UMERegRobust - Universal Manifold Embedding Compatible Features for Robust Point Cloud Registration

Yuval Haitman*, Amit Efraim, and Joseph M. Francos

Ben-Gurion University, Beer-Sheva, Israel

Abstract. In this paper, we adopt the Universal Manifold Embedding (UME) framework for the estimation of rigid transformations and extend it, so that it can accommodate scenarios involving partial overlap and differently sampled point clouds. UME is a methodology designed for mapping observations of the same object, related by rigid transformations, into a single low-dimensional linear subspace. This process yields a transformation-invariant representation of the observations, with its matrix form representation being covariant (*i.e.* equivariant) with the transformation. We extend the UME framework by introducing a UME-compatible feature extraction method augmented with a unique UME contrastive loss and a sampling equalizer. These components are integrated into a comprehensive and robust registration pipeline, named *UMERegRobust*. We propose the RotKITTI registration benchmark, specifically tailored to evaluate registration methods for scenarios involving large rotations. UMERegRobust achieves better than state-of-the-art performance on the KITTI benchmark, especially when strict precision of $(1^\circ, 10cm)$ is considered (with an average gain of +9%), and notably outperform SOTA methods on the RotKITTI benchmark (with +45% gain compared the most recent SOTA method). Our code is available at https://github.com/yuvalH9/UMERegRobust.

Keywords: Point clouds · Registration · Rigid transformation estimation · Invariant Representations · Equivariant Representations

1 Introduction

Point cloud registration is a critical component in many vision-based applications, such as perception for autonomous systems. The registration of point cloud observations on a rigid object, or scene, amounts to estimating the rigid transformation relating them. However, in practical scenarios, these observations are often characterized by partial overlap as a result of being acquired from different viewpoints, as well as by different sampling patterns.

The conventional approach to addressing point cloud registration relies on three basic components: feature extraction, keypoint matching, and the estimation of rigid transformations based on corresponding keypoints.

* Corresponding Author

Recent studies have shifted towards employing learning-based feature extraction, demonstrating promising results even in scenarios with significant variations among observations [6, 17, 22, 24, 30]. Despite the notable advancements facilitated by learning-based techniques, the core elements of the registration pipeline have remained largely unchanged, relying heavily on traditional methods such as Horn [15] for estimating rigid transformations.

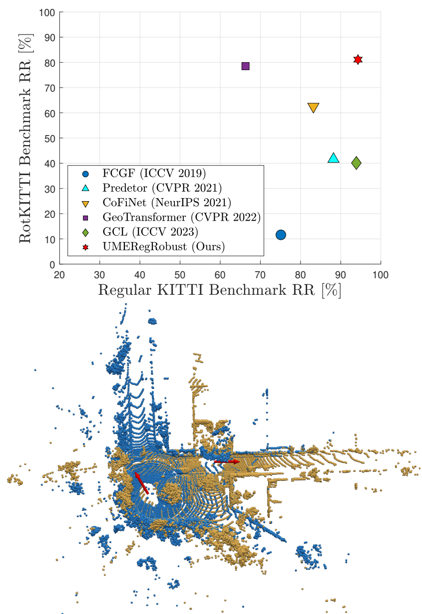

Fig. 1: Top: Registration Recall (RR) performance of different baselines on regular KITTI (x-axis) and RotKITTI (y-axis) registration benchmarks. UMERegRobust outperforms the compared SOTA methods, on both benchmarks. **Bottom**: Registration problem from RotKITTI benchmark, highlighting significant rotation between measurements. Source and target point clouds are shown in different colors, with arrow direction representing vehicle heading, indicating a 120° rotation problem.

In this study, we adopt the Universal Manifold Embedding (UME) framework [10, 14] for the estimation of rigid transformations and extend it, so that it can accommodate scenarios involving large transformations, partial overlap and differently sampled point clouds. UME is a methodology designed for mapping observations (*e.g.*, images, 3D point clouds, etc.) of the same object, related by rigid transformations, into a single low-dimensional linear subspace. This process yields a transformation-invariant representation of the observations, with its matrix form representation being covariant (*i.e.*, equivariant) with the transformation. This duality is advantageous, as the invariant representation facilitates matching of corresponding observations, while the covariant property of the matrix representation enables estimating the transformation relating the observations. A prerequisite for generating a UME descriptor for a given observation is to define a transformation-invariant function over the observations, which we name the observation coloring function.

Unlike registration methods that rely solely on point-wise matched point's coordinates information for transformation estimation [15], the UME leverages both local geometric correspondences and their matched neighborhood coloring. This approach leads to a closed-form solution that offers greater accuracy and robustness.

While the UME registration method has demonstrated effectiveness on synthetic closed objects like [5, 29], it encountered difficulties with outdoor and indoor scans due to lack of invariant coloring functions, robust to sampling variations and partial overlap. In this study, we propose a novel UME-compatible coloring solution that is both invariant to transformations and robust to sampling variations. We introduce a coloring module based on a Fully-Convolutional neural network, which we train using a novel UME contrastive learning approach on pairs of point cloud observations, pre-processed by a Sampling-Equalizer Module aimed at enhancing robustness to sampling variations. Finally, we introduce a comprehensive registration pipeline built upon the UME framework.

We showcase the performance of our proposed method on various registration benchmarks including outdoor(KITTI [13], nuScenes [3]) and indoor (3DMatch [33]). For many perception tasks, especially in autonomous driving, high precision is crucial for system safety and performance. Hence, we evaluate and compare our method on the outdoor benchmarks with a strict precision criteria of $(1^{\circ}, 10cm)$. We also suggest RotKITTI - a new outdoor registration benchmark focusing on problems with large rotations. This type of problems is of high importance in evaluating registration methods for SLAM systems loop closure [8]. We compare the performance of the proposed method against a large set of baseline methods (some of the comparisons are shown in Fig. 1).

The main contributions of this paper are:

1. We introduce a novel Universal Manifold Embedding (UME) compatible coloring method, augmented with a unique UME contrastive loss and a Sampling Equalizer Module. The proposed UME-compatible coloring provides an **enabler** that facilitates high performance UME-based registration for general scenes such as outdoor/indoor scans. Additionally, we present a comprehensive and robust RANSAC-Free registration pipeline for 3D point clouds, comprising a dedicated feature extractor for UME descriptor generation, a matched manifold detector for point cloud matching, a UME-based estimator for hypothesis estimation, and a hypothesis selection module for selecting the best estimator.
2. We propose the RotKITTI and RotnuScenes registration benchmarks, specifically tailored to evaluate registration methods for scenarios involving large rotations. These benchmarks are crucial for assessing methods intended for integration into SLAM systems, particularly for the task of loop closure.
3. We achieve better than state-of-the-art performance on the KITTI benchmark, especially when strict precision of $(1^{\circ}, 10cm)$ is considered (with an average gain of +9%), and notably outperform SOTA methods on the RotKITTI benchmark (with +45% gain compared the most recent SOTA method).

2 Related Work

When the relative pose of point clouds undergoing registration is unknown, the most common approach begins with matching key points, followed by estimating transformations between these correspondences using Horn's solution via constrained least squares estimation [15]. Due to the presence of outliers in the estimated correspondences, robust registration algorithms, such as Random Sample Consensus (RANSAC), are necessary to estimate the registration parameters and achieve an approximate alignment [12]. Following this initial alignment, local optimization is typically performed (e.g., [2, 25]).

Recent advancements in point cloud registration have primarily focused on developing high-performance feature descriptors, while the core registration methodology has remained largely unchanged. Modern 3D feature descriptors incorporate FCGF [6] and KPConv [26] architectures as part of broader solutions for both indoor and outdoor registration challenges. Overlap Predator [17], for example, employs KPConv convolutions with self and cross-overlap attention modules to learn local and co-contextual information. GeoTransformer [24] uses similar backbone with a geometric transformer module in a hierarchical approach to estimate point correspondences. GCL [22] adopts a unique group-wise contrastive learning approach, achieving high performance in scenarios with very low overlap and large translations. CofiNet [31] combines a coarse-to-fine matching approach with attentional feature aggregation to generate point correspondences.

Other methods focus on indoor scenarios only. Lepard [21], for instance, utilizes positional encoding along with a reposition technique to modify cross-point cloud relative positions, applicable for both rigid and non-rigid transformation estimation. PEAL [32] initially employs a GeoTransformer model to estimate the overlap region between two-point clouds and then uses a similar model to estimate the transformation based on this overlap. YOHO [27] and RoReg [28] propose rotation-invariant descriptors by averaging descriptors obtained over multiple rotations, addressing rotation invariance but not translation. E2E [4] suggests a rotation-invariant descriptor using Spherical CNN [7].

A common aspect of these approaches is that registration is achieved by aligning the estimated set of corresponding points, meaning only the coordinates of the matched points are used in the transformation estimation [15]. In contrast, the proposed UME registration pipeline uses both the estimated correspondences and their extracted features, thereby embedding valuable contextual information in the transformation estimation process.

3 Method

Let $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^3$ be two partially overlapping point clouds related by a rigid transformation and sampling variations. We solve the registration problem between $\mathcal{P}$ and $\mathcal{Q}$ by estimating the rigid transformation $(\boldsymbol{R}, \boldsymbol{t})$ where $\boldsymbol{R} \in \mathrm{SO}(3)$ and $\boldsymbol{t} \in \mathbb{R}^3$. The *UMERegRobust* pipeline is depicted in Fig.2. It incorporates three primary components. (1) UME Compatible Feature Extractor which assigns dense features to the input point clouds. It is designed to satisfy the requirements of the UME framework; (2) Key-point matching and hypothesis generation module, that employs the UME descriptors evaluated from the point cloud dense features, to initially establish putative matches through a Matched Manifold Detector (MMD), followed by generation of hypothesized estimates of the rigid transformation using UME-based estimators; (3) Hypothesis selection module, that selects the best transformation estimate through maximization of the point clouds feature correlation.

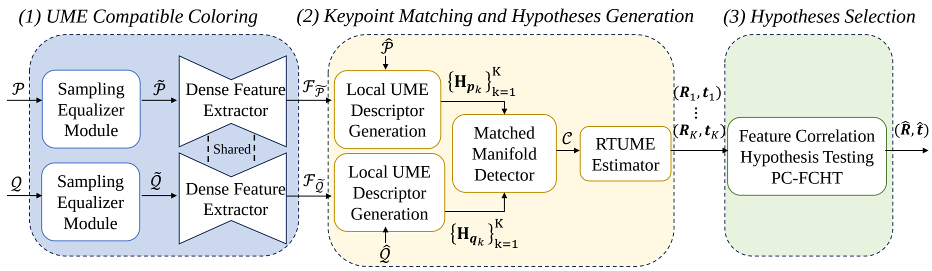

Fig. 2: UMERegRobust Overview. $\mathcal{P}, \mathcal{Q}$ are the input point clouds; (1) The UME Compatible Coloring Module resamples the point clouds into a uniform grid generating $\widetilde{\mathcal{P}}$ and $\widetilde{\mathcal{Q}}$, then assigns each point with a transformation invariant feature vector creating the colored point clouds $\mathcal{F}_{\widetilde{\mathcal{P}}}, \mathcal{F}_{\widetilde{\mathcal{Q}}}$. (2) Local UME descriptors, generated on a down-sampled versions of the point clouds $\widehat{\mathcal{P}}, \widehat{\mathcal{Q}}$, are denoted by $\{\mathbf{H}_{\boldsymbol{p}}\}_{k=1}^{K}, \{\mathbf{H}_{\boldsymbol{q}}\}_{k=1}^{K}$, respectively. A Matched Manifold Detector identifies corresponding local UME descriptors, forming a set of K putative matched pairs $\mathcal{C}$. For each matched pair, an estimated transformation is obtained using the RTUME estimator. (3) Feature correlation is used to select the hypnosis that maximize the feature correlation between the point clouds.

3.1 Universal Manifold Embedding Overview

Let s be a 3D object and $\mathcal{O}_s \subset \mathbb{R}^3$ be the set of all possible observations on s generated by the action of the transformation group SE(3), *i.e.*, $\mathcal{O}_s$ is the *orbit* of s. It has been shown [10] that the Universal Manifold Embedding (UME) provides a mapping from the orbit of possible observations on the object, generated by the action of the transformation group SE(3) to a single low dimensional linear subspace of Euclidean space. This linear subspace is *invariant* to geometric transformation and hence is a unique representative of the object, regardless of its observed pose, while its matrix representation is *covariant* (*i.e.* equivariant) with the transformation. It thus naturally serves as an invariant statistic for solving problems of joint detection and transformation estimation.

Let $o \in \mathcal{O}_s$ be a point cloud observation on the object s and $f : \mathbb{R}^3 \to \mathbb{R}^d$ is a function that assigned a real valued vector to each point in the observation. We name f the *observation coloring function* and $f_o(\mathbf{x}) = \{f(\mathbf{x})|\mathbf{x} \in o\}$ the *colored observation*. The UME matrix of the colored observation $h(\mathbf{x}) = f_o(\mathbf{x})$, which serves as the *UME Descriptor* of the observation is given by:

$$\mathbf{T}(h) = \begin{bmatrix} \int\limits_{\mathbb{R}^3} w_1 \circ h(\mathbf{x}) d\mathbf{x} & \int\limits_{\mathbb{R}^3} x_1 w_1 \circ h(\mathbf{x}) d\mathbf{x} & \dots & \int\limits_{\mathbb{R}^3} x_3 w_1 \circ h(\mathbf{x}) d\mathbf{x} \\ & & \vdots & \\ \int\limits_{\mathbb{R}^3} w_M \circ h(\mathbf{x}) d\mathbf{x} & \int\limits_{\mathbb{R}^3} x_1 w_M \circ h(\mathbf{x}) d\mathbf{x} & \dots & \int\limits_{\mathbb{R}^3} x_3 w_M \circ h(\mathbf{x}) d\mathbf{x} \end{bmatrix} \quad (1)$$

where $\{w_m\}_{m=1}^{M}$ are measurable functions aimed at generating many compandings of the observation and $\mathbf{x} = [x_1, x_2, x_3]^T$. A necessary condition for the applicability of the UME representation for point cloud registration is that the coloring function f be invariant to the transformation $i.e.$,that $f(\mathbf{x}) = f(\boldsymbol{R}\mathbf{x}+\boldsymbol{t})$ where $(\boldsymbol{R}, \boldsymbol{t}) \in \mathrm{SE}(3)$. For any two colored observations $h(\mathbf{x}) = f_o(\mathbf{x})$ and $g(\mathbf{x}) = f_{o'}(\mathbf{y})$ where o, o' are related by rigid transformation, their corresponding UME matrices are related by $\mathbf{T}(h) = \mathbf{T}(g)\mathbf{D}^{-1}(\boldsymbol{R}, \boldsymbol{t}), \quad \mathbf{D}(\boldsymbol{R}, \boldsymbol{t}) = \begin{bmatrix} 1 & \boldsymbol{t}^T \\ \mathbf{0} & \boldsymbol{R}^T \end{bmatrix}$ Therefore, the UME matrix representation of the observation $\mathbf{T}(h)$ is covariant with the transformation, where the linear subspace spanned by its column space $\langle \mathbf{T}(h) \rangle$ is invariant to the transformation (since $\mathbf{D}$ is an inevitable matrix). Note that $\langle \mathbf{T}(h) \rangle \in \mathrm{Gr}(M, 4)$, where $\mathrm{Gr}(M, 4)$ is the Grassmann manifold of 4-dimensional linear subspaces of an M-dimensional Euclidean space.

UME Local Descriptor. Since the UME representation in (1) was originally developed for an observation on a single object, adaptations to the general case where multiple objects are present ($e.g.$, LiDAR scans), are required. Thus, rather than generating a single UME descriptor for the entire observation, multiple local UME descriptors are generated. Given that the entire observation undergoes the same rigid transformation, each local descriptor undergoes the same transformation: Let h be a colored observation of the point cloud $\mathcal{P} \subset \mathbb{R}^3$ (colored by the observation coloring function) and $\{\boldsymbol{p}_k\}_{k=1}^K \subseteq \mathcal{P}$ a set of selected points each with corresponding local neighborhood of radius R, $\mathcal{P}_k = \{\boldsymbol{p}_i \in \mathcal{P} \big| \|\boldsymbol{p}_i - \boldsymbol{p}_k\|_2 \leq R\}$. The *Local UME Descriptor* of $\boldsymbol{p}_k$ denoted by $\mathbf{H}_{\boldsymbol{p}_k}$ is obtained using a local adaption of (1) where the integrals are evaluated locally on the subset $\mathcal{P}_k$, and thus for correctly matched points in two point clouds to be registered, the UME relations holds.

3.2 UME Compatible Features

As outlined in Sec. 3.1, a necessary step in adapting the UME framework for point cloud observations is to define a coloring function that assigns each point in the cloud with a value. This function should be invariant to the action of rigid transformations. For closed objects, natural candidates for the coloring function, such as distance from the center of mass, or surface curvature have been employed [10]. However, these functions are not optimized for scans of outdoor or indoor scenarios. Moreover, since the point clouds to be registered may be acquired by different sensors, at different times and from different points of view, the coloring function has to be robust to different sampling patterns where it is evident that the aforementioned candidates may be sensitive to them.

Since defining such a function analytically can be very hard, if at all possible, we adopt a data-driven approach for implementing the coloring function so that it is compatible with the UME framework such that three primary requirements are satisfied: (1) Invariance to rigid transformations; (2) Robustness to sampling variations; (3) High expressibility of the observation. The UME compatible coloring module, illustrated in Fig.2, comprises two main building blocks. To address sampling differences, we first introduce a Sampling Equalizer

Module, serving as a preprocessing technique for the subsequent dense feature extraction, implemented by a deep neural network.

Sampling Equalizer Module. To mitigate the mismatch in the sampling patterns of a pair of point cloud observations we implemented a two-stage *Sampling Equalizer Module* (SEM). Initially, we employ an off-the-shelf surface reconstruction technique for point cloud observations. Numerous options exist in the literature, ranging from classical methods such as Poisson reconstruction [18] to DNN-based approaches. In this study, we adopted the Neural Kernel Surface Reconstruction (NKSR) [16], a single-shot surface reconstruction technique, chosen for its robust and real-time performance compared to other methods. Next, the reconstructed surface generated from the point cloud observation is sampled into a uniform grid with voxel size of ρ, resulting in a new, uniformly sampled point cloud denoted in Fig. 2 by $\widetilde{\mathcal{P}}, \widetilde{\mathcal{Q}}$. In the Supplementary, examples are provided of point cloud observations - before and after the SEM. The application of SEM contributes to UME compatibility in two main ways. First, we can control the density of the processed point clouds by adjusting the value of ρ, thereby enhancing the precision of numerically evaluating the integrals in (1). Second, by introducing uniformity in sampling the point clouds, we reduce mismatches between local UME descriptors of matching neighborhoods, that may result from evaluating the integrals in (1) over differently sampled neighborhoods.

Dense Feature Extractor. We adopt a fully convolutional neural network as our feature extractor, employing a similar architecture to [6]. Our feature extractor is a Unet-shaped DNN utilizing sparse 3D convolutions with skip-connections and it is the only learnable module in the entire registration pipeline (see detailed implementation in the Supplementary). The feature extractor assigns a feature vector to every point in the input point cloud. These features encapsulate both global and local contextual information based solely on the observation geometry, thus creating a coloring function with high expressibility of the observation. Thus, the coloring function is the result of two cascaded blocks: the SEM and the dense feature extractor. Invariance to rigid transformations is achieved by training the model using augmentations, along with utilizing UME-compatible losses that optimize the coloring module to match the UME theoretical requirements.

3.3 Keypoint Matching and Hypothesis Generation

Matched Manifold Detector. Since the linear subspace spanned by each UME local descriptor is invariant to rigid transformations, UME local descriptors of matching points are mapped into the same point on the Grassmann Manifold $\mathrm{Gr}(M, 4)$, regardless of their pose. Therefore, an effective approach to match these descriptors is by assessing their affinity on the manifold. To select the putative points to be matched, we create a down-sampled version of the input point clouds $\widehat{\mathcal{P}} = \{\boldsymbol{p}_k\}_{k=1}^{K} \subseteq \mathcal{P}$, $\widehat{\mathcal{Q}} = \{\boldsymbol{q}_k\}_{k=1}^{K} \subseteq \mathcal{Q}$. For each selected point we generate its local UME descriptor from its corresponding local neighbourhood

$\{\mathcal{P}_k\}_{k=1}^K$,$\{\mathcal{Q}_k\}_{k=1}^K$ using the assigned features $\{\mathcal{F}_{\mathcal{P}_k}\}_{k=1}^K$, $\{\mathcal{F}_{\mathcal{Q}_k}\}_{k=1}^K$ to obtain $\{\mathbf{H}_{\boldsymbol{p}_k}\}_{k=1}^K$, $\{\mathbf{H}_{\boldsymbol{q}_k}\}_{k=1}^K$, respectively. Let $\langle\mathbf{X}\rangle$ and $\langle\mathbf{Y}\rangle$ be two linear subspaces spanned by the columns of full rank matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{M\times r}$, respectively. Their distance on $\mathrm{Gr}(M, r)$ is [9]:

$$d_{pF}\left(\mathbf{X}, \mathbf{Y}\right) = \frac{1}{\sqrt{2}}\|\mathbf{P}_{\mathbf{X}} - \mathbf{P}_{\mathbf{Y}}\|_F = \|\sin(\boldsymbol{\theta})\|_2 \tag{2}$$

where $\mathbf{P}_{\mathbf{X}}, \mathbf{P}_{\mathbf{Y}}$ are the orthogonal projection matrices on the column space of $\mathbf{X}$ and $\mathbf{Y}$, respectively. The distance is also equivalent to the ℓ_2-norm of the sine vector of the principal angles between the two subspaces where the maximal distance is $\sqrt{r}$, and is zero for identical subspaces. Matched Manifold Detection (MMD) is defined as the operation of detecting an observation that belongs to the same orbit (as defined in Sec. 3.1) as a given query observation, regardless of their relative transformation. Ideally, MMD operation is equivalent to finding a UME descriptor with zero d_{pF} distance to the tested query. However, in practice, local UME descriptors of matching neighborhoods are not identical due to the differences in the sampling of these neighbourhoods. Hence, the zero d_{pF} constraint is relaxed to a minimal distance, and the keypoint matching in the evaluation step is done by employing the Hungarian Algorithm [20] on the pair-wise d_{pF} distance matrix. Note that while the standard point-to-point matching [1,6,17,24,31] may be sensitive to the noisy nature of the measurements, applying neighbourhood-to-neighbourhood matching using a Matched Manifold Detector applied to the UME local descriptors provides higher robustness.

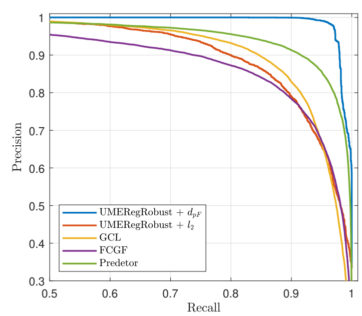

Fig. 3: PR-Curve of Keypoint matching performance under rigid transformation, partial overlap and sampling variations of UME Local descriptor vs. other descriptors.

To demonstrate the robustness of local UME descriptors in keypoint matching under rigid transformations, partial overlap, and varying sampling patterns, we compared their performance to that of STOA local point-wise descriptors. Fig. 3 presents a precision-recall curve comparing local UME descriptors (distances evaluated by d_{pF}) matching accuracy, to other descriptors matching accuracy. The results clearly show the superior accuracy and robustness of local UME descriptors, especially at Recall @0.95, where they achieve a precision gain of approximately 15% over the leading competitor. Additionally, we evaluated the matching performance of Local UME point-wise descriptors using the ℓ_2 metric, which resulted in a significant deterioration of 30% compared to the UME neighborhood-based descriptor and d_{pF} distance. These findings underscore the significant advantage of using UME-based distance between neighborhoods over other local point-wise descriptors.

Hypothesis Generation. Following the detection of matching neighborhoods, the Rigid Transformation UME (RTUME) estimator [10] is employed for generating multiple hypotheses of the underlying transformation relating every pair of matching neighborhoods. More specifically, for every one of K putative matched pairs, the RTUME estimator generates an hypothesized estimate, with total of K hypotheses $\{(\boldsymbol{R}_k, \boldsymbol{t}_k)\}_{k=1}^{K}$. Note that unlike common estimation methods, *e.g.*, [15] that require multiple point correspondences to generate a *single* estimate of the transformation, the RTUME provides a transformation estimate from every single pair of matched neighborhood descriptors. The RTUME estimate of the transformation employs information from the entire neighborhood of the matched point thus leveraging the covariant property of the UME descriptor. Unlike competing methods [17, 24, 31], where only the coordinates of the matched points are used to estimate the transformation, the RTUME employs information on both the coordinate values and the feature values, resulting in higher accuracy and robustness of the estimates.

3.4 Loss Functions

The loss function adopted to train the dense feature extractor, in a supervised fashion, $\mathcal{L} = \lambda_1 \mathcal{L}_{pw} + \lambda_2 \mathcal{L}_{\text{UME}} + \lambda_3 \mathcal{L}_{reg}$ is composed of three losses aiming at complementary goals: a point-wise contrastive loss to increase point-level features invariance, a UME-contrastive loss for optimizing the features towards conforming with the UME framework assumptions, and an auxiliary registration loss to guide the feature extractor towards optimizing its performance in the task of point cloud registration.

Point-wise Contrastive Loss. Metric learning, and specifically, contrastive learning, is a widely used approach for training feature extraction models [6,22]. Despite the differences between the two input point clouds, the point-wise contrastive loss aims to enforce similarity between the features of matching points (positive group) while reducing the similarity between non-matching points (negative group). The employed point-wise loss is an adapted version of the Supervised Contrastive Learning loss (SCL) [19] for optimizing the feature extraction from 3D point clouds. Let $\mathcal{M}_{pw}$ be the set of positive matches, which includes all point pairs that are *at most* ϵ meters apart under the ground-truth transformation. Let $\mathcal{N}_{pw}^{\boldsymbol{p}}$ be the set of negative pairs of $\boldsymbol{p} \in \mathcal{P}$, *i.e.*, they are *at least* R away from each other under the ground-truth transformation. The Point-Wise Contrastive Loss is then given by:

$$\mathcal{L}_{pw} = - \sum_{(\boldsymbol{p},\boldsymbol{q}) \in \mathcal{M}_{pw}} \log \frac{\exp\left(\boldsymbol{f}_{\boldsymbol{p}}^T \boldsymbol{f}_{\boldsymbol{q}} / \tau\right)}{\exp\left(\boldsymbol{f}_{\boldsymbol{p}}^T \boldsymbol{f}_{\boldsymbol{q}} / \tau\right) + \sum_{\boldsymbol{z} \in \mathcal{N}_{pw}^{\boldsymbol{p}}} \exp\left(\boldsymbol{f}_{\boldsymbol{p}}^T \boldsymbol{f}_{\boldsymbol{z}} / \tau\right)} \tag{3}$$

where $\boldsymbol{p} \in \mathcal{P}$ is the anchor point, while $\boldsymbol{q}, \boldsymbol{z} \in \mathcal{Q}$ are its positive match and a negative match, respectively. $\boldsymbol{f}_{\boldsymbol{p}}$ is the feature vector of $\boldsymbol{p}$ and τ is a temperature parameter.

UME Contrastive Loss. To optimize the features compatibility with the UME framework, we aim at minimizing the distance on the Grassmann Manifold between local UME descriptors of matching neighborhoods, while maximizing the distance between UME descriptors of non-matching neighborhoods. This is achieved through a novel UME Contrastive Loss, which adapts the SCL loss to manifold metric learning by evaluating (2) between UME local descriptors of the selected keypoints. During the training procedure, we sample $\{\boldsymbol{p}_k\}_{k=1}^{K}$ from $\widetilde{\mathcal{P}}$ with radius R neighborhood around each point, such that each neighborhood contains at least N points both in the source point cloud and under the ground-truth transformation, in the target point cloud. The points on the target point cloud are denoted by $\{\boldsymbol{q}_k\}_{k=1}^{K}$ (note that in general, $\boldsymbol{q}_k$ is not necessarily an actual point that exists in $\widetilde{\mathcal{Q}}$ although its neighborhood is). For each point we generate a UME local descriptors $\{\mathbf{H}_{\boldsymbol{p}_k}\}_{k=1}^{K}, \{\mathbf{H}_{\boldsymbol{q}_k}\}_{k=1}^{K}$. Let $\mathcal{M}_{\text{UME}}$ be the set of UME local descriptors of matching pairs under the ground-truth transformation and $\mathcal{N}_{\text{UME}}^{\boldsymbol{p}}$ the set of all non-matching neighborhoods to that of $\boldsymbol{p}$ w.r.t. ground truth transformation. The UME Contrastive loss is given by:

$$\mathcal{L}_{\text{UME}} = -\sum_{(\boldsymbol{p},\boldsymbol{q})\in\mathcal{M}_{\text{UME}}} \log \frac{\exp\left(s_{\boldsymbol{pq}}^{\text{UME}}/\tau\right)}{\exp\left(s_{\boldsymbol{pq}}^{\text{UME}}/\tau\right) + \sum_{\boldsymbol{z}\in\mathcal{N}_{\text{UME}}^{\boldsymbol{p}}} \exp\left(s_{\boldsymbol{pz}}^{\text{UME}}/\tau\right)} \tag{4}$$

where $s_{\boldsymbol{pq}}^{\text{UME}} = 1 - d_{pF}\left(\mathbf{H}_{\boldsymbol{p}}, \mathbf{H}_{\boldsymbol{q}}\right)$ is the UME Similarity based on the d_{pF} distance.

Registration Loss. For every matched pair of UME local descriptors we estimate the rigid transformation relating them (as detailed in Sec. 3.3). To guide the training procedure towards the task of point cloud registration, we employ an auxiliary loss that ensures the hypothesized transformation estimates from correct matches are accurate. By doing so we tune the feature extractor module to a solution that is implicitly UME compatible as well. We employ the Cube Reprojection Error (CRE) [23] as our registration loss:

$$\mathcal{L}_{reg} = \frac{1}{K}\sum_{k=1}^{K}\sum_{\boldsymbol{p}\in C_R}\left\|\left(\widehat{\boldsymbol{R}}_k - \boldsymbol{R}^{GT}\right)\boldsymbol{p} + \left(\widehat{\boldsymbol{t}}_k - \boldsymbol{t}^{GT}\right)\right\|_2 \tag{5}$$

C_R is a 3D cube centred at the origin with R length sides, $\{(\widehat{\boldsymbol{R}}_k, \widehat{\boldsymbol{t}}_k)\}_{k=1}^{K}$ are the hypnotized and $(\boldsymbol{R}^{GT}, \boldsymbol{t}^{GT})$ the ground truth transformations, respectively.

3.5 Hypothesis Selection

Once a set of hypotheses is obtained, the best hypothesis is selected by using the *Point Clouds Feature Correlation Hypothesis Testing* (PC-FCHT) [11], instead of the commonly employed sample consensus criterion: Define the feature correlation between two point clouds $\mathcal{P}$ and $\mathcal{Q}$ as function of an arbitrary rigid transformation $T \in \text{SE}(3)$ by:

$$(\mathcal{P} * \mathcal{Q})(T) = \sum_{\mathbf{p}\in\mathcal{P}} \kappa(f(\mathbf{p}), f(T^{-1}(\mathcal{Q}))) \tag{6}$$

where $f(\mathbf{p})$ is the observation coloring function and κ is a weighted correlation function of a single point defined by: $\kappa(f(\mathbf{p}), f(\mathcal{Q})) = \sum_{\mathbf{q}\in\mathcal{Q}} w_\sigma(\|\mathbf{p} - \mathbf{q}\|) f(\mathbf{p})^T f(\mathbf{q})$, and $w_\sigma : \mathbb{R}^+ \to [0,1]$ is a weight function, inversely related to the distance between given points. Inspired by classic matched filtering detection, (6) is employed to *decide* on the best hypothesis among a set of hypothesized transformations in a *global* registration framework. More specifically, let $\mathcal{D} = \{(\boldsymbol{R}_k, \boldsymbol{t}_k)\}_{k=1}^K$ be a set of hypothesis estimates. (6) is used to measure the quality of transformation estimates, where the goal is to find the transformation $\widehat{D}$ that maximizes (6), instead of the commonly used consensus size criterion.

4 Experimental Results

We evaluated UMERegRobust on the outdoor registration benchmarks of KITTI [13] and nuScenes [3] (Sec. 4.1), as well as on the indoor registration benchmark of 3DMatch [33] (Sec. 4.2). Each benchmark was compared against a wide range of SOTA registration baselines. Our findings highlight UMERegRobust's superior performance in most tested scenarios, particularly in extreme cases involving large rotations in outdoor settings, even under strict registration criteria. Additionally, UMERegRobust demonstrated comparable results on the indoor benchmark. Implementation details are introduced in the Supplementary.

4.1 Outdoor Registration Benchmarks: KITTI & nuScenes

Datasets. We follow the KITTI and nuScenes registration benchmarks as suggested in [22], which includes challenging examples of LiDAR scans that are at most $50m$ apart. Upon analyzing these benchmarks, we observed that most registration problems involve small rotations ($\leq 30°$). Therefore, we propose RotKITTI and RotnuScenes, registration benchmarks specifically designed for scenarios with large real rotations (problems distributed uniformly within $[30°, 180°]$). As shown in Fig. 1, scenarios with large rotations often exhibit significant differences between point clouds due to major differences in their field of view, causing partial overlap and sampling variations between the observations. These types of scenarios are suitable for examining registration methods aimed for loop closure in SLAM. RotKITTI and RotnuScenes are used solely for testing. Additionally, we report results on the LoKITTI and LonuScenes registration benchmarks as defined in [22]. These benchmarks focus on low overlapped scans ($< 30\%$ overlap) and emphasize large translations with rare occurrences of large rotations (contrary to RotKITTI and RotnuScenes, which include both large rotations and low overlap). Additional details on RotKITTI and RotnuScenes are in the Supplementary.

Metrics. We follow the standard evaluation metrics as defined in [6]. These are Relative Rotation Error (RRE) and Relative Translation Error (RTE). Our main evaluation metric is the Registration Recall (RR) that gather both rotation and translation estimation performance and defined by: RR@(θ, d) =

$\frac{1}{N}\sum_{n=1}^{N}[\mathrm{RRE}_n \leq \theta \wedge \mathrm{RTE}_n \leq d]$ where $[\cdot]$ is the Iverson bracket. We define two working points for the RR: *Normal precision*, RR@$(1.5°, 0.6m)$ and *Strict precision*, RR@$(1°, 0.1m)$. Evaluating registration performance with high precision is crucial in various tasks, as explained in Sec. 1. (Additional details regarding metrics are provided in the Supplementary).

Results. Tab. 1 compares various baseline methods on the KITTI and nuScenes registration benchmarks. We report the registration Recall (RR) under Normal precision (*N.*) and Strict precision (*S.*), with the best results in each column highlighted in **bold** and the second-best results underlined. On the regular KITTI benchmark, our method achieves slightly better results under normal precision compared to the previous SOTA. However, under strict precision criteria, we observe a performance gain of +9%.

In the case of RotKITTI, most of the compared methods show significant deterioration, indicating a lack of robustness to rotations. In contrast, UMERegRobust maintains high and stable performance across both the RotKITTI and regular KITTI benchmarks. While GeoTransformer [24] achieves comparable results to our method under normal precision, UMERegRobust outperforms GeoTransformer under strict precision with a notable gain of +23%. Additionally, we observe a significant gain of +45% over the GCL [22] SOTA method. Fig. 4 showcases qualitative examples from the RotKITTI benchmark, highlighting challenging scenarios of large rotations along with our registration results.

Regarding the LoKITTI benchmark, our method achieves the second-best results under normal precision, while GCL maintains the best results. GCL's optimization for low overlap and large translations gives it an inherent advantage on the LoKITTI benchmark compared to other methods. However, our method performs better on low overlap and large rotations and still achieves comparable results on low overlap and large translations, especially under strict precision.

For the nuScenes registration benchmarks, we observe an overall performance degradation in RR values compared to KITTI. This can be attributed to the low-density LiDAR used in the nuScenes dataset (32 beams) [3] compared to the high-density LiDAR (64 beams) [13] in the KITTI dataset. Similar conclusions to

Table 1: Outdoor Registration Benchmarks - Registration Recall [%]

	KITTI Benchmarks						nuScenes Benchmarks					
	KITTI		RotKITTI		LoKITTI		nuScenes		RotnuScenes		LonuScenes	
Method	*N.*	*S.*	*N.*	*S.*	*N.*	*S.*	*N.*	*S.*	*N.*	*S.*	*N.*	*S.*
FCGF [6]	75.1	73.1	11.6	3.6	17.2	6.9	58.2	37.8	5.5	5.2	1.9	0.0
Predetor [17]	88.2	58.7	41.6	35.0	33.7	<u>28.4</u>	53.9	48.1	16.5	15.7	35.6	4.2
CoFiNet [31]	83.2	56.4	62.5	30.1	11.2	1.0	62.3	56.1	27.0	<u>23.6</u>	30.3	<u>23.5</u>
GeoTrans [24]	66.3	62.6	<u>78.5</u>	<u>50.1</u>	37.8	7.2	70.7	37.9	<u>34.3</u>	13.1	48.1	17.3
GCL [22]	<u>93.9</u>	<u>78.6</u>	40.1	28.8	**72.3**	26.9	<u>82.0</u>	<u>67.5</u>	21.0	19.6	<u>62.3</u>	5.6
Ours	**94.3**	**87.8**	**81.1**	**73.3**	<u>59.3</u>	**30.2**	**85.5**	**76.0**	**51.9**	**39.7**	**70.8**	**56.3**

these observed on KITTI can be drawn from Tab. 1 regarding the nuScenes and RotnuScenes benchmarks. However, for LonuScenes, UMERegRobust achieves better results compared to GCL, particularly under strict precision. This can be explained by the influence of the SEM, which has a more significant impact on the low-density nuScenes dataset, particularly affecting the LonuScenes examples.

4.2 Indoor Registration Benchmark: 3DMatch

Dataset. We adhere to the 3DMatch registration benchmark as used in prior studies [6, 17, 24]. The dataset comprises 48 scenes for training, 8 scenes for validation, and 8 scenes for testing.

Metrics. The common metrics for the 3DMatch benchmark are Feature Matching Recall (FMR), Inlier Ratio (IR), and Registration Recall (RR). FMR is not reported in this work because it aims to predict method performance under RANSAC, while our method is RANSAC-free. Therefore, we report both IR and RR, with the later as our primary metric. Additional information regarding these metrics can be found in the supplementary material.

Results. Tab. 2 presents a comparison between UMERegRobust and various baseline methods, including additional SOTA methods primarily evaluated on indoor scenarios [4, 21, 27, 28, 32]. The 3DMatch benchmark shows saturated results for recent methods, with our method achieving comparable results in both the RR and IR metrics.

4.3 Ablation Studies

Tab. 3 presents ablation studies assessing the impact of various components within the UME-compatible coloring module. Specifically, we examine the influence of the *Sampling Equalizer Module* (SEM), the training of the feature extractor using UME-compatible losses, and their combined effect. Models trained

Table 2: 3DMatch Benchmark

Method	IR [%]	RR [%]
Lepard [21]	55.5	<u>93.5</u>
PEAL [32]	72.4	**94.6**
YOHO [27]	64.4	90.8
RoReg [28]	**86.0**	93.2
E2E [4]	53.0	91.2
FCGF [6]	56.8	85.1
Predator [17]	58.0	89.0
CoFiNet [31]	49.8	89.3
GeoTrans [24]	71.9	92.0
Ours	<u>79.7</u>	93.4

Table 3: Ablation Studies

SEM	UME - Coloring	Registration Recall [%]			
		KITTI		RotKITTI	
		N.	S.	N.	S.
✗	✗	79.8	67.3	17.5	13.1
✓	✗	81.9	68.1	17.5	12.5
✗	✓	90.3	74.9	74.4	66.2
✓	✓	94.3	87.7	81.0	73.2
✗	–	93.9	78.6	40.1	28.8
✓	–	84.4	81.4	49.6	48.4

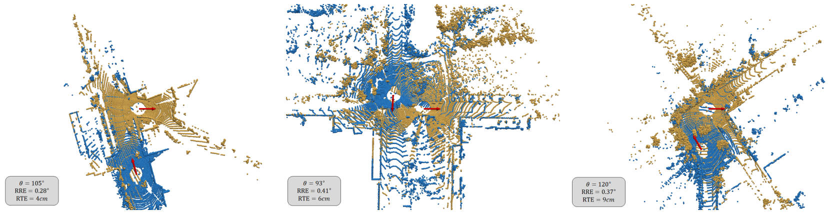

Fig. 4: Registration results using UMERegRobust. The red arrows depict the vehicle direction at the time the LiDAR scans were acquired. The relative rotation between point clouds (θ), and registration rotation and translation errors (RRE and RTE) are given for each of the examples. Best viewed zoomed in.

with non-UME-compatible coloring use a contrastive loss, as described in [6]. Their performance is compared to the performance obtained using the losses outlined in Sec. 3.4. These properties are evaluated across two categories: the regular KITTI benchmark and the RotKITTI benchmark, with registration recall reported for both Normal precision ($N.$) and Strict precision ($S.$).

When the coloring is not enforced to be UME-compatible, a performance degradation is observed across all categories and precision levels. However, when the SEM is employed, a slight performance improvement is evident. Optimal results are achieved when both UME-compatible coloring and the SEM are utilized, highlighting the synergistic benefits of their combination for the UME framework. An additional ablation study is depicted in the last two lines of Tab. 3, where one of the baseline methods (in this case, GCL [22]) is used along with the SEM. The SEM did not provide a performance gain for Normal precision but did show some improvement on RotKITTI. Thus, the SEM's contribution to performance is not stand-alone; rather, its integration with the UME maximizes their joint contribution.

5 Conclusions

In this paper, we adopt the Universal Manifold Embedding framework for the estimation of rigid transformations and extend it, so that it can accommodate scenarios involving partial overlap and differently sampled point clouds. We extend the UME framework by introducing a UME-compatible feature extraction method augmented with a unique UME contrastive loss and a sampling equalizer. These components are integrated into a comprehensive and robust registration pipeline, named UMERegRobust. UMERegRobust achieves better than state-of-the-art performance on the KITTI benchmark, especially when strict precision of $(1°, 10cm)$ is considered, and notably outperform SOTA methods on the RotKITTI benchmark where scenarios involving large rotations are considered. Although UMERegRobust outperforms SOTA methods, the performance measures of UMERegRobust, while closing the gap, still require further improvement in order to handle the considered difficult scenarios, with respect to the strict performance measures required for actual deployment on autonomous robots.

Acknowledgments

This work was supported by Israel Innovation Authority under Grant 77887.

References

1. Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.L.: D3feat: Joint learning of dense detection and description of 3d local features. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6358–6366 (2020). https://doi.org/10.1109/CVPR42600.2020.00639
2. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2), 239–256 (1992). https://doi.org/10.1109/34.121791
3. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. arXiv preprint arXiv:1903.11027 (2019)
4. Cao, Y., Shi, Y., Cheng, Z., Li, H.: End-to-end point cloud registration via rotation equivariant descriptors. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5804–5811 (2023). https://doi.org/10.1109/IROS55552.2023.10342154
5. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
6. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: ICCV (2019)
7. Cohen, T.S., Geiger, M., Köhler, J., Welling, M.: Spherical CNNs. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=Hkbd5xZRb
8. Dellenbach, P., Deschaud, J.E., Jacquet, B., Goulette, F.: Ct-icp: Real-time elastic lidar odometry with loop closure. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 5580–5586. IEEE (2022)
9. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM journal on Matrix Analysis and Applications **20**(2), 303–353 (1998)
10. Efraim, A., Francos, J.M.: Estimating rigid transformations of noisy point clouds using the universal manifold embedding. Journal of Mathematical Imaging and Vision **64**(4), 343–363 (May 2022). https://doi.org/10.1007/s10851-022-01070-6, https://doi.org/10.1007/s10851-022-01070-6
11. EFRAIM, A., Francos, J.M.: On minimizing the probability of large errors in robust point cloud registration. IEEE Open Journal of Signal Processing **5**, 39–47 (2024). https://doi.org/10.1109/OJSP.2023.3340111
12. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (Jun 1981). https://doi.org/10.1145/358669.358692, https://doi.org/10.1145/358669.358692
13. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the KITTI vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2012)

14. Hagege, R.R., Francos, J.M.: Universal manifold embedding for geometrically deformed functions. IEEE Transactions on Information Theory **62**(6), 3676–3684 (2016)
15. Horn, B.K.: Closed-form solution of absolute orientation using unit quaternions. J. Opt. Soc. Am. A **4**(4), 629–642 (Apr 1987). https://doi.org/10.1364/JOSAA.4.000629, http://josaa.osa.org/abstract.cfm?URI=josaa-4-4-629
16. Huang, J., Gojcic, Z., Atzmon, M., Litany, O., Fidler, S., Williams, F.: Neural kernel surface reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4369–4379 (2023)
17. Huang, S., Gojcic, Z., Usvyatsov, M., A. Wieser, K.S.: PREDATOR: Registration of 3d point clouds with low overlap. In: CVPR (2021)
18. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the fourth Eurographics symposium on Geometry processing. vol. 7, p. 0 (2006)
19. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. Advances in neural information processing systems **33**, 18661–18673 (2020)
20. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
21. Li, Y., Harada, T.: Lepard: Learning partial point cloud matching in rigid and deformable scenes. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
22. Liu, Q., Zhu, H., Zhou, Y., Li, H., Chang, S., Guo, M.: Density-invariant features for distant point cloud registration. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 18169–18179. IEEE Computer Society, Los Alamitos, CA, USA (oct 2023). https://doi.org/10.1109/ICCV51070.2023.01670, https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.01670
23. Mazzotti, C., Sancisi, N., Parenti-Castelli, V.: A measure of the distance between two rigid-body poses based on the use of platonic solids. In: Parenti-Castelli, V., Schiehlen, W. (eds.) ROMANSY 21 - Robot Design, Dynamics and Control. pp. 81–89. Springer International Publishing, Cham (2016)
24. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric transformer for fast and robust point cloud registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11143–11152 (June 2022)
25. Schmiedel, T., Einhorn, E., Gross, H.M.: Iron: A fast interest point descriptor for robust ndt-map matching and its application to robot localization. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3144–3151 (Sep 2015). https://doi.org/10.1109/IROS.2015.7353812
26. Thomas, H., Qi, C., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.: Kpconv: Flexible and deformable convolution for point clouds. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 6410–6419 (2019)
27. Wang, H., Liu, Y., Dong, Z., Wang, W.: You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 1630–1641 (2022)
28. Wang, H., Liu, Y., Hu, Q., Wang, B., Chen, J., Dong, Z., Guo, Y., Wang, W., Yang, B.: Roreg: Pairwise point cloud registration with oriented descriptors and local rotations. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–18 (2023). https://doi.org/10.1109/TPAMI.2023.3244951

29. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
30. Yew, Z.J., Lee, G.H.: 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In: Computer Vision – ECCV 2018. pp. 630–646. Springer International Publishing, Cham (2018)
31. Yu, H., Li, F., Saleh, M., Busam, B., Ilic, S.: Cofinet: Reliable coarse-to-fine correspondences for robust point cloud registration. In: Neural Information Processing Systems (2021), https://api.semanticscholar.org/CorpusID:239998094
32. Yu, J., Ren, L., Zhang, Y., Zhou, W., Lin, L., Dai, G.: Peal: Prior-embedded explicit attention learning for low-overlap point cloud registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 17702–17711 (June 2023)
33. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3dmatch: Learning local geometric descriptors from rgb-d reconstructions (2017)