig. 1. Summary of the proposed PCRP method. First, a similar object to the input query object (in red) is retrieved from the gallery set (top row). Then, the query object is registered with the retrieved object (bottom row) to estimate its pose.

Built upon enhanced R-PointHop, PCRP registers the unknown point cloud object with those in the gallery set to achieve content-based object retrieval and pose estimation jointly. As shown in Fig. 1, PCRP consists of "object retrieval" and "pose estimation" two functions. For object retrieval, it first aggregates the pointwise features learned from R-PointHop based on VLAD (Vector of Locally Aggregated Descriptors) [2] to obtain a global point cloud feature vector and then use it to retrieve a similar pre-aligned object from the gallery set. For pose estimation, the 6-DOF pose of the query object is found by registering it with the retrieved object. Experiments on the ModelNet40 dataset demonstrate the superior performance of PCRP in comparison with traditional and learning based methods.

This work has two main contributions. First, we extend R-PointHop, which was originally designed for point cloud registration, to object retrieval and pose estimation. We show how features derived from R-PointHop can be aggregated to yield a global feature descriptor for object retrieval and reuse point features for pose estimation. Second, we propose ways to modify the attribute representation in R-PointHop and make it more general. As a result, any traditional point local descriptors, such as FPFH [3] and SHOT [4], can be adopted by R-PointHop. The rest of the paper is organized as follows. Related previous work is reviewed in Sec. 2. The PCRP method is proposed in Sec. 3. Experimental results are shown in Sec. 4. Finally, concluding remarks are given in Sec. 5.

2. REVIEW OF RELATED WORK

Point cloud processing includes classification, registration, segmentation, retrieval, pose estimation, etc. There has been major advancement in point cloud processing due to deep learning. PointNet [5] employed an MLP-based permutation invariant network for point cloud classification and segmentation. It is followed by a series of work [6, 7, 8]. Besides permutation invariance, features invariant to point cloud rotation [9, 10] are desirable for applications such as data association, registration and classification of unaligned objects. Learning based global registration methods [11, 12, 1] outperform traditional handcrafted descriptors such as FPFH [3] and SHOT [4] in registration performance and ICP-based [13] methods in object registration. Recently, point cloud equivariant networks [14, 15, 16] show impressive performance for object pose estimation, retrieval and classification under different poses. Furthermore, the learning based methods can be successfully applied to complex indoor scene registration and visual odometry estimation.

As an alternative to deep learning, successive subspace learning (SSL) has been proposed for point cloud processing. Its potential was demonstrated in PointHop [17], PointHop++ [18], SPA [19], R-PointHop [1] and GPCO [20]. The unsupervised feature learning in SSL consists of attribute construction, neighborhood expansion, and dimensionality reduction using the Saab transform. R-PointHop extracts a local reference frame for every point and builds the local feature by aligning neighborhood points to it. The neighborhood grows successively and short-, mid-, and long-range point information is captured. Later, a set of point correspondences are found and the 3D transformation is estimated using singular value decomposition of the covariance matrix of corresponding points.

One limitation of R-PointHop and other exemplary deep networks for pair-wise registration [11, 12, 21] is the assumption that a pre-aligned reference object is available, which is the same instance of the input object whose pose is to be estimated. However, such an assumption may not hold in practice and, as a result, the pose estimation problem cannot be solved by registration. One way to avoid this difficulty is to retrieve a similar pre-aligned object from a gallery set first and then estimate the object pose by registering it with the retrieved object as done in CORSAIR [22]. CORSAIR uses the bottleneck layer representation of a registration network to train another network for retrieval using metric learning. It is worthwhile to mention that some pose estimation methods are reference object free [16, 15]. They adopt point cloud equivariant networks as the backbone and do not use object retrieval for pose estimation.

Point cloud retrieval has been studied extensively due to its rich applications. Noteworthy work includes 3D ShapeNets [23] for shape retrieval and PointNetVLAD [24] for place recognition. Our retrieval idea is inspired by PointNetVLAD, which aggregates pointwise features from PointNet using the VLAD feature [2]. Their VLAD feature is obtained by taking a weighted average of point features, where weights are jointly trained with the PointNet network under supervised learning. In contrast, our R-PointHop feature is learned without supervision. We should emphasize that most retrieval methods assume that the query object and objects from the gallery set are pre-aligned. Yet, in the context of joint object retrieval and pose estimation, this assumption does not hold. It is essential to develop a retrieval method, where the query object can possess any arbitrary rotation. Our work is uniquely positioned in addressing two challenges at one shot: 1) pose estimation without identical object instances, and 2) object retrieval without pre-alignment. Furthermore, we do not use deep leaning but successive subspace learning.

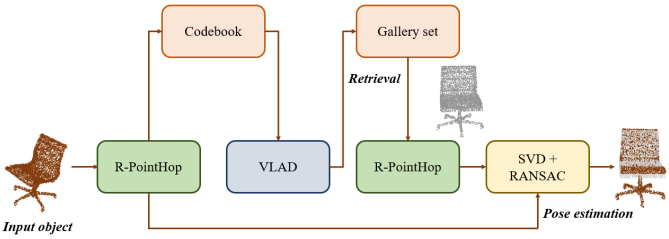

Fig. 2. An overview of the proposed PCRP method: 1) pointwise feature extraction from the input point cloud using R-PointHop, 2) aggregation of point features into a global VLAD representation and retrieval of similar objects from the gallery set based on similarity of VLAD representations, and 3) the pose of the input is estimated by registering the query object with the retrieved object.

3. PROPOSED PCRP METHOD

As shown in Fig. 2, the PCRP method consists of the following three stages. First, features of every point in the input point cloud are extracted using R-PointHop. Second, features of all points are aggregated into a global descriptor using the VLAD method. The nearest neighbor search is then used to retrieve a similar pre-aligned object from the gallery set. Finally, the input object is registered to the retrieved object to obtain the 6-DOF pose. Each of them is elaborated below.

3.1. Feature Extraction

In the R-PointHop originally proposed in [1], point attributes are constructed by partitioning the 3D space around a point into eight octants using three orthogonal directions given by the local reference frame. The mean of points in each octant is concatenated to get a 24D attribute vector. Yet, we observe that the 24D attribute vector is sensitive to noise and unable to capture complex local surface patterns for distinction. A modified version that appends point coordinates with eigen features was used for indoor scene registration and odometry [20]. Actually, histogram-based point descriptors such as SHOT (Signature Histogram of Orientations) [4] and FPFH (Fast Point Feature Histogram) [3] have been widely used to describe the local surface geometry. We may leverage them as well. One drawback of histogram-based descriptors is that they cannot capture the far-distance information since they have a single scale only.

We propose to integrate histogram-based local descriptors with R-PointHop, which has a multi-scale representation capability, to get a new descriptor. Specifically, we replace the octant-based mean-coordinates attributes in the original R-PointHop with the FPFH descriptor in the first hop. The rest is kept the same as the original R-PointHop. That is, the Saab transform is used to get the first-hop spectral representation, and subsequent hops still involve attribute construction by partitioning the 3D space into eight octants and getting 8D attributes for each spectral component. We should emphasize that FPFH is invariant to rotations so that the rotation-invariant property of R-PointHop is preserved without any other adjustment. The output of the first-hop stemming from FPFH is more powerful than the original R-PointHop design. We give the new descriptor a name - FR-PointHop. This modification enriches R-PointHop since it can take any local rotation-invariant feature representation and generalize it to a multi-scale descriptor using the standard R-PointHop pipeline.

3.2. Feature Aggregation

For an input point cloud, we extract features of each point using FR-PointHop. Features of all points need to be aggregated to yield a global descriptor for object retrieval. One choice of aggregation is to use global max/mean pooling. It has been widely adopted in point cloud classification. However, global pooling is not a good choice since point features obtained by FR-PointHop only cover the information of a local neighborhood. In the retrieval literature, Bag of Words (BoW) and vectors of locally aggregated descriptors (VLAD) [2] are popular methods in aggregating local features such as SIFT. Here, we adopt VLAD [2] to aggregate point descriptors obtained by R-PointHop to yield a global feature vector that is suitable for retrieval. The global feature aggregation process is stated below.

The first step is to generate a codebook of k codewords of d dimensions, where d is the feature dimension. The k-means clustering algorithm is used to achieve this objective. Learned point features from the training data are used to form clusters, whose centroids are computed. The k centroids represent k codewords in the codebook. Given an input point cloud, its global VLAD feature vector is calculated as follows. The feature vector of each point is first assigned to the nearest codeword based on the shortest distance criterion. Next, for each point, the difference between its descriptor and the assigned codeword is calculated, which represents the error vector in the feature space. Then, the differences with respect to the same codeword are added together. Finally, error sums of all codewords are concatenated to get the VLAD feature. For a d-dimensional feature vector, the VLAD feature is of dimension $k \times d$, where k is the codeword number.

In the training process, we use the generated codebook to precompute the VLAD features of point cloud objects in the gallery set. They are stored along with the codebook. In the inference stage, the query object is first passed through FR-PointHop to extract pointwise features. Then, its VLAD feature is calculated using the same codebook and compared with the VLAD features of point cloud objects from the gallery set. The nearest neighbor search is used to retrieve the best matching point cloud. It is worthwhile to mention that, due to the rotation invariant nature of FR-PointHop, point features and, hence, VLAD features are invariant with object's pose, which facilities retrieval in presence of pose variations.

3.3. Pose Estimation

Once an object from the gallery set is retrieved, the next step is to register the query and the retrieved objects. The process closely resembles the registration task of R-PointHop with additional aid of the object symmetry information. The pointwise features extracted from the query object for retrieval are reused here. For objects in the gallery set, we only store their VLAD descriptors rather than their pointwise features due to the high memory cost of the latter. Yet, pointwise features of the retrieved object can be computed again using FR-PointHop at run time. The cost is manageable since it is done for one retrieved object. Afterwards, corresponding points between query and retrieved objects are found using the nearest neighbor criterion in the feature space.

We exploit the object symmetry information to limit the correspondence search region. 3D objects often possess different forms of symmetry. For example, chair objects have a planar symmetry. To avoid mismatched correspondences arising due to object symmetry, correspondences are constrained to be among disjoint sets of points. For every object, we divide its points into two disjoint sets using its principal components.

First, we calculate 1D moments of the points projected along each principal direction about the origin. Then, we take the absolute difference between sum of moments of positive coordinates and negative coordinates. Then, the principal component with the least absolute difference is used to divide the points in disjoint sets. The main intuition is to find an axis along which the projection of points is most symmetric. The search space for this axis is restricted to object's principal components for simplicity. Yet, it yields reasonable results as seen from top row of Fig. 3. Afterwards, the point correspondences are found in each disjoint set (see the bottom row of Fig. 3) and then concatenated.

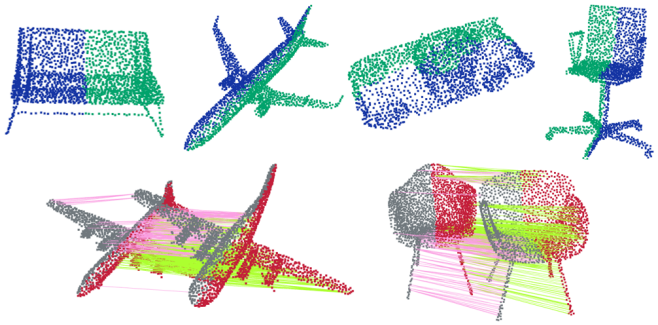

Fig. 3. (Top) Illustration of partitioning of point clouds into two symmetrical parts and (Bottom) point correspondences between symmetric parts.

Once the point correspondence is built, we use the orthogonal procrustes method [25], which is based on the singular value decomposition (SVD) of the data covariance matrix, to estimate the rotation and translation that aligns the query object to the retrieved object. This part is identical with that in R-PointHop. The transformation is optimal in the sense that it achieves the least sum of squared errors between matching points after alignment. RANSAC [26] can be used to get a more robust solution and avoid the noise effect in building correspondences. Since objects in the gallery set are pre-aligned, the obtained rotation and translation gives the 6-DOF pose of the input query object.

4. EXPERIMENTS

We use point cloud objects from the ModelNet40 dataset [23] in all experiments. For retrieval and pose estimation, we focus on four object categories: airplane, chair, sofa, and car. For every object, the dataset consists of 2048 points sampled from the surface. We use the original train/test split. The experiments are divided into three parts as discussed below. For R-PointHop, we use two hops and adjust the energy threshold so as to have a 200-D point feature. The farthest point sampling strategy is used after the first hop where the point cloud is downsampled by half.

4.1. Object Registration with FR-PointHop

After replacing the conventional R-PointHop attributes with the FPFH descriptor, we see an improvement in all object registration tasks of FR-PointHop over R-PointHop. A random rotation and translation is applied to the input object. For the challenging case of partial object registration (under the assumption that the partial reference point cloud is available), we show the rotation and translation errors Table 1 in terms of the mean squared error (MSE), the root

mean squared error (RMSE) and the mean absolute error (MAE). We conduct performance benchmarking with SPA [19] (SSL-based), FGR [27] (FPFH-based), and PRNet [21]. Clearly, the modified feature representation in FR-PointHop is favorable even in presence of the reference object.

Table 1. Performance comparison of object registration.

	Rotation error (in degree)			**Translation error**		
Method	**MSE**	**RMSE**	**MAE**	**MSE**	**RMSE**	**MAE**
SPA [19]	229.09	15.13	8.22	0.0019	0.0435	0.0089
FGR [27]	126.29	11.24	2.83	0.0009	0.0300	0.0080
PR-Net [21]	10.24	3.12	1.45	0.0003	0.0160	0.0100
R-PointHop	2.75	1.66	0.35	**0.0002**	**0.0149**	0.0008
FR-PointHop	**2.68**	**1.64**	**0.33**	**0.0002**	**0.0149**	**0.0007**

4.2. Object Retrieval with PCRP

We use Precision@M and the chamfer distance as two evaluation metrics for object retrieval. For every query object, we generate its ground truth by rank-ordering objects in the gallery set based on the least chamfer distance. The chamfer distance is calculated as the sum of Euclidean distances of every point in one point cloud to its nearest point in the other point cloud. The Precision@M is an average measure of the number of top M retrieved objects that match with the top M objects from its ground truth. We expect a higher Precision@M score while a lower chamfer distance for good retrieval methods.

Ten codewords are used in the VLAD implementation. we report Precision@10 scores and Top-1 chamfer distances for two cases in Table 2. First, the query object is aligned with those in the gallery set. Second, a uniform random rotation and translation is applied to the query object so that it has an arbitrary pose. We provide comparisons with PointNet [5] (an exemplary deep learning method), CORSAIR [22] (on similar lines to our work, but supervised), PointHop [17] (SSL-based), and FPFH [3]. For PointNet and PointHop, globally pooled features are adopted for retrieval. For FPFH, we aggregate point features using VLAD. Furthermore, we replace our VLAD aggregated feature with max pooling and report the performance separately.

Results in Table 2 show the superiority of PCRP over others. For PointNet and PointHop, we see a drop in performance in the case of arbitrary poses. We expect similar performance for other methods that do not take pose variations into account. Since CORSAIR, FPFH and PCRP use pose invariant feature representations, their performance is robust against pose variation. PCRP is the best among the three. Moreover, aggregating local features with max pooling in PCRP degrades the performance significantly, thereby justifying the inclusion of VLAD in PCRP.

Table 2. Comparison of point cloud retrieval performance.

	Pre-aligned objects		Arbitrary poses	
Method	Precision@10 (%)	Top-1 Chamfer distance	Precision@10 (%)	Top-1 Chamfer distance
PointNet [5]	60.66	0.121	53.40	0.145
PointHop [17]	58.23	0.129	19.71	0.211
FPFH [3]	53.23	0.164	52.12	0.160
CORSAIR [22]	61.28	0.106	61.24	**0.107**
PCRP (max pool)	43.23	0.147	41.89	0.131
PCRP (VLAD)	**63.23**	**0.101**	**63.07**	0.111

4.3. Object Pose Estimation with PCRP

We report mean and median rotation errors between the predicted and ground truth pose for all four object classes in Table 3. For performance benchmarking, we select ICP [13] and FGR [27] among traditional methods for which we provide the template point cloud. For learning-based methods, we selected Chen *et al.*[15] (supervised) and Li *et al.*[16] (self-supervised) two methods, which are based on equivariant networks. Finally, CORSAIR [22] is also included.

Table 3. Mean and median rotation errors in degrees.

	Chair		Airplane		Car		Sofa	
Method	Mean	Median	Mean	Median	Mean	Median	Mean	Median
ICP [13]	88.92	96.28	8.11	1.22	22.76	2.94	39.00	9.69
FGR [27]	22.10	6.04	6.84	3.33	18.44	2.69	9.97	2.36
CORSAIR [22]	13.99	4.58	8.09	3.43	12.09	2.13	9.12	3.24
Chen *et al.*[15]	**8.56**	**3.87**	**3.35**	**1.12**	**9.48**	**1.85**	**4.76**	**1.56**
Li *et al.*[16]	7.05	4.55	23.09	1.66	17.24	2.13	8.87	3.22
PCRP	14.42	4.24	2.98	1.65	11.22	2.11	8.84	2.29

From mean and median rotation errors, PCRP is significantly better than ICP which is only good for local registration. It also outperforms FGR and CORSAIR. Its performance is slightly inferior to the method of Li *et al.*[16]. In all the cases, the mean rotation errors are higher than the median error. We study the distribution of rotation errors across all point clouds and observe that the higher error is only due to a large registration error in only a few point clouds. Actually, after plotting the CDF of the error for all point clouds, more than 90% of test samples have a rotation error less than 5 degrees.

It is worthwhile to point out the advantages of PCRP. First, it combines unsupervised feature learning with established non-learning-based pose estimation using point correspondences. The training time is typically less than 30 minutes in building the Saab kernels and the VLAD codebook. The FR-PointHop model size is only 230kB along with 1.6MB to store the VLAD features and the codebook. In contrast, we find that the model size of the exemplary works is 30MB for Chen *et al.*and 72MB for Li *et al.*. This clearly highlights that PCRP would be favorable in resource constrained occasions.

Further investigation into failure cases reveals that most of them occur when a similar matching point cloud is not available in the database for retrieval. This is a bottleneck of PCRP. Under the assumption that a similar object is present in the gallery set, it can estimate the pose accurately. One way to filter out query samples automatically is to compare the Chamfer distance to the best retrieved object. If the Chamfer distance is above a certain threshold, it cannot be treated as a reliable result.

5. CONCLUSION

An unsupervised feature learning method for point cloud retrieval and pose estimation, called PCRP, was proposed in this paper. PCRP estimates the 6-DOF pose comprising of rotation and translation of a point cloud object using a similar pre-aligned object of the same object category. It uses the R-PointHop method to extract point features from 3D point cloud objects. The features are aggregated into a global descriptor using VLAD and used to retrieve similar pre-aligned objects. Finally, the point features of the query and retrieved point clouds are used to estimate the 3D pose of the query point cloud using registration.

6. REFERENCES

[1] Pranav Kadam, Min Zhang, Shan Liu, and C-C Jay Kuo, "R-PointHop: A Green, Accurate and Unsupervised Point Cloud Registration Method," *arXiv preprint arXiv:2103.08129*, 2021.

[2] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez, "Aggregating local descriptors into a compact image representation," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3304–3311.

[3] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 3212–3217.

[4] Federico Tombari, Samuele Salti, and Luigi Di Stefano, "Unique signatures of histograms for local surface description," in *European conference on computer vision*. Springer, 2010, pp. 356–369.

[5] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[6] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.

[7] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon, "Dynamic graph cnn for learning on point clouds," *Acm Transactions On Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[8] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen, "Pointcnn: Convolution on x-transformed points," *Advances in neural information processing systems*, vol. 31, pp. 820–830, 2018.

[9] Haowen Deng, Tolga Birdal, and Slobodan Ilic, "Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 602–618.

[10] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser, "The perfect match: 3d point cloud matching with smoothed densities," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5545–5554.

[11] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey, "Pointnetlk: Robust & efficient point cloud registration using pointnet," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7163–7172.

[12] Yue Wang and Justin M Solomon, "Deep closest point: Learning representations for point cloud registration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3523–3532.

[13] Paul J Besl and Neil D McKay, "Method for registration of 3-D shapes," in *Sensor fusion IV: control paradigms and data structures*. International Society for Optics and Photonics, 1992, vol. 1611, pp. 586–606.

[14] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas Guibas, "Vector Neurons: A General Framework for SO (3)-Equivariant Networks," *arXiv preprint arXiv:2104.12229*, 2021.

[15] Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill, "Equivariant Point Network for 3D Point Cloud Analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14514–14523.

[16] Xiaolong Li, Yijia Weng, Li Yi, Leonidas Guibas, A Lynn Abbott, Shuran Song, and He Wang, "Leveraging SE (3) Equivariance for Self-Supervised Category-Level Object Pose Estimation," *arXiv preprint arXiv:2111.00190*, 2021.

[17] Min Zhang, Haoxuan You, Pranav Kadam, Shan Liu, and C-C Jay Kuo, "PointHop: An explainable machine learning method for point cloud classification," *IEEE Transactions on Multimedia*, vol. 22, no. 7, pp. 1744–1755, 2020.

[18] Min Zhang, Yifan Wang, Pranav Kadam, Shan Liu, and C-C Jay Kuo, "Pointhop++: A lightweight learning model on point sets for 3d classification," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 3319–3323.

[19] Pranav Kadam, Min Zhang, Shan Liu, and C-C Jay Kuo, "Unsupervised Point Cloud Registration via Salient Points Analysis (SPA)," in *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2020, pp. 5–8.

[20] Pranav Kadam, Min Zhang, Shan Liu, and C-C Jay Kuo, "GPCO: An Unsupervised Green Point Cloud Odometry Method," *arXiv preprint arXiv:2112.04054*, 2021.

[21] Yue Wang and Justin M Solomon, "PRNet: Self-Supervised Learning for Partial-to-Partial Registration," in *Advances in Neural Information Processing Systems*, 2019, pp. 8814–8826.

[22] Tianyu Zhao, Qiaojun Feng, Sai Jadhav, and Nikolay Atanasov, "CORSAIR: Convolutional Object Retrieval and Symmetry-AIded Registration," *arXiv preprint arXiv:2103.06911*, 2021.

[23] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[24] Mikaela Angelina Uy and Gim Hee Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4470–4479.

[25] Peter H Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[26] Martin A Fischler and Robert C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[27] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun, "Fast global registration," in *European conference on computer vision*. Springer, 2016, pp. 766–782.