

# SGPCR: Spherical Gaussian Point Cloud Representation and its Application to Object Registration and Retrieval

Driton Salihu    Eckehard Steinbach

Chair of Media Technology and Munich Institute of Robotics and Machine Intelligence (MIRMI)  
Technical University of Munich, School of Computation, Information and Technology

Department of Computer Engineering

{driton.salihu, eckehard.steinbach}@tum.de



Figure 1: SGPCR generates a rotation-equivariant representation of point clouds for point cloud registration and retrieval. Compared to DeepUME [31] and DeepGMR [51], SGPCR results in higher alignment quality on noisy ModelNet40 [49] data with no points shared between source and target point cloud (left). Further, compared to MCSS [21], SGPCR increases the registration quality of scan-to-CAD procedures on the Scan2CAD [3] dataset (right).

## Abstract

Retrieving and aligning CAD models from databases with scanned real-world point clouds remains an important topic for 3D reconstruction. Due to zero point-to-point correspondences between the sampled CAD model and the scanned real-world object, an information-rich representation of point clouds is needed. We propose SGPCR, a novel method for representing 3D point clouds by Spherical Gaussians for efficient, stable, and rotation-equivariant representation. We also propose a rotation-invariant convolution to improve the representation quality through a trainable optimization process. In addition, we demonstrate the strengths of SGPCR-based point cloud representation using the fundamental challenge of shape retrieval and point cloud registration on point clouds with zero point-to-point correspondences. Under these conditions, our approach improves registration quality by reducing chamfer distance by up to 90% and rotation root mean square error by up to 86% compared to the state of the art. Furthermore, the proposed SGPCR is used for one-shot shape retrieval and registration and improves retrieval precision by up to 58% over comparable methods.

## 1. Introduction

The recreation of 3D indoor environments is an important aspect for a variety of 3D content applications. For the geometry of these environments, 3D point clouds are often used as the underlying representation. While the accuracy of point clouds generated by mobile 3D sensing devices (e.g., Intel RealSense, Navvis VLX, Microsoft Kinect, Mobile Phones) has improved in recent years, missing surface information limits its usability in many applications such as virtual reality and augmented reality. To address this issue, 3D reconstruction has become a popular process for transforming raw point clouds into surface representations.

The existing literature approaches the 3D reconstruction problem by either surface reconstruction [12, 23, 35, 16] or scan-to-CAD registration methods [3, 24, 29, 4]. Due to occlusions and unavoidable inaccuracies of scanners and/or human operators, surface reconstruction methods may lead to more holes and lower precision in the reconstruction of specific objects compared to scan-to-CAD methods [29]. In addition, by using existing CAD objects from a database, movements, rearrangements, and removal of objects from a scene can easily be adjusted. Hence, in this work, we

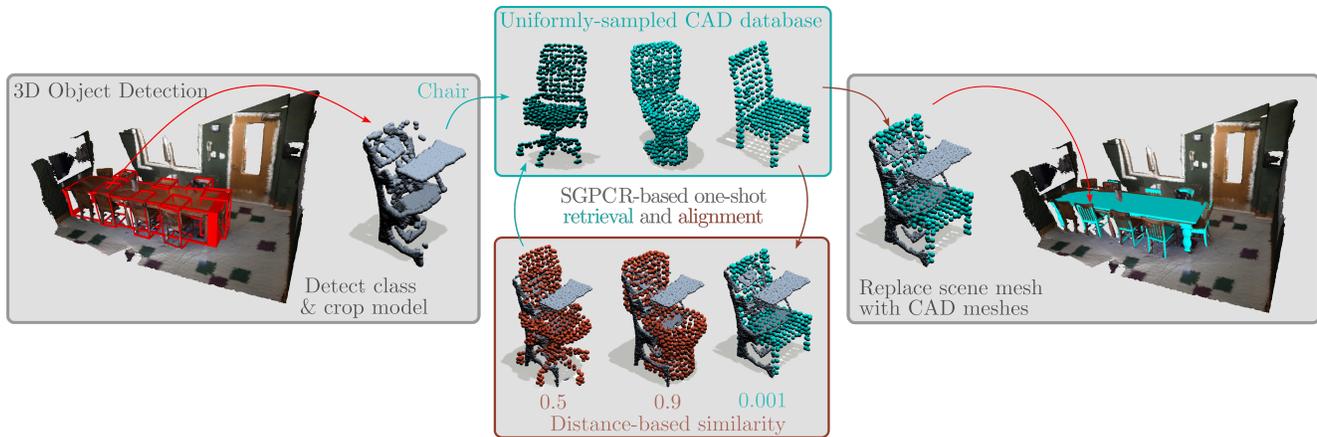


Figure 2: Based on already detected 3D objects of a 3D indoor environment, SGPCR-based object retrieval aims to retrieve uniformly sampled CAD objects from a database (blue) and align (red) them simultaneously to synthetic or real-world (gray) models with zero point-to-point correspondences. The estimated transformation is used on the CAD mesh to create a surface representation of the scanned environment.

concentrate on improving 3D reconstruction based on the registration of CAD models with the raw 3D sensor data, as shown in the qualitative results of Figure 1.

In general, the pipeline of any scan-to-CAD method can be structured into three steps: i) 3D Object Detection/ Semantic Instance Segmentation [20, 37, 22, 36], ii) Shape Retrieval [54, 27, 8, 14], and iii) Point Cloud Registration [51, 31, 47, 19, 34]. As 3D Object Detection/ Semantic Segmentation has been studied extensively [32, 30, 55, 39, 33], this work adopts existing solutions for this step.

In a real-world scan-to-CAD registration task, **shape retrieval** and **point cloud registration** steps are usually performed on 3D point clouds with zero point-to-point correspondences, different densities and varying scaling. This is defined as cross-source registration and retrieval. It occurs in situations in which object models are scanned or created, using different sensors or with different occlusions. These variations require a stable and invariant representation of the point cloud.

This work introduces a novel point cloud representation using **Spherical Gaussians** (SGs) for both, cross-source retrieval and registration, shown in Figure 2. By the nature of SGs, the representation is rotation-equivariant similar to [17] and is thus a good choice for cross-source point cloud registration. Further, SGs are represented on a 3D sphere with a clearly defined lobe axis, amplitude and lobe sharpness. As a result, the transformation between two point clouds can be estimated by finding the rotation for which the correlation between the lobes is the highest. By exploiting the inherent characteristics of SGs, we are able to improve the quality of cross-source point cloud registration and retrieval.

Our contributions can be summarized as follows

- We introduce a novel rotation-equivariant representation of point clouds through SGs.
- We exploit the properties of SGs to propose a novel rotation-invariant convolution, which can be used for registration and retrieval tasks.
- We design a point cloud registration pipeline for 6 and 9 degrees of freedom on highly noisy point clouds with zero point-to-point correspondences.
- We show that the proposed point cloud representation allows for one-shot retrieval and registration.

## 2. Related Work

Tsai [42] first introduced the concept of Gaussian spherical radial basis functions, which was later termed **Spherical Gaussians** (SGs). Initially, SGs were developed for the illumination of rendered objects in simulated static scenes. For this, SGs give an efficient approximation of the rendering equation to estimate lighting and specular BRDFs [52]. To the best of our knowledge, we are the first to propose the use of SG Mixture Models to represent point clouds for scan-to-CAD alignment applications.

VoteNet [37] introduced voting to detect and classify 3D objects in point clouds. By processing the point cloud through PointNet [38], centers of objects are estimated. Each point that the network assumes corresponds to an object is assigned to one of the voted centers. We use VoteNet to predict classes and bounding boxes from 3D indoor environments such as ScanNet [15]. The resulting 3D model point clouds are used for CAD model-based cross-source retrieval and alignment.

## 2.1. Point Cloud Registration

Point Cloud Registration has been a long-standing research topic [7, 11]. With PointNet [38], extracting features from point clouds improved significantly. One of the earlier works, PointNetLK [1] leveraged these improvements for more accurate registration. RGM [19] proposed the use of a graph-based method, thus further using structural information to reduce the effect of outliers. The introduction of Transformers [44] led to further improvements in point cloud registration, as demonstrated by DCP [47] and DeepUME [31]. In contrast, our approach is based on SGs and SG convolutions, which significantly reduces the inference time and the number of parameters. It also improves the quality of alignment for cross-source data. Furthermore, our method leverages the rotation-equivariant property of SGs and the rotation-invariant SG convolutions to improve the registration procedure.

DeepGMR [51] extracts rotation-invariant features from the point cloud from which a corresponding Gaussian Mixture Model (GMM) is obtained. These GMMs are minimized with the KL-divergence to attain the final transformation. While our work is inspired by DeepGMR’s GMM registration, we use a novel spherical representation of point clouds based on SG Mixture Models, which can be used not only for registration but also retrieval tasks. Moreover, compared to DeepGMR our method does not need externally introduced rotation-invariant features [10] or feature sampling through PointNet.

## 2.2. Shape Retrieval

Shape Retrieval is one of the core elements for the CAD-based reconstruction of indoor environments. The availability of large CAD databases [9, 49] enabled the possibility of using shape retrieval to reconstruct indoor environments, as shown by Scan2CAD [3].

To address the retrieval problem, CORSAIR [54] extends FCGF [18] to learn global object-shape representations and local point-wise features, thus allowing for the retrieval of a similar object from a category-based CAD database. PCR [27] is an unsupervised method based on the unsupervised point cloud registration method of R-PointHop [26] and extends it for the shape retrieval task. By modifying R-PointHop, through aggregating point features into global VLAD representation [25], PCR works as a more generalizable method compared to CORSAIR.

Different from the existing works, our approach results in a one-shot registration and retrieval method, thus reducing complexity. Additionally, this one-shot procedure removes the need to separate retrieval and point cloud registration methods commonly used in scan-to-CAD methods [4, 54]. Furthermore, training time and training complexity are reduced since the SG representation is trained only once for the registration procedure.

## 3. Problem Statement

We consider a scanned indoor environment from which a CAD-based representation is reconstructed. Given a point cloud of the indoor environment and the point clouds of CAD models, there is a need for shape retrieval and point cloud registration.

Let  $\mathcal{X} = \{X^{c,m} | X^{c,m} \in \mathbb{R}^{N_1 \times 3}, c = \{1, \dots, C\}, m = \{1, \dots, M_z\}\}$  be the  $M_z$  segmented objects of different classes  $C$  with  $N_1$  points from an indoor environment  $z$ . In this case, segmentation does not necessarily mean some form of semantic segmentation. It can also be in the form of a 3D bounding volume containing points of the model and the background environment.

In addition,  $\mathcal{Y} = \{Y^{c,k} | Y^{c,k} \in \mathbb{R}^{N_2 \times 3}, c = \{1, \dots, C\}, k = \{1, \dots, K\}\}$  is the database containing  $K$  uniformly sampled CAD models represented as point clouds with  $N_2$  points. The number of points is simplified to  $N = N_1 = N_2$  for a fair comparison to the state of the art. For a retrieval task, we assume that  $K \gg M_z \forall z$  and that all classes represented in  $\mathcal{X}$  are also represented in  $\mathcal{Y}$ .

### 3.1. Point cloud registration

Point cloud registration describes the method of finding a rigid transform of a point cloud  $Y^{c,k}$  to a corresponding point cloud  $X^{c,m}$  of the same class. Depending on the degrees of freedom needed for the point cloud registration, the formulation of the rigid transformation differs slightly. A rigid transform with **6 degrees of freedom** (6DoF) defines the optimization problem

$$\min_{\mathbf{R} \in SO(3), t \in \mathbb{R}^3} d(X^{c,m}, \mathbf{R}Y^{c,k} + t) \quad (1)$$

in which a rotation  $\mathbf{R} \in SO(3)$  and a translation  $t \in \mathbb{R}^3$  with respect to  $Y^{c,k}$  is estimated. For **9 degrees of freedom** this formulation is extended by a diagonal scaling matrix  $\mathbf{S} \in \text{diag}(\mathbb{R}_{\geq 0}^3)$

$$\min_{\mathbf{R} \in SO(3), t \in \mathbb{R}^3, \mathbf{S} \in \text{diag}(\mathbb{R}_{\geq 0}^3)} d(X^{c,m}, \mathbf{R}\mathbf{S}Y^{c,k} + t) \quad (2)$$

where  $d$  represents the distance metric, which depends on the specific use case. If  $X^{c,m}$  and  $Y^{c,k}$  contain point-to-point correspondences created by the same sensing or sampling technique,  $d$  is defined as the average distance between the points  $y_j, x_j \in \mathbb{R}^3$  as in Eq. 3.

$$d(X, Y) = \frac{1}{N} \sum_{j=1}^N \|y_j - x_j\|_2^2 \quad (3)$$

For  $X^{c,m}$  and  $Y^{c,k}$  with no point-to-point correspondence, Barrow *et al.* [6] defined the chamfer distance  $d_C$

$$d_C = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \min_{y \in \mathcal{Y}} \|x - y\|_2^2 + \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} \|x - y\|_2^2. \quad (4)$$

### 3.2. Shape Retrieval

For the retrieval of the model  $Y^{c,k}$  from the database, the optimization can be defined as

$$\arg \min_{k=\{1,\dots,K\}} d_S(Y^{c,k}, X^{c,m}) \quad (5)$$

which represents the minimization of the similarity distance  $d_S$  between one scanned source object model  $X^{c,m}$  and a target model  $Y^{c,k}$  from a database depending on a class  $c \in C$  and a model id  $k \in K$ .

The representation through SGPCR allows us to either use the chamfer distance or the mean square error, depending on the task.

### 4. Preliminaries - Spherical Gaussian Mixture Models

SGPCR core representation is through SGs, which follow the definition of Wang *et al.* [46]:

$$G(\boldsymbol{\nu}; \mathbf{a}, \lambda, \mathbf{p}) = \mathbf{a}e^{\lambda(\boldsymbol{\nu}^T \mathbf{p} - 1)} \quad (6)$$

with the lobe axis  $\mathbf{p} \in \mathbb{R}^3$ , lobe sharpness  $\lambda \in (0, +\infty)$ , the lobe amplitude  $\mathbf{a} \in \mathbb{R}^3$ , and the spherical direction parameter  $\boldsymbol{\nu} \in \mathcal{S}^3$ . The convenient description of spherical radial basis functions allows for the simple extension of Eq. 6 for the use on volumetric data [48].

As [42] and [46] show, SGs are by definition symmetric around the lobe axis  $\mathbf{p}$ . Hence, if a point cloud is represented by SGs and is rotated, this is represented by a rotation of the parameter  $\mathbf{p}$ . This results in a rotation-equivariant description of a model through SGs. This is an essential property for the simplification of the point cloud registration process.

#### 4.1. Spherical Gaussian Mixtures

A Spherical Gaussian Mixture is generated by the summation of various SGs.

$$SGM(\boldsymbol{\nu}, \mathbf{a}, \lambda, \mathbf{p}) = \sum_{i=1}^{|\boldsymbol{\nu}|} G(\boldsymbol{\nu}; \mathbf{a}_i, \lambda_i, \mathbf{p}_i) \quad (7)$$

Eq. 7 represents the parametric description of point clouds by the three parameters  $\mathbf{a}_i$ ,  $\lambda_i$  and  $\mathbf{p}_i$  times the number of SGs, which is the number of spherical samples  $|\boldsymbol{\nu}|$ .

In addition, as defined by [46], the SG Mixture in Eq. 7 has a closed form under rotation. This important property enables us to simplify the estimation of a rotation. For example, if we assume two SG representations of a point cloud, one arbitrarily rotated by  $\mathbf{R}_p \in SO(3)$  and the other not, our problem reduces to finding a rotation  $\mathbf{R}_v \in SO(3)$  on the surface of a sphere for which  $\mathbf{R}_v^T \mathbf{R}_p$  results in the identity matrix  $\mathbf{I}$ . Eq. 8 shows that by finding  $\mathbf{R}_v^T \mathbf{R}_p = \mathbf{I}$ , the non-rotated SG representation can be attained.

$$G(\mathbf{R}_v \boldsymbol{\nu}, \mathbf{R}_p \mathbf{p}, :) = \mathbf{a}e^{\lambda(\boldsymbol{\nu}^T \mathbf{R}_v^T \mathbf{R}_p \mathbf{p} - 1)} = \mathbf{a}e^{\lambda(\boldsymbol{\nu}^T \mathbf{p} - 1)} \quad (8)$$

### 4.2. Spherical Gaussian Convolution

We are also interested in extending the parameter space and improving the quality of the SG representation without changing the rotation-equivariant property of the SGs. Following the derivation of [42], and the simplifications introduced in [46], the rotation-invariant convolution of two SGs ( $G_s(\boldsymbol{\nu}; \mathbf{a}_s, \lambda_s, \mathbf{p}_s)$  and  $G_t(\boldsymbol{\nu}; \mathbf{a}_t, \lambda_t, \mathbf{p}_t)$ ) is defined by

$$(G_s * G_t)(\boldsymbol{\nu}) = \frac{4\pi \mathbf{a}_s \mathbf{a}_t \sinh(d_{st})}{e^{\lambda_s + \lambda_t} d_{st}} \quad (9)$$

with  $d_{st} = \|\lambda_s \mathbf{p}_s + \lambda_t \mathbf{p}_t\|$ , thus allowing for an efficient calculation and representation through SG Mixtures.

The introduction of these two properties, rotation-equivariant representation and rotation-invariant convolution, enables the easy integration of SG-based structures into any deep learning-based system.

### 5. SGPCR

This section describes our approach for representing 3D point clouds through SG mixtures (see also pipeline in Figure 3).

#### 5.1. Spherical Sampling

An SG is the reformulation of a Gaussian function on a 3D surface. A Gaussian function contains the distance between a point and the center of the Gaussian. To represent this on a sphere instead of using the cartesian distance, we have to use the angle between a point sampled on a sphere  $\boldsymbol{\nu} \in \mathcal{S}^3$  and a point on our SG. In Eq. 6, this is shown by the dot product between  $\boldsymbol{\nu}$  and  $\mathbf{p}$ . As such, to represent the point cloud with Eq. 6 spherical sampling is required.

As [41] and [17] mention, there does not exist an optimal discretization on a sphere. Nevertheless, we assume a sub-optimal sampling on a sphere but reduce the error through the use of nearest neighbor clustering.

To sample the sphere, we use the method presented by Vogel [45]. The points sampled by Vogel in cartesian coordinates are converted into spherical coordinates [2] to represent a sampled sphere. Following the notation of [2], a golden angle  $\delta$  is defined, which is derived from the golden ratio  $\phi$

$$\delta = 2\pi(1 - \frac{1}{\phi}) = \pi(3 - \sqrt{5}) \quad (10)$$

where  $\phi = \arccos(1 - 2\frac{i}{N})$  is a ratio that defines the number of consecutive terms of a Fibonacci sequence. Through this angle a sample point  $\boldsymbol{\nu}_i$

$$\boldsymbol{\nu}_i = (\cos(i \cdot \delta) \cdot \sin \phi, \sin(i \cdot \delta) \cdot \sin \phi, \cos \phi) \quad (11)$$

can be constructed. The method creates a spiral structure, which according to [2] generates more evenly spread samples compared to related methods.

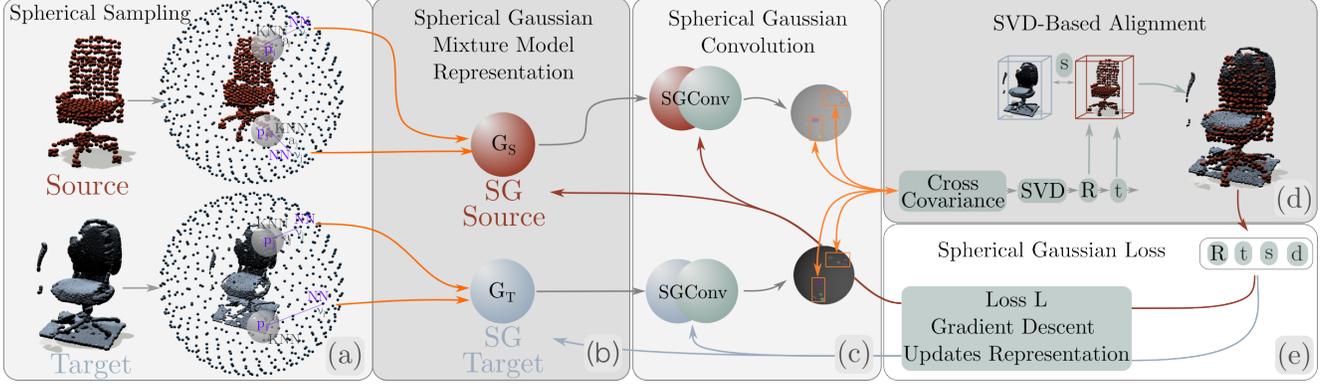


Figure 3: **General pipeline for training the proposed SGPCR-based registration method.** The source and target are used to generate the initial parameters of the Spherical Gaussian Mixture (SG Mixture) shown in a). In b) by combining previously attained a and  $\mathbf{p}$  with trainable  $\lambda$  the SG is generated. After passing the SG Mixtures through c), the SG convolution described in Section 5.3 creates the final trainable SG Mixtures. In d) the representations of target and source are compared, through the cross-covariance. By using SVD, the SGPCR-based registration approach attains the rotation value following [43], the translation value by solving Eq. 14 and the scaling through bounding box comparison as described in Section 5.4. Lastly in e) the resulting transformation and distance metric is used to update the learnable parameters in b) and c) through a gradient descent algorithm. At test time the trained parameters are reused.

## 5.2. Spherical Gaussian Representation

After defining the spherical samples  $\nu$ , this subsection now defines the representation of  $\mathbf{p} \in \mathbb{R}^3$ ,  $\lambda \in \mathbb{R}$  and  $\mathbf{a} \in \mathbb{R}^3$  for point clouds.

For  $\lambda$  a trainable parameter is chosen and initialized between  $[0, +\infty]$ . To obtain the lobe axis  $\mathbf{p}$  the closest point of the sampled sphere to the point cloud is attained by the nearest neighbor search algorithm. After finding this lobe axis  $\mathbf{p}$ , a  $k$ -nearest neighbor search averages the  $k$ -nearest points of the local neighborhood and returns the local feature  $\mathbf{a}$ . Figure 3 shows this in part (b).

## 5.3. Spherical Gaussian Convolution

In addition, a convolutional layer is used in this work to improve the registration and retrieval task by providing more parameters to the optimization algorithm.

In this work, SGConv in Eq. 12, our SG convolution, is defined between the SG representation  $G_X$ , obtained from the point cloud, and our trainable SG kernel  $G_R$ . For registration, we calculate the convolution only once before estimating the rotation matrix.

$$G_X^y = (G_X *_2 G_R)(\nu) = \frac{4\pi \mathbf{a}_X \mathbf{a}_R \sinh(d_{XR})}{e^{\lambda_X + \lambda_R} d_{XR}} \quad (12)$$

Eq. 12, with  $d_{XR} = \|\lambda_X \mathbf{p}_X + \lambda_R \mathbf{p}_R\|$ , shows that the convolution of two SGs is rotation-invariant and requires low computational effort. Thus, we use Eq. 12 to calculate the convolution between the representation of either the source SG  $G_S$  or the target SG  $G_T$  and our SG kernel  $G_R$ . Figure 3 (c) shows this part of the pipeline.

## 5.4. Spherical Gaussian Alignment

First, we want to find the rotation for which the lobes of the source and target SG Mixtures on the sphere have the highest correlation. For this, we calculate the cross-covariance matrix  $\mathbf{W} \in \mathbb{R}^{3 \times 3}$  between  $G_S^{y,i}$  and  $G_T^{y,i}$

$$\mathbf{W} = \begin{bmatrix} G_T^{y,1} & \dots & G_T^{y,|\nu|} \end{bmatrix} \begin{bmatrix} G_S^{y,1} & \dots & G_S^{y,|\nu|} \end{bmatrix}^T. \quad (13)$$

Following the approach of Umeyama *et al.* [43], we can calculate the rotation matrix  $\mathbf{R}_p$  from  $\mathbf{W}$ . Since the SVD is differentiable it can be integrated into the pipeline.

For the translation vector  $t_p \in \mathbb{R}^3$  the lobe centers of  $\mathbf{p}_S$  and  $\mathbf{p}_T$  are compared, thus

$$t_p = \left( \frac{1}{|\nu|} \sum_{i=0}^{|\nu|} \mathbf{p}_T \right) - \mathbf{R}_p \left( \frac{1}{|\nu|} \sum_{i=0}^{|\nu|} \mathbf{p}_S \right). \quad (14)$$

To scale an object, first, the source and target point clouds are transformed through the estimated rotation and translation. Then the oriented bounding box of the target  $O_t \in \mathbb{R}^{8 \times 3}$  is calculated and compared with the oriented bounding box of the source  $O_s \in \mathbb{R}^{8 \times 3}$ . We use Eq. 15 to attain the first part of the scaling estimation  $s_b \in \mathbb{R}^3$ .

$$s_b = \frac{1}{8} \sum_{i=1}^8 \frac{|O_t^i|}{|O_s^i|} \quad (15)$$

$$s_p = \xi s_b \quad (16)$$

Since the bounding boxes of the real-world scanned objects, through occlusions at the time of scanning, do not necessarily cover the whole size of the model, a learnable parameter

$\xi \in \mathbb{R}^3$  is introduced. This results in the final scaling estimation  $s_p \in \mathbb{R}^3$ , seen in Eq. 16.

### 5.5. Spherical Gaussian Loss

Since the registration method is supervised, the estimated  $\mathbf{R}_p$  is directly used in the loss function  $L_R$  shown in Eq. 17 and compared to the ground truth  $\mathbf{R}_g$ . Further, the mean isotropic error (MIE) of [50] is added. While there is no significant reduction in rotation error through MIE, it reduces the time of convergence.

$$L_R = \|\mathbf{R}_g^T \mathbf{R}_p - \mathbf{I}_{3 \times 3}\|_2^2 + \text{MIE}(\mathbf{R}_g, \mathbf{R}_p) \quad (17)$$

The translation loss  $L_T$  is calculated using the mean square error between the ground truth and the prediction.

For the scaling loss  $L_S$ , the L1 loss between the predicted scaling  $s_p$  and the ground-truth scale is calculated.

Lastly, the chamfer distance is used as the distance loss  $L_d$  to gain further information on the distance between the point clouds.

Finally, the loss of the 9DoF registration pipeline can be summarized to

$$L = \alpha L_d + \beta L_T + \gamma L_R + \delta L_S \quad (18)$$

with  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  used as optional weighting factors to allow for trade-off optimization.

The loss  $L$  is used by the gradient-descent algorithm to update the SG kernel  $G_R$ , the initial  $\lambda$  of the SG representation and the scaling variable  $\xi$ . The parameters of  $G_R$  are initialized as random and both  $\lambda$  and  $\xi$  are set to 0.5 in the initialization phase.

## 6. Evaluation

**Datasets.** We evaluate SGPCR-based CAD to scan alignment on two synthetic datasets **ModelNet40** [49] and **ShapeNet** [9] and one real-world **Scan2CAD** [3] dataset.

On synthetic data, the evaluation of registration and retrieval will focus on point cloud data containing zero-intersection noise as defined in [31]. The zero-intersection noise approach simulates a cross-source environment, which is the most common occurrence for the scan-to-CAD approaches as the CAD object can not easily be sampled with the same density distribution of an RGB-D indoor scan such as ScanNet [15]. Thus, this allows for a more realistic comparison to the state-of-the-art on synthetic data.

For retrieval tasks on synthetic data, we either follow [27] on ModelNet40 or [54] on ShapeNet and sample 2048 points from the surface of each object. For registration tasks on synthetic data, we evaluate our work on ModelNet40 following [31], thus uniformly sampling the surface of the objects to generate 1024 points.

As Scan2CAD was one of the first works to use data from ShapeNet to replace objects from a real-world scan taken

from ScanNet, we use it to evaluate our method by following the evaluation approaches of [3] and [54].

**Metrics.** For registration, we use root rotation mean square error RRMSE, translational root mean square error RMSE(t), chamfer distance  $d_C$  and inference time  $\bar{R}$ . Further, we evaluate the number of parameters #Params in a small study.

For retrieval tasks we evaluate the Precision@10 [27], Percision@M=0.1n [54] and the chamfer distance of the Top-1 retrieved model. Precision@10 shows that the correct object is among the top 10 retrieved models, while Percision@M=0.1n, shows that the correct object is among the top 0.1n, with  $n$  the size of the test dataset.

Finally, for scan-to-CAD approaches, we use the popular Scan2CAD benchmark [3]. Here the average of all models that fulfill  $\mathbb{1}[R_{err} \leq 20^\circ \wedge t_{err} < 20cm \wedge s_{err} < 20\%]$  is evaluated. Further, we follow [54] and evaluate using  $R_{err}$  and the one-sided chamfer distance  $d_{SC}$ .

**Training.** For optimization, we use ADAM [28] with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and a learning rate of  $10^{-3}$ . The representation is trained for 300 epochs and contains a step scheduler reducing the learning rate by  $10^{-1}$  every 60 iterations. At test time, the trained parameters are reused and made non-trainable.

### 6.1. ModelNet40

To evaluate ModelNet40 for registration tasks, we follow the approach of DeepUME [31] and compare SGPCR-based registration to other state-of-the-art methods for zero-intersecting noise. Following DeepUME, we remove 1024 points from point cloud  $X^{c,m}$  and then remove the remaining points of  $X^{c,m}$  from the corresponding  $Y^{c,k}$ .

In Table 1 we show the effect of different number of samples on the final alignment. Even with a low amount of samples (or SGs), the RRMSE stays stable. The number of samples also correlates with the number of parameters. For reference, DeepUME [31], which uses a Transformers-based backbone, uses at least 310720 parameters.

Table 2 extends this study and shows the stabilization effect of the number of nearest neighbors (#k) in the k-nearest neighbors algorithm on the amplitude parameter  $\mathbf{a}$ . Here we further notice that increasing #k too much achieves an inverse effect and reduces alignment quality.

Table 3 shows the comparison to other state-of-the-art methods. In addition, we show SGPCR results using different spherical sampling strategies: i) the introduced Vogel sampling, ii) equirectangular distant sampling (ED), and iii) random sampling. We also introduce a PointNet-based (PN-Base) baseline that replaces SG representation and SG-Conv with a PointNet backbone. We use a attained by the spherical sampling as input for PointNet. Comparison with PN-Base shows that SG representation of point clouds compared to PointNet leads to a higher alignment quality. Ta-

| #SG  | $d_C$         | RRMSE       | RMSE(t)       | $\bar{R}(ms)$ | #Params |
|------|---------------|-------------|---------------|---------------|---------|
| 32   | 0.0014        | 10.50       | 0.0077        | <b>0.0016</b> | 515     |
| 64   | 0.0011        | 9.47        | 0.0053        | <b>0.0016</b> | 1027    |
| 128  | <b>0.0010</b> | 8.96        | 0.0042        | <b>0.0016</b> | 2051    |
| 256  | <b>0.0010</b> | 8.60        | 0.0036        | 0.0017        | 4099    |
| 512  | <b>0.0010</b> | 8.57        | <b>0.0034</b> | 0.0017        | 8195    |
| 1024 | <b>0.0010</b> | <b>8.48</b> | <b>0.0034</b> | 0.0022        | 16387   |

Table 1: **Study on different sample densities (#SG).** The alignment of zero-intersecting ModelNet40 data shows that even with a low amount of parameters SGPCR attains stable results in the alignment. The number of parameters is shown for a batch size of 1 and 1024 number of points.

| #k  | $d_C^{-4}$   | RRMSE       | RMSE(t)       | $\bar{R}(ms)$  |
|-----|--------------|-------------|---------------|----------------|
| 1   | 10.19        | 8.61        | <b>0.0034</b> | 0.00173        |
| 2   | 10.14        | 8.58        | <b>0.0034</b> | <b>0.00170</b> |
| 8   | <b>10.04</b> | <b>8.57</b> | <b>0.0034</b> | 0.00177        |
| 32  | <b>10.04</b> | 9.01        | 0.00351       | 0.00250        |
| 128 | 10.52        | 11.08       | 0.00362       | 0.00600        |
| 512 | 16.69        | 37.02       | 0.00553       | 0.02035        |

Table 2: **Study on the number of nearest neighbors (#k).** This study shows that the choice of #k for #SG=512, while not being major, still has an important impact on the final solution.

ble 3 shows that our SGPCR-based approach outperforms comparable methods in terms RRMSE, RMSE(t) and  $d_C$ . For  $\bar{R}$ , SGPCR achieves similar results to comparable methods. Figure 4 shows qualitative comparison with DeepGMR [51] and DeepUME. As noise-free point clouds with existing point-to-point correspondences are highly implausible for the intended use case, they are not considered in our evaluation.

Finally, SGPCR-based retrieval is evaluated on ModelNet40 as in PCRP [27]. Table 4 shows the improvements given by our method for pre-aligned and arbitrary poses. While the one-shot retrieval and registration yielded fruitful results, the evaluation of pre-aligned objects does not simulate real-world problems. For the arbitrary poses, improvements are visible but again the Top-1 metric shows that there still exists a lot of possibility for growth.

## 6.2. ShapeNet

Following the evaluation of CORSAIR [54], we show the results of the retrieval task on the ShapeNet dataset. Table 5 shows the improvement of SGPCR on ShapeNet retrieval compared to comparable methods on the Precision@M=0.1n metric [54]. Since SGPCR not only retrieves objects but also aligns the retrieved object the Top-1  $d_C$  results show lower values compared to the competitors.

| Model          | $d_C$         | RRMSE       | RMSE(t)       | $\bar{R}(ms)$ |
|----------------|---------------|-------------|---------------|---------------|
| PointNetLK [1] | 0.028         | 80.858      | 1.023         | -             |
| DCP [47]       | 0.059         | 93.221      | 0.014         | -             |
| RGM [19]       | 0.254         | 100.97      | 0.388         | -             |
| DeepGMR [51]   | 0.026         | 67.282      | 0.010         | 0.0025        |
| DeepUME [31]   | <b>0.011</b>  | 70.818      | 0.009         | 0.0375        |
| PN-Base        | 0.0258        | 46.23       | 0.0068        | 0.0521        |
| R + SGPCR      | 0.0088        | 34.10       | 0.0129        | <b>0.0017</b> |
| ED + SGPCR     | <b>0.0010</b> | 8.75        | 0.0035        | <b>0.0017</b> |
| Vogel + SGPCR  | <b>0.0010</b> | <b>8.57</b> | <b>0.0034</b> | <b>0.0017</b> |

Table 3: **Registration results on ModelNet40 with zero-intersection noise** trained on data with no sampling noise, thus following the same training/testing method as [31]. Our SGPCR-based registration approach results in significantly lower RMSE(R) and RMSE(t) values. This shows that SGPCR-based point cloud representation generalizes better for zero-intersecting models. Here and for all further results, we used 512 #SGs and #k=8.

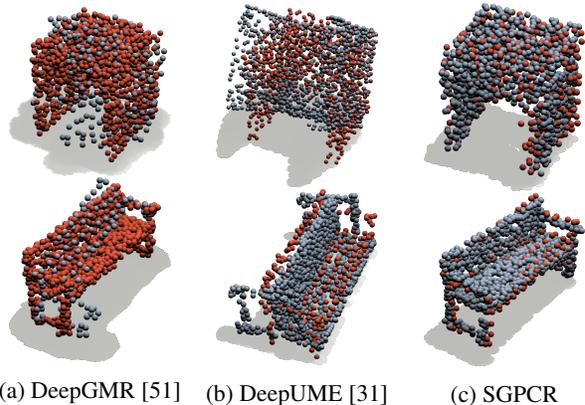


Figure 4: **Qualitative comparison between non-intersecting data on ModelNet40.** This shows qualitatively the results of Table 2. Our registration results in a more stable alignment compared to the state of the art.

| Model         | Pre-aligned objects |               | Arbitrary poses |              |
|---------------|---------------------|---------------|-----------------|--------------|
|               | P@10                | Top-1 $d_C$   | P@10            | Top-1 $d_C$  |
| PointHop [53] | 58.23               | 0.129         | 19.71           | 0.211        |
| FPFH [40]     | 53.23               | 0.164         | 52.12           | 0.160        |
| PointNet [38] | 60.66               | 0.121         | 53.40           | 0.145        |
| CORSAIR [54]  | 61.28               | 0.106         | 61.24           | 0.107        |
| PCRP [27]     | 63.23               | 0.101         | 63.07           | 0.111        |
| SGPCR         | <b>100.0</b>        | <b>0.0076</b> | <b>86.41</b>    | <b>0.020</b> |

Table 4: **Retrieval evaluation on ModelNet40.** SGPCR-based retrieval on the ModelNet40 following the results of [27] outperforms competitors by a great margin.

| Model        | Chair        |              | Table        |              |
|--------------|--------------|--------------|--------------|--------------|
|              | P@M          | Top-1 $d_C$  | P@M          | Top-1 $d_C$  |
| FCGF [13]    | 31.83        | 0.132        | 36.19        | 0.135        |
| CORSAIR [54] | 51.47        | 0.115        | 57.77        | 0.112        |
| SGPCR        | <b>76.20</b> | <b>0.014</b> | <b>75.15</b> | <b>0.016</b> |

Table 5: **Retrieval evaluation on ShapeNet.** Following the results and approach of [54], the improvements through SGPCR-based retrieval are shown.

| Model         | Chair        | Sofa         | Table        |
|---------------|--------------|--------------|--------------|
| Baseline [21] | 42.02        | 27.70        | 18.52        |
| Scan2CAD [3]  | 44.26        | 30.66        | 30.11        |
| E2E [4]       | 73.04        | 76.92        | 48.15        |
| MCSS [21]     | 74.32        | 78.70        | 24.28        |
| SGPCR         | <b>77.39</b> | <b>86.69</b> | <b>55.69</b> |

Table 6: **Evaluation on Scan2CAD benchmark.** Evaluation of object alignment on the Scan2CAD benchmark [3]. Similar to [21] we do not compare ourselves to SceneCad [5], as it uses relationships between objects and layouts, which we do not have access to.

### 6.3. Scan2CAD

Additionally, SGPCR is evaluated on the Scan2CAD [3] dataset. For comparison, we take the results from MCSS [21], which also contains a baseline. For 3D object detection, we use the results of VoteNet [37] followed by SGPCR-based retrieval and registration, which structurally matches the baseline of [21]. Table 6 shows that a considerable improvement can be observed by using SGPCR while evaluating on the Scan2CAD benchmark. MCSS [21] uses an optimization algorithm in the proposal part to improve the scan-to-CAD procedure. While MCSS improves the scan-to-CAD method, high inference times (approximately 15 min) make it not real-time capable. Further, it is to consider that E2E [4] uses a symmetry-aware method which can also be used to extend SGPCR and further improve the results on the Scan2CAD benchmark. The results of Table 6 show that SGPCR outperforms both E2E and MCSS on all three classes. Figure 5 shows a qualitative evaluation.

Finally, Table 7 shows the results of the one-shot retrieval and registration procedure of SGPCR compared to CORSAIR [54]. For the given results even though SGPCR does not consider point cloud symmetry it outperforms CORSAIR on  $d_{SC}$  and  $R_{err} < 45^\circ$ .

## 7. Conclusion

We propose SGPCR, a novel rotation-equivariant approach for representing point clouds by using Spherical

| Model                | C     | $d_{SC}$      | $R_{err} < 45^\circ$ |
|----------------------|-------|---------------|----------------------|
| CORSAIR [54] w/o sym | Chair | 0.0753        | 86.4                 |
| CORSAIR [54] w/ sym  |       | 0.0681        | 88.9                 |
| SGPCR                |       | <b>0.0054</b> | <b>90.89</b>         |
| CORSAIR [54] w/o sym | Table | 0.0906        | 52.9                 |
| CORSAIR [54] w/ sym  |       | 0.0714        | 57.0                 |
| SGPCR                |       | <b>0.0150</b> | <b>76.59</b>         |

Table 7: **Retrieval and registration on the Scan2CAD dataset** following the training and evaluation of [54].

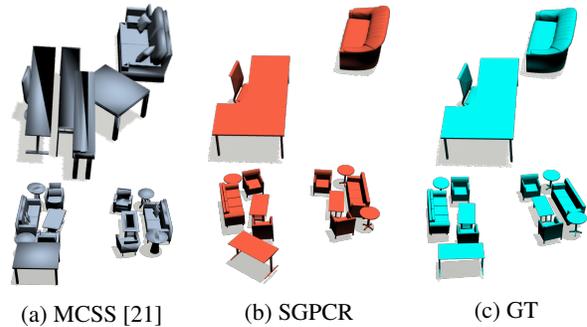


Figure 5: **Qualitative comparison on the Scan2CAD dataset.** MCSS [21] is able to attain more CAD objects, even though those are not in the ground truth dataset. Still through Table 6 and the given qualitative results SGPCR shows overall improvements.

Gaussians. Using the proposed rotation-equivariant representation and the trainable rotation-invariant convolution, we can create a more stable representation of point clouds.

Our experiments show that this form of representation improves alignment and retrieval between point clouds with zero point-to-point correspondences, especially in the context of scan-to-CAD applications. While our evaluation significantly improves alignment on synthetic data, improvements on scan-to-CAD tasks still suffer due to the multi-stage approach required to achieve these results.

Since SGPCR has been shown to significantly improve alignment quality with fewer parameters, we hope that it will serve as a replacement for PointNet or Transformers in cross-source point cloud registration applications.

## Acknowledgements

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the programme of ‘‘Souverän. Digital. Vernetzt.’’. Joint project 6G-life, project identification number: 16KISK002

## References

- [1] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7156–7165, 2019.
- [2] M. K. Arthur. Point picking and distributing on the disc and sphere. 2015.
- [3] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X. Chang, and Matthias Nießner. Scan2cad: Learning cad model alignment in rgb-d scans. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2609–2618, 2019.
- [4] Armen Avetisyan, Angela Dai, and Matthias Nießner. End-to-end cad model retrieval and 9dof alignment in 3d scans. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2551–2560, 2019.
- [5] Armen Avetisyan, Tatiana Khanova, Christopher Bongsoo Choy, Denver Dash, Angela Dai, and Matthias Nießner. Scenecad: Predicting object alignments and layouts in rgb-d scans. *ArXiv*, abs/2003.12622, 2020.
- [6] Harry G. Barrow, Jay M. Tenenbaum, Robert C. Bolles, and Helen C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *IJ-CAI*, 1977.
- [7] Paul J. Besl and Neil D. McKay. Method for registration of 3-d shapes. In *Other Conferences*, 1992.
- [8] Tim Beyer and Angela Dai. Weakly-supervised end-to-end cad retrieval to scan objects. *ArXiv*, abs/2203.12873, 2022.
- [9] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.
- [10] Chao Chen, Guanbin Li, Ruijia Xu, Tianshui Chen, Mei Wang, and Liang Lin. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4989–4997, 2019.
- [11] Yang Chen and Gérard G. Medioni. Object modeling by registration of multiple range images. *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 2724–2729 vol.3, 1991.
- [12] Sungjoon Choi, Qian-Yi Zhou, and Vladlen Koltun. Robust reconstruction of indoor scenes. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5556–5565, 2015.
- [13] Christopher Bongsoo Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8957–8965, 2019.
- [14] Manuel Dahnert, Angela Dai, Leonidas J. Guibas, and Matthias Nießner. Joint embedding of 3d scan and cad objects. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8748–8757, 2019.
- [15] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2443, 2017.
- [16] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: real-time globally consistent 3d reconstruction using on-the-fly surface re-integration. *ArXiv*, abs/1604.01093, 2017.
- [17] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so(3) equivariant representations with spherical cnns. In *ECCV*, 2018.
- [18] Qiaojun Feng and Nikolay A. Atanasov. Fully convolutional geometric features for category-level object alignment. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8492–8498, 2020.
- [19] Kexue Fu, Shaolei Liu, Xiaoyuan Luo, and Manning Wang. Robust point cloud registration framework based on deep graph matching. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8889–8898, 2021.
- [20] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:4338–4364, 2021.
- [21] Shreyas Hampali, Sinisa Stekovic, Sayan Deb Sarkar, C. Srinivasa Kumar, Friedrich Fraundorfer, and Vincent Lepetit. Monte carlo scene search for 3d scene understanding. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13799–13808, 2021.
- [22] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4416–4425, 2019.
- [23] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard A. Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew J. Davison, and Andrew W. Fitzgibbon. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011.
- [24] Hamid Izadinia, Qi Shan, and Steven M. Seitz. Im2cad. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2422–2431, 2017.
- [25] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3304–3311, 2010.
- [26] Pranav Kadam, Min Zhang, Shan Liu, and C.-C. Jay Kuo. R-pointhop: A green, accurate, and unsupervised point cloud registration method. *IEEE Transactions on Image Processing*, 31:2710–2725, 2022.
- [27] Pranav Kadam, Qingyang Zhou, Shan Liu, and Cs Kuo. Pcrp: Unsupervised point cloud object retrieval and pose estimation. *ArXiv*, abs/2202.07843, 2022.

- [28] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.
- [29] Weicheng Kuo, Anelia Angelova, Tsung-Yi Lin, and Angela Dai. Mask2cad: 3d shape prediction by learning to segment and retrieve. In *ECCV*, 2020.
- [30] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12689–12697, 2019.
- [31] Natalie Lang and Joseph M. Francos. Deepume: Learning the universal manifold embedding for robust point cloud registration. *ArXiv*, abs/2112.09938, 2021.
- [32] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2929–2938, 2021.
- [33] Yecheng Lyu, Xinming Huang, and Ziming Zhang. Ellipsoidnet: Ellipsoid representation for point cloud classification and segmentation. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 256–266, 2022.
- [34] Amar Man Maharjan and Xiaohui Yuan. Registration of human point set using automatic key point detection and region-aware features. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2264–2272, 2022.
- [35] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew W. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011.
- [36] C. Qi, Xinlei Chen, Or Litany, and Leonidas J. Guibas. Imvotenet: Boosting 3d object detection in point clouds with image votes. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4403–4412, 2020.
- [37] C. Qi, Or Litany, Kaiming He, and Leonidas J. Guibas. Deep hough voting for 3d object detection in point clouds. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9276–9285, 2019.
- [38] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.
- [39] Danila D. Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1265–1274, 2022.
- [40] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009.
- [41] William P. Thurston and Silvio V. F. Levy. Three-dimensional geometry and topology. 1997.
- [42] Yu-Ting Tsai and Zen-Chung Shih. All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. *ACM Trans. Graph.*, 25:967–976, 2006.
- [43] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13:376–380, 1991.
- [44] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017.
- [45] Helmut Vogel. A better way to construct the sunflower head. *Bellman Prize in Mathematical Biosciences*, 44:179–189, 1979.
- [46] Jiaping Wang, Peiran Ren, Minmin Gong, John M. Snyder, and Baining Guo. All-frequency rendering of dynamic, spatially-varying reflectance. *ACM SIGGRAPH Asia 2009 papers*, 2009.
- [47] Yue Wang and Justin M. Solomon. Deep closest point: Learning representations for point cloud registration. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3522–3531, 2019.
- [48] Zian Wang, Jonah Philion, Sanja Fidler, and Jan Kautz. Learning indoor inverse rendering with 3d spatially-varying lighting. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12518–12527, 2021.
- [49] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015.
- [50] Zi Jian Yew and Gim Hee Lee. Rpm-net: Robust point matching using learned features. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11821–11830, 2020.
- [51] Wentao Yuan, Benjamin Eckart, Kihwan Kim, V. Jampani, Dieter Fox, and Jan Kautz. Deepgmr: Learning latent gaussian mixture models for registration. In *ECCV*, 2020.
- [52] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5449–5458, 2021.
- [53] 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*, 22:1744–1755, 2020.
- [54] Tianyu Zhao, Qiaoqun Feng, Sai Jadhav, and Nikolay A. Atanasov. Corsair: Convolutional object retrieval and symmetry-aided registration. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 47–54, 2021.
- [55] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.