SDRSAC: Semidefinite-Based Randomized Approach for Robust Point Cloud Registration without Correspondences

Huu M. Le[1], Thanh-Toan Do[2,3], Tuan Hoang[1], and Ngai-Man Cheung[1]

[1]Singapore University of Technology and Design [2]University of Liverpool [3]AIOZ Pte Ltd

Abstract

This paper presents a novel randomized algorithm for robust point cloud registration without correspondences. Most existing registration approaches require a set of putative correspondences obtained by extracting invariant descriptors. However, such descriptors could become unreliable in noisy and contaminated settings. In these settings, methods that directly handle input point sets are preferable. Without correspondences, however, conventional randomized techniques require a very large number of samples in order to reach satisfactory solutions. In this paper, we propose a novel approach to address this problem. In particular, our work enables the use of randomized methods for point cloud registration without the need of putative correspondences. By considering point cloud alignment as a special instance of graph matching and employing an efficient semi-definite relaxation, we propose a novel sampling mechanism, in which the size of the sampled subsets can be larger-than-minimal. Our tight relaxation scheme enables fast rejection of the outliers in the sampled sets, resulting in high quality hypotheses. We conduct extensive experiments to demonstrate that our approach outperforms other state-of-the-art methods. Importantly, our proposed method serves as a generic framework which can be extended to problems with known correspondences. [1]

1. Introduction

Point cloud registration is important in many computer vision applications, including range scan alignment [37], 3D object recognition and localization [18, 48], large scale reconstruction [2, 44]. Given two sets of points in three-dimensional (3D) Euclidean space, the objective is to search for the optimal rigid transformation, which comprises a rotation matrix $\mathbf{R}^* \in SO(3)$ and a translation vector $\mathbf{t}^* \in \mathbb{R}^3$, that optimally aligns the two input point sets. In many practical applications, the input data contains a significant amount of noise. Moreover, the overlapping region between

---

[1]Source code is available at: https://github.com/intellhave/SDRSAC.

the two point sets can be small, resulting in a large number of outliers, i.e., non-overlapping points. Therefore, the registration needs to be conducted in a robust manner so that the final estimates are not affected by the contamination. Formally, let $\mathcal{S} = \{\mathbf{s}_i \in \mathbb{R}^3\}_{i=1}^{N_s}$ and $\mathcal{D} = \{\mathbf{d}_j \in \mathbb{R}^3\}_{j=1}^{N_d}$ denote the source and destination (target) point clouds, respectively, the problem of robust rigid registration can be formulated as

$$\min_{\mathbf{R}\in SO(3), \mathbf{t}\in\mathbb{R}^3} \sum_{i=1}^{N_s} \rho\left(\|\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{d}_j\|\right), \qquad (1)$$

where the notation $\|\cdot\|$ represents the ℓ_2 norm, $\mathbf{d}_j$ is a point in the target set $\mathcal{D}$ that is closest to the transformed point $\mathbf{R}\mathbf{s}_i + \mathbf{t}$, i.e.,

$$\mathbf{d}_j = \arg\min_{\mathbf{d}_k \in \mathcal{D}} \|\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{d}_k\|, \qquad (2)$$

and ρ is a robust loss function. Here, $SO(3)$ denotes the space of rotation matrices. In order for the registration to be robust, ρ is typically chosen from a set of robust kernels [27, 1, 33]. In this work, we make use of the popular maximum consensus criterion [26], in which ρ is defined as

$$\rho(x) = \begin{cases} 0 & \text{if } x \le \epsilon, \\ 1 & \text{otherwise.} \end{cases} \qquad (3)$$

The threshold $\epsilon > 0$ is a user-defined parameter that specifies the maximum allowable distance for a correspondence pair to be considered as an inlier. Intuitively, by solving (1) with ρ defined as per (3), we search for the optimal alignment $(\mathbf{R}^*, \mathbf{t}^*)$ that maximizes the set of overlapping points, where $\mathbf{s}_i \in \mathcal{S}$ overlaps $\mathbf{d}_j \in \mathcal{D}$ if the transformation $(\mathbf{R}^*, \mathbf{t}^*)$ brings $\mathbf{s}_i$ to a new location that lies within the ball $(\mathbf{d}_j, \epsilon)$. The problem (1) is an active research topic in computer vision due to its computational complexity.

Many existing algorithms need to take as input a set of putative correspondences, which are usually obtained (as a pre-processing step) by extracting local invariant features on the given point sets, and executing multiple routines of nearest neighbor search to propose initial key-

point matches [5, 25]. Several types of 3D local features [45, 15, 52, 43] have demonstrated to provide promising results throughout a variety of challenging datasets. However, noise and contamination would degrade the quality of extracted features. Furthermore, in order for the local features to be precisely computed, many feature extractors require the surface representation to be dense; however, sparse point clouds are common in practice. In particular, it has been demonstrated in [4] that, for noisy datasets with high proportion of outliers, alignments using features have poorer results compared with using raw data. Therefore, there is interest to develop registration algorithms that directly align raw point cloud data without the need of a priori correspondences [11, 4, 9, 36, 21].

Due to the computational complexity of the problem under different input settings, randomized hypothesize-and-verify algorithms such as RANSAC [22] and its variants [17, 16, 47, 46, 42, 32] are popular approaches. Randomized techniques have been employed to address problems with known correspondences. On the other hand, similar random sampling strategy as that of RANSAC can also be applied to problems of without correspondences. Specifically, at each iteration, a minimal subset (of three points) on each point cloud can be sampled to form three pairs of correspondences, which are used for estimating and validating one hypothesis. Such procedure can then be repeated until a satisfactory solution is obtained. However, with noise and outliers, the likelihood of picking outlier-free subsets degrades rapidly. Therefore, much efforts have been devoted to develop better sampling strategies, notably the 4-Points Congruent Sets (4PCS) method proposed in [4] and its improved variant [38]. Although 4PCS provides considerable advantages over conventional randomized methods, the enumeration of (approximately) congruent sets that underpins this algorithm is the main issue when working with point clouds with large number of points and high outlier rates. In fact, for dense point clouds, 4PCS and its variants need to down-sample the point cloud before conducting the sampling process to reduce processing time.

In this paper, we address the above-mentioned limitations. Specifically, by employing a special instance of graph matching formulation, we propose a new *larger-than-minimal sampling* strategy for point cloud registration without correspondences. The advantage of our method is that the task of searching for correspondences is quickly approximated by solving a relaxed convex problem, rather than subset enumeration. This allows us to sample subsets with arbitrarily large size, in which sets of correspondences are obtained from solutions of convex semi-definite programming. These correspondences can then be used for estimating and validating hypotheses. A large subset of points in the source point cloud represent better its structure, and by identifying its corresponding points on the target set, the

two point clouds can be coarsely align faster, which can then be refined by local methods, e.g., ICP [8]. Empirical results show that the proposed method is very competitive. **Our main contributions are:**

- We apply graph matching for the registration problem without correspondences using a novel cost function to enforce robustness and a tight semidefinite (SDP) relaxation.

- From the SDP formulation, we then develop a new *larger-than-minimal* subset sampling scheme, leading to an effective randomized algorithm that outperforms other state-of-the-art methods.

2. Related Work

In practice, if the two input point clouds are coarsely aligned, the well-known Iterative Closest Point (ICP) [8] is commonly employed. Like other iterative algorithms, ICP alternates between establishing correspondences and estimating the transformation. The main drawback of this method lies in the fact that it requires a good starting point (initial pose) to prevent itself from converging to poor alignments. Additionally, in terms of robustness, ICP suffers from the same weakness as that of the least squares estimator, i.e., it is easily biased by erroneous outliers. Several works [13, 7, 40] have been proposed to improve the such shortcomings of ICP. However, these variants still need to be bootstrapped by a descent initial pose. ICP is therefore commonly employed as a local refinement procedure, which is executed after the point clouds are roughly registered by some type of global alignment algorithms.

Algorithms that offer globally optimal solutions are also actively developed in the literature. To address the initialization-dependent issue of ICP, Go-ICP [50] employs the branch and bound strategy to search in the space of $\mathbf{R}$ and $\mathbf{t}$ for the optimal transformation. However, due to the least squares objective of Go-ICP, its returned optimal solutions are still non-robust to outliers. The robustness of Go-ICP can be improved by incorporating robust loss functions in place of least squares. Another globally optimal algorithm to tackle the robust registration without correspondences was proposed by Bustos et al. [11]. Unlike Go-ICP, [11] solves (1) directly and its solution is robust to outliers. Branch and bound is also the mechanism behind [11], with a novel steographic projection implementation for fast matching query and tighter bound computation. Although the convergence to globally optimal solutions is guaranteed for these methods, they are still impractical for large datasets due to their expensive computational cost.

As previously mentioned, in some point cloud registration applications, randomized strategies – with the famous RANSAC [22] representative – are still the dominant approaches. Our work also belongs to this category. Gener-

ally speaking, for most randomized methods, the underlying sampling strategy significantly affects the run time, since the termination of a sampling mechanism depends largely on its ability to quickly identify outlier-free subsets. For the case with no correspondences, it is even harder since two outlier-free subsets, one in each input point cloud, need to be identified *and* the elements in these two subsets must form correct correspondences. Different sampling strategies have been proposed in the literature [17, 16, 49]. For instance, in problems with known correspondences, one of the notable improvements for RANSAC is LO-RANSAC (Locally Optimal RANSAC) [17], which proposes to sample *larger-than-minimal* subsets when RANSAC solution is updated. This improved strategy has been shown to significantly outperform conventional RANSAC. Randomized methods can also be improved using local refinement techniques [31, 12, 41].

In the context of robust point cloud alignment without correspondences, however, the idea of larger-than-minimal sampling has yet to be thoroughly explored. Unlike the case of known correspondences, the "inlier" set at each RANSAC iteration may not be true inliers, and purely applying LO-RANSAC to these subsets may not be of any help, while the run time is increased. Besides, even if true inliers reside in the subset, a larger-than-minimal subset may be contaminated with outliers, which can deteriorate the estimation since it is done solely by solving least squares over the sampled subsets. From the insight discussed above, it can be seen that from any larger-than-minimal subsets, if the outliers are efficiently rejected, the ability to discover good hypotheses can be accelerated. This idea is analyzed in our paper, in which we propose a new algorithm that enables the sampling of any arbitrarily large subsets. Such sampling scheme is incorporated with an oracle that allows the outliers in the sampled point sets to be efficiently discarded. Our experiments show that this newly-proposed algorithm outperforms previous approaches.

Our work is closely related to 4PCS [3], which is the state-of-the-art sampling approach that also solves the same problem as ours. Instead of randomly picking minimal subsets of three points, 4PCS works on sampled tuples of four co-planar points. This method is later improved by Super4PCS [38], where the complexity of congruent set extraction is reduced from quadratic to linear with respect to the number of points. Though efficient, especially with the improvements introduced in [38], 4PCS still suffers from the same weakness as RANSAC, e.g., the low likelihood of quickly sorting out the right subsets under the presence of high outliers. In fact, for each subset of 4 co-planar points on one point cloud, 4PCS needs to enumerate all congruent subsets on the other point cloud. Therefore, this method needs to subsample the point set into a smaller subset before the sampling is conducted.

The technique developed in this work is also inspired by the class of methods that solve 3D registration problem using graph matching [21, 35, 34] and recent semi-definite relaxation for non-rigid registration [19, 28, 30]. These algorithms, however, can only work for a small number of data points, which is very inefficient if applied directly on large point clouds. We, on the other hand, propose to employ graph matching as sub-problems embedded in a random sampling framework. This allows us to combine the strengths of both classes of methods, i.e., randomized techniques and graph matching, to derive an efficient sampling mechanism for point cloud registration.

3. Semidefinite-Based Randomized Approach

3.1. The Correspondence Problem

When the putative correspondences are not given, the task of solving (1) can be viewed as jointly estimating the transformation $(\mathbf{R}^*, \mathbf{t}^*)$ and the best subset of correspondences $\mathcal{C}^*$. Hence, if $\mathcal{C}^*$ is provided, the optimal alignment can be computed in a straightforward manner, and vice versa. With that in mind, in this section, we first introduce the correspondence problem to search for $\mathcal{C}^*$. In the latter sections, we then discuss its semidefinite (SDP) relaxation, which can be employed as a fast hypothesis generation mechanism that lies at the core of our efficient randomized algorithm.

To simplify the formulation, let us for now assume that we are given two 3D point sets $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^N$ and $\mathcal{Q} = \{\mathbf{q}_j\}_{j=1}^N$, each contains N data points. The task is to find the optimal set of correspondences $\mathcal{C}_{\mathcal{PQ}}^*$ that can be used to robustly align $\mathcal{P}$ and $\mathcal{Q}$. Note that we use different notations for input data here compared to (1), since this new problem will be used as sub-problem for solving (1), which will be clarified in the latter sections.

Let $\mathbf{X} \in \{0, 1\}^{N \times N}$ be the permutation matrix in which the element at the i−th row and j−th column (denoted by $\mathbf{X}_{i,j}$) is assigned the value of 1 if the pair $\mathbf{p}_i \in \mathcal{P}$ and $\mathbf{q}_j \in \mathcal{P}$ belongs to the set of correspondences and 0 otherwise. To account for outliers, let us further assume that the optimal solution contains $m \leq N$ pairs of correspondences. The value of m can be chosen to be greater than or equal to 3 (the size of the minimal subset), or estimated based on the known outlier ratio of the problem at hand. Note that with the introduction of m, $\mathbf{X}$ is now a sub-permutation matrix. Denote by $\hat{\mathbf{X}}$ the vectorization of the matrix $\mathbf{X}$ obtained by stacking its columns:

$$\hat{\mathbf{X}} = [\mathbf{X}_{:,1}^T \ \mathbf{X}_{:,2}^T \ \dots \ \mathbf{X}_{:,N}^T]^T, \qquad (4)$$

where $\mathbf{X}_{:,j}$ denotes the j−th column of $\mathbf{X}$. In order to search for the best correspondence assignments, consider

the following optimization problem:

$$\max_{\mathbf{X}} \quad \hat{\mathbf{X}}^T \mathbf{A} \hat{\mathbf{X}}, \tag{5a}$$

$$\text{subject to} \quad \mathbf{X}_{i,j} \in \{0,1\} \ \forall i,j \in \{1,\ldots,N\}, \tag{5b}$$

$$\sum_{j=1}^{N} \mathbf{X}_{i,j} \leq 1 \ \forall i \in \{1,\ldots,N\}, \tag{5c}$$

$$\sum_{i=1}^{N} \mathbf{X}_{i,j} \leq 1 \ \forall j \in \{1,\ldots,N\}, \tag{5d}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \mathbf{X}_{i,j} = m, \tag{5e}$$

where $\mathbf{A} \in \mathbb{R}^{N^2 \times N^2}$ is the symmetric matrix that characterizes the matching potentials between pairs of points (line segments) in $\mathcal{P}$ and $\mathcal{Q}$. In particular, we define the elements of the matrix $\mathbf{A}$ as

$$\mathbf{A}_{ab,cd} = \begin{cases} f(\mathbf{p}_a, \mathbf{p}_c, \mathbf{q}_b, \mathbf{q}_d) \text{ if} |\delta(\mathbf{p}_a, \mathbf{p}_c) - \delta(\mathbf{q}_b, \mathbf{q}_d)| \leq \gamma, \\ 0 \qquad\qquad\qquad \text{otherwise,} \end{cases} \tag{6}$$

where $a, b, c, d \in \{1..N\}$ are point indexes, $ab = a + (b - 1)N$ and $cd = c + (d - 1)N$ are the indexes for the row and column of $\mathbf{A}$, respectively; $\mathbf{p}_a, \mathbf{p}_c \in \mathcal{P}$; $\mathbf{q}_b, \mathbf{q}_d \in \mathcal{Q}$; $\gamma > 0$ is a predefined threshold and $\delta(\mathbf{p}_1, \mathbf{p}_2)$ computes the Euclidean distance between two 3D points $\mathbf{p}_1$ and $\mathbf{p}_2$; f is a function that takes as input two pairs of points $(\mathbf{p}_a, \mathbf{p}_c)$ and $(\mathbf{q}_b, \mathbf{q}_d)$ and outputs a scalar that represents matching potential for these two pairs. Typically, f in (6) is chosen to be the function that penalizes the difference in length between the two line segments. For simplicity, we choose f to be $\exp(-|\delta(\mathbf{p}_a, \mathbf{p}_c) - \delta(\mathbf{q}_b, \mathbf{q}_d)|)$.

The constraints in (5) are to assure that $\mathbf{X}$ must lie in the space of permutation matrices. Specifically, besides the binary constraint (5b) to restrict $\mathbf{X}_{i,j} \in \{0,1\}$, it is also required that each point $\mathbf{p}_i \in \mathcal{P}$ can only be assigned to at most one point in $\mathcal{Q}$ and vice versa, which is reflected by the constraints (5c) and (5d). Finally, by enforcing (5e), the optimal solution of (5) will only contain m pairs of correspondences. The solution to (5) provides the optimal assignment such that the sum of matching potentials gained yield from the corresponding pairs is maximized.

While the use of graph matching has been explored in several rigid and non-rigid registration problems [21, 35, 30, 28], most previous work consider solving graph matching for the whole input data, which is infeasible for large datasets. This work proposes to solve graph matching on very small subsets of points, then embed them into a random sampling framework. Additionally, we also propose an better formulation of the matrix $\mathbf{A}$ for the special case of robust rigid registration. Particularly, the matrix $\mathbf{A}$ is

designed based on the fact that if $\mathbf{p}_a \in \mathcal{P}$ corresponds to $\mathbf{q}_b \in \mathcal{Q}$ and $\mathbf{p}_c \in \mathcal{P}$ corresponds to $\mathbf{q}_d \in \mathcal{Q}$, since the transformation is rigid, the lengths of the two line segments $\mathbf{p}_a\mathbf{p}_c$ and $\mathbf{q}_b\mathbf{q}_d$ must be approximately the same (due to the effect of noise). With the new formulation (6), instead of assigning the matching potentials to all the elements of $\mathbf{A}$, we only allow pairs of segments whose lengths differ by a small value γ to be considered as candidate for matching, while pairs having large gap are rejected.

3.2. Semidefinite Relaxation

Let us first consider the equivalent formulation of (5). Let $\mathbf{Y} = \hat{\mathbf{X}}\hat{\mathbf{X}}^T \in \mathbb{R}^{N^2 \times N^2}$, the problem (5) becomes

$$\max_{\mathbf{X},\mathbf{Y}} \quad \text{trace}(\mathbf{A}\mathbf{Y}), \tag{7a}$$

$$\text{subject to} \quad \mathbf{Y} = \hat{\mathbf{X}}\hat{\mathbf{X}}^T, \tag{7b}$$

$$\text{trace}(\mathbf{Y}) = m, \tag{7c}$$

$$0 \leq \mathbf{X}_{i,j} \leq 1 \ \forall i,j, \tag{7d}$$

$$(5c), (5d), (5e). \tag{7e}$$

Note that in (7), the constraint (5b) can be removed without affecting the equivalence between (7) and (5). Indeed, since $\text{trace}(\mathbf{Y}) = m$, it must hold that $\sum_{i,j} \mathbf{X}_{i,j}^2 = m$. On the other hand, due to the constraint (5e), $\sum_{i,j} \mathbf{X}_{i,j} = m$. These conditions result in $\sum_{i,j} \mathbf{X}_{i,j}^2 = \sum_{i,j} \mathbf{X}_{i,j}$. Also, due to the condition $0 \leq \mathbf{X}_{i,j} \leq 1$, $\mathbf{X}_{i,j}$ can only be either 0 or 1. In other word, $\mathbf{X}$ can now be constrained in the convex hull of the sub-permutation matrices [39].

By introducing $\mathbf{Y}$, the problem is lifted into the domain of $\mathbb{R}^{N^2 \times N^2}$ and the binary constraint of $\mathbf{X}$ can be relaxed without changing the solution of the problem. However, the problem (7) is still non-convex due to the rank-one constraint (7b). In order to approximate (7), we employ the common convex relaxation approach, in which (7b) is relaxed to the semidefinite constraint $\mathbf{Y} - \hat{\mathbf{X}}\hat{\mathbf{X}}^T \succeq \mathbf{0}$. Then, we arrive at the following convex optimization problem:

$$\max_{\mathbf{X},\mathbf{Y}} \quad \text{trace}(\mathbf{A}\mathbf{Y}), \tag{8a}$$

$$\text{subject to} \quad \mathbf{Y} - \hat{\mathbf{X}}\hat{\mathbf{X}}^T \succeq \mathbf{0}, \tag{8b}$$

$$(5c), (5d), (5e), (7c), (7d). \tag{8c}$$

The problem (8) introduced above is a convex semidefinite program (SDP), whose globally optimal solution can be obtained using many off-the-shelf SDP solvers. In this work, we use SDPNAL+ [51] throughout all the experiments.

3.2.1 Tightening the Relaxation

As (8) is solved as an approximation for (7) with rank-one constraint, one would expect that the two solutions to be as

close as possible. Inspired by [30], we add the following constraints to (8) to tighten the relaxation:

$$\mathbf{Y}_{ab,cd} \leq \begin{cases} 0, & \text{if } a = c, b \neq d, \\ 0, & \text{if } b = d, a \neq c, \\ \min(\mathbf{X}_{ab}, \mathbf{X}_{cd}), & \text{otherwise.} \end{cases} \quad (9)$$

The intuition behind (9) is that one point in $\mathcal{S}$ is not allowed to match with more than one point in $\mathcal{D}$ and vice versa. Also, since $\mathbf{Y}_{ab,cd} = \mathbf{X}_{ab}\mathbf{X}_{cd}$ and these are binary numbers, it must hold that $\mathbf{Y}_{ab,cd} \leq \min(\mathbf{X}_{ab}, \mathbf{X}_{cd})$.

In addition, thanks to the special case of robust registration, (8) can be further tightened. Observe that, based on the discussion of formulating the matrix $\mathbf{A}$, the following constraints can be enforced:

$$\mathbf{Y}_{ab,cd} = 0 \text{ if } \mathbf{A}_{ab,cd} = 0, \quad (10)$$

which means if the pairs $\mathbf{p}_a\mathbf{p}_c$ and $\mathbf{p}_b\mathbf{p}_d$ differ too much in length (more than γ, we directly disallow them to be matched.

Finally, with the addition of (9) and (10), our SDP relaxation becomes:

$$\max_{\mathbf{X},\mathbf{Y}} \quad \text{trace}(\mathbf{A}\mathbf{Y}) \quad (11a)$$

$$\text{subject to} \quad (8b), (7c), (7d), (8c), (9), (10) \quad (11b)$$

Note that (11) is still a convex SDP since the additional constraints (9) and (10) are linear.

3.3. Projection of Solutions to Permutation Matrix

After solving (11) up to its global optimality using a convex solver, the remaining task is to project its solution back to the space of permutation matrices. This task can be done using different strategies [23]. In this work, we apply the linear assignment problem [10], in which the projection can be computed efficiently by solving a linear program (LP). Specifically, let $\tilde{\mathbf{X}}$ be the optimal solution of (11). The LP for projection can be formulated as

$$\max_{\mathbf{X}} \quad \langle \mathbf{X}, \tilde{\mathbf{X}} \rangle$$

$$\text{subject to} \quad 0 \leq \mathbf{X}_{i,j} \leq 1 \quad \sum_i \mathbf{X}_{i,j} = 1, \quad \sum_j \mathbf{X}_{i,j} = 1$$

$$(12)$$

where $\langle \cdot, \cdot \rangle$ represents the inner product of two matrices.

The solution of (12) provides us with a set of N correspondences. However, in (5), we only want to pick m pairs. To tackle this with a simple heuristic, observe that in the optimal solution $\mathbf{X}^*$ of (5), only m rows/columns of $\mathbf{X}^*$ contain the value of 1, while the rest of the rows/columns contain all zeros. Therefore, from correspondence set obtained from (12), we associate each pair of correspondence $(\mathbf{p}_i, \mathbf{q}_j)$ with a score that is equal to $\tilde{\mathbf{X}}_{i,j}$. Finally, m pairs

with the highest scores are chosen as the approximate solution for (5). This approach has been empirically shown to be very efficient throughout our experiments.

3.4. Main Algorithm

Algorithm 1 SDRSAC

Require: Input data $\mathcal{S}$ and $\mathcal{D}$, `max_iter`, `inner_iters`, size of sampled subsets N_{sample}

1: `iter`$\leftarrow 0$; `best_score`$\leftarrow 0$;
2: **while** `iter`$<$ `max_iter` **do**
3:    $\mathcal{S}' \leftarrow$ Randomly sample N_{sample} points from $\mathcal{S}$
4:    **for** $t = 1$ to `inner_iters` **do**
5:       $\mathcal{D}' \leftarrow$ Randomly sample N_{sample} points from $\mathcal{D}$
6:       $\{\mathcal{M}, \mathbf{R}, \mathbf{t}\} \leftarrow$ `SDRMatching` $(\mathcal{S}, \mathcal{D}, \mathcal{S}', \mathcal{D}')$
      /*Described in Alg. 2 */
7:       **if** $|\mathcal{M}| >$ `best_score` **then**
8:          `best_score`$\leftarrow |\mathcal{M}|$; $\mathbf{R}^* \leftarrow \mathbf{R}$; $\mathbf{t}^* \leftarrow \mathbf{t}$
9:       **end if**
10:   **end for**
11:   `iter`$\leftarrow$ `iter`$+$ `inner_iters`
12:   $T \leftarrow$ Number of iterations that satisfies the probabilistic stopping criteria (Sec. 3.5)
13:   **if** `iter`$\geq T$ **then**
14:      `return`
15:   **end if**
16: **end while**
17: **return** Best transformation $(\mathbf{R}^*, \mathbf{t}^*)$, `best_score`

Algorithm 2 `SDRMatching`

Require: Input data $\mathcal{S}$ and $\mathcal{D}$, sampled subsets $\mathcal{S}'$ and $\mathcal{D}'$, threshold ϵ.
1: $\mathbf{A} \leftarrow$ Matrix generated using Eq. (6) with $\mathcal{P} = \mathcal{S}'$ and $\mathcal{Q} = \mathcal{D}'$
2: $\tilde{\mathbf{X}} \leftarrow$ Solve (11) with $\mathbf{A}$ generated from Step 1.
3: $\mathbf{X} \leftarrow$ Solve (12) with $\tilde{\mathbf{X}}$ obtained from Step 2.
4: $\mathcal{M}' \leftarrow \{(\mathbf{s}'_i \in \mathcal{S}', \mathbf{d}'_j \in \mathcal{D}') | \mathbf{X}_{i,j} = 1\}$
5: $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}}) \leftarrow$ Estimate transformation based on correspondence set $\mathcal{M}'$ [20, 29].
6: $(\mathbf{R}, \mathbf{t}) \leftarrow$ Refinement using ICP, initialized by $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$
7: $\mathcal{M} \leftarrow \{(\mathbf{s}_i \in \mathcal{S}, \mathbf{d}_j \in \mathcal{D}) \mid \|\mathbf{R}\mathbf{s}_i + \mathbf{t} - \mathbf{d}_j\| \leq \epsilon\}$
  /*$\mathbf{d}_j$ is defined in (2) */
8: **return** $\mathcal{M}, \mathbf{R}, \mathbf{t}$

Although (11) is convex, solving it for large value of N is inefficient since the lifted variable $\mathbf{Y}$ belongs to the domain of $\mathbb{R}^{N^2 \times N^2}$. However, for a small value of N (typically, $N \leq 20$), this problem can be optimized efficiently and (11) provides a good approximation for the original problem (7). Taking advantage of this observation, one can

develop a sampling approach where the number of points in each sample can be of any size, up to the limit that the employed solver can handle (11) efficiently. More specifically, at each iteration, two subsets of points $\mathcal{S}' \subseteq \mathcal{S}$ and $\mathcal{D}' \subseteq \mathcal{D}$ are randomly sampled from $\mathcal{S}$ and $\mathcal{D}$, respectively. The cardinality of $\mathcal{S}'$ and $\mathcal{D}'$ can be controlled depending on the capability of the employed convex solver. By solving (11), the outliers in the sampled subsets can be rejected and the best subset in each sample is retained for estimating the transformation. Like other randomized paradigms, this process can be repeated after a fixed number of iterations or until the stopping criteria is satisfied. The insight behind this strategy is that by sampling large subsets at each iteration, one is more likely to encounter the right subsets that contains inliers, since if any sampled subset is contaminated, the outliers can be rejected efficiently by solving the SDP approximation discussed in Section 3.1. Algorithm 1 summarizes our method.

3.5. Stopping Criterion

We follow [4] to derive our stopping criterion. Let us denote by p_I be the probability of selecting one inlier (correct correspondence pair), and by T the number of trials. Note that from statistics [24], the expected inlier rate in a sample of size N_{sample} is also p_I. Moreover, after running Alg. 2, only m pairs of correspondences remain, allowing us to compute the stopping criterion based on m. Denote by p_f the probability that the algorithm fails to find an all-inlier correspondence set after T trials, p_f can be computed as

$$p_f = (1 - p_I^m)^T, \tag{13}$$

Therefore, in order to get a success probability to be greater than p_s, we must have the number of iterations greater than

$$T \geq \frac{\log(1 - p_s)}{\log(1 - p_I^m)}. \tag{14}$$

Since the real inlier rate p_I is not known in advance, following common practice of several randomized methods, we update this value during the sampling process, i.e., p_I is iteratively updated using the inlier ratio of the current best-so-far solutions.

4. Experimental Results

To evaluate the performance of our proposed algorithm (SDRSAC), we conduct experiments on multiple sets of synthetic and real datasets and compare our approach against several state-of-the-art algorithms that can be used to solve point cloud registration without correspondences, including ICP [8], Trimmed ICP (TrICP) [13], Iteratively Re-weighted Least Squares (IRLS) [6], 4PCS [4] and its improvement Super4PCS [38]. Within the class of globally

optimal algorithms, we also compare SDRSAC against Go-ICP [50] and its robust version with trimming (TrGoICP). Note that conventional RANSAC method [22] performs poorly for this type of problem, and in many cases it turns into a brute-force type algorithm. Thus, to save space, we only show results from established methods listed above.

As we focus on validating the effectiveness of robust global registration, throughout the experiments, we measure and report the number of matches (objective function of (1)) and run time for each method.

All experiments are executed on a standard Ubuntu machine with 16GB of RAM. SDRSAC, ICP, TrimmedICP were implemented using MATLAB. For 4PCS, Super4PCS [38], Go-ICP [50], we use the released C++ code and the parameters suggested by the authors. All results of randomized methods are reported by averaging the outcomes obtained from 20 different runs. In the following, we only report representative results. More results, implementation details and extensions to the case with known correspondences can be found in the supplementary material.

4.1. Synthetic Data

We first evaluate the performance of SDRSAC on synthetically generated data. The bunny point cloud from the Stanford dataset[2] is loaded and uniformly sampled to yield a source point cloud $\mathcal{S}$ containing $N_s = 10,000$ points. To generate the target set $\mathcal{D}$, we apply a random transformation $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$ to $\mathcal{S}$. Each point in $\mathcal{D}$ is then perturbed with a Gaussian noise of zero mean and variance of $\sigma_{\text{noise}} = 0.01$. To simulate partial overlapping, we randomly pick and remove $r\%$ of the points in $\mathcal{D}$. In order to evaluate the performance of the algorithms with different outlier ratios, we repeat the experiments with $r = 10, 15, \ldots, 50\%$. The threshold ϵ in (3) was chosen to be 0.01 for all the methods. For SDRSAC, we choose the sample size to be $N_{\text{sample}} = 16$, and $m = 4$. (The choice of N_{sample} and m will be studied in Sec. 4.3). Figure 1 shows the number of matches (inliers) and run time for all the methods. It is evident that SDRSAC outperforms other methods in terms of correspondences obtained with faster (or comparable) execution time. Note that the run time of SDRSAC is very close to Super4PCS, though SDRSAC consistently attains higher solution quality. The performance of ICP, as anticipated, is unstable due to the effect of initialization (i.e., wrong initialization may lead to poor convergence), as shown in Fig. 1. Moreover, as the outlier ratio increases, the consensus sizes produced by most ICP-based methods are poor due to their non-robustness.

4.2. Real Data

In this section, we evaluate the performance of our proposed method on real datasets and compare it with existing
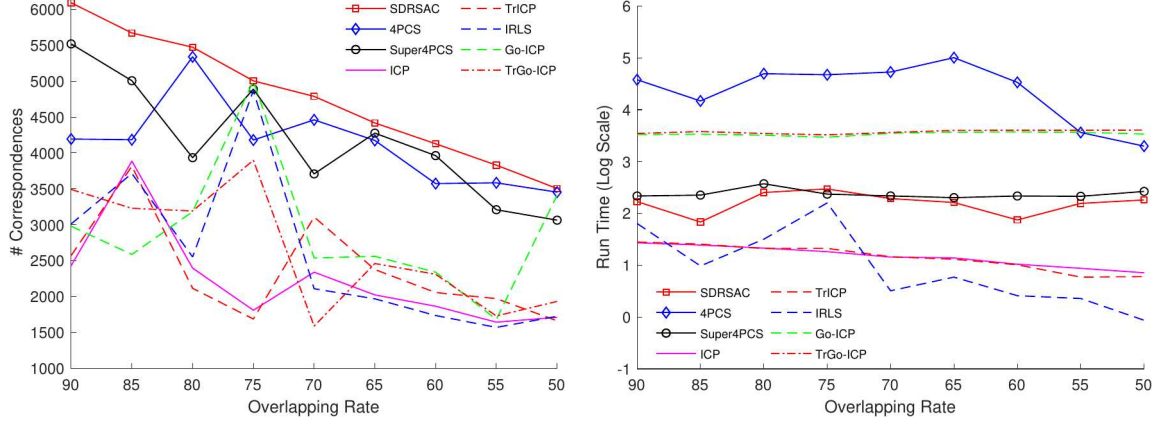
---

[2]http://graphics.stanford.edu/data/3Dscanrep/

Figure 1. Results for experiments with synthetic data. Left: Number of correspondences (Objective of (1)). Right: Run time (in log scale).

Pairs		SDRSAC	4PCS	S-4PCS	ICP	TrICP	IRLS	GoICP	TrGoICP
Office1 1,2	#Corrs	**6990**	5922	6388	2433	4455	2429	2664	6828
	Time(s)	8.69	11.18	10.05	4.69	4.35	3.16	50.9	33.4
Office1 8, 9	#Corrs	**4353**	4184	2901	2132	1205	2125	2253	4101
	Time(s)	9.53	15.08	15.06	4.44	4.45	4.06	63.2	38.3
Office1 15, 16	#Corrs	**4992**	4954	4933	655	487	673	1513	1413
	Time(s)	7.56	15.95	15.45	4.35	4.63	4.19	60.3	37.6
Office1 51, 52	#Corrs	**4490**	2998	3714	60	612	178	554	848
	Time(s)	10.05	15.45	15.03	4.29	4.55	4.39	60.5	38.5
Living1 1, 2	#Corrs	**5817**	5602	4590	3609	4599	4016	4060	5655
	Time(s)	8.34	15.09	15.39	4.26	4.96	5.13	32.5	35.6
Living1 25, 26	#Corrs	**4768**	3615	4327	1350	1364	4240	4398	4567
	Time(s)	9.27	15.07	15.95	4.34	4.42	4.35	65.3	36.1
Living1 54, 55	#Corrs	**5558**	5155	5433	4104	4783	4009	5120	5250
	Time(s)	9.2	15.05	15.65	4.29	4.32	5.19	61.2	35.9
Living1 32, 33	#Corrs	**5570**	5459	3269	2028	1519	2808	2334	5319
	Time(s)	8.32	15.15	15.23	4.65	4.44	5.1	34.6	33.3

Table 1. Results for real data experiments. For each pairs, the first row is the number of correspondences (#Corrs) and the second row shows the run time in second. Note that S-4PCS represent Super4PCS.
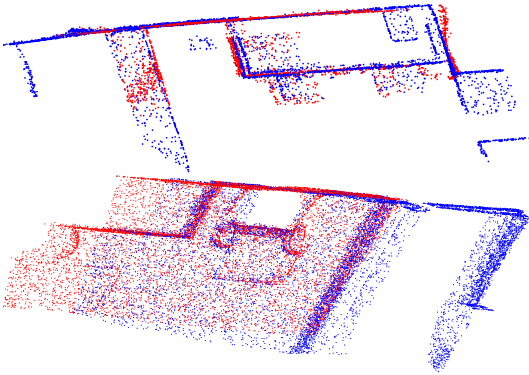

Figure 2. Examples of alignments using SDRSAC. From top to bottom: Office 8 and Office 9; LivingRoom 1 and LivingRoom 2.

approaches. The input point clouds for this experiment are obtained from the challenging Redwood 3D dataset [14]. We randomly pick eight pairs of point clouds from the Office and Living Room repository, then uniformly sampled the point sets to obtain $10,000$ points on each input set. The input threshold ϵ in (3) was chosen to be in the range of 0.01 to 0.05 and for each set of input data, ϵ was chosen to be the same for all benchmarking methods.

Table 1 shows the number of matches and run time for each method. The sample qualitative results are displayed in Figure 2. As can be seen, on average, SDRSAC obtains higher matches compared to other competitors. We also observe that local methods such as ICP or IRLS can sometimes converge fast to very good results, but are unstable due to the effect of bad initializations. The robust version TrGo-ICP performs slightly better than Go-ICP, but

also fails in some cases due to high outlier ratios residing in the input data.

4.3. Ablation Studies

This section analyzes the effect of different parameter settings to the performance of our proposed method, and suggest the choices for the hyperparameters.
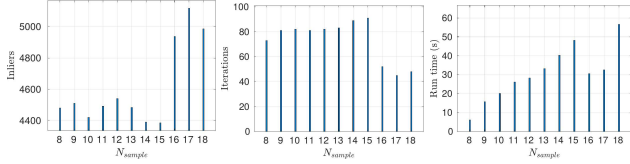
4.3.1 Effect of Sample Size N_{sample}

Figure 3. Performance analysis for different values of N_{sample}. Left: Consensus size, Middle:# Iterations, Right: Run time.

Fig. 3 plots the accuracy (consensus size), number of iterations (to satisfy the stopping criterion (14)), and the total run time when N_{sample} increases from 8 to 18 and $m = 4$ for a synthetic dataset (generated as described in Sec. 4.1) with $N = 10,000$ and $p_I \approx 30\%$ (the plotted results are the median over 20 runs). Our obtained results conform to statistical theory [24]. Specifically, the inlier rates in the subsets of N_{sample} points have a mean of p_I and standard deviation (SD) of $\sqrt{\frac{p_I(1-p_I)}{N_{\text{sample}}}}$. From Fig. 3, we observe that for small N_{sample} ($N_{\text{sample}} < 16$), the obtained consensus sizes are low and the algorithm requires a large number of iterations. In particular, if N_{sample} is small, then SD of the inlier rates is large among sampled subsets, thus some subsets may be contaminated with large numbers of outliers, which affects the quality and stability of our algorithm. When N_{sample} reaches around 16, SD is $\sqrt{\frac{0.3*0.7}{16}} \approx 0.1$, hence the inlier rates vary only slightly among subsets, leading to a stable performance, as shown in Fig. 3 (left, middle). While it is good to use large N_{sample} from the statistical point of view, SDP solver would take more time at each iteration for large N_{sample}, resulting in longer overall run time as shown in Fig. 3 (right). Empirically, we found that $N_{\text{sample}} = 16$ provides a good trade-off between algorithm stability (accuracy) and run time in many settings.

4.3.2 Effect of m

This experiment is conducted to study on the effect of m (introduced in Sec. 3.1) on the solution quality and run time of our approach. We repeat the experiment for a synthetic dataset containing $N = 5,000$ points per point cloud with 10% outliers. The sample size N_{sample} is set to 16. All results are obtained as median over 20 runs. Fig. 4 (left) plots the number of inliers obtained at termination (using

the stopping criterion (14)). Observe that the performance is quite stable at different values of m, which demonstrates the effectiveness of SDP to reject outliers and the strengths of our proposed sampling scheme. In Fig. 4 (right), we also plot the run time required until termination. Apparently, although the solution qualities for different values of m are similar as discussed above, the run time is affected by m. Specifically, as m increases, SDRSAC takes more iterations before the stopping criterion (14) is met. This can be explained by recalling from (13) that large m increase the failure probability p_f. Moreover, for any sampled subset, the expected number of inliers is $N_{\text{sample}} \times p_I$, hence if m is larger than this value, outliers may be included. In our experiments, the choice of $m = 4$ works best in most scenarios (the minimal case of $m = 3$ may not be very stable due to noise in 3D data points and the 3 correspondences could be near each other spatially).
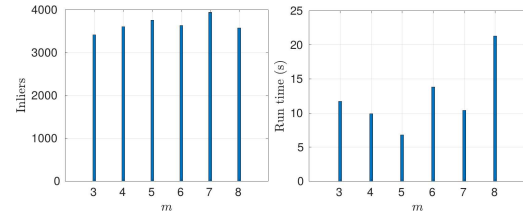

Figure 4. Plots of consensus sizes (left) and run times (right) over different values of m.

5. Conclusions

We have presented a novel and efficient randomized approach for robust point cloud registration without correspondences. We apply graph matching formulation for the correspondence problem. We propose a novel cost function and a tight SDP relaxation scheme. We embed the formulation into a new sampling strategy which samples larger-than-minimal subsets. Extensive experiments with several synthetic and real datasets show that our algorithm is competitive. Our framework can also be extended for the problems with known correspondences.

Acknowledgement

References

[1] Khurrum Aftab and Richard Hartley. Convergence of iteratively re-weighted least squares to robust m-estimators. In *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, pages 480–487. IEEE, 2015.

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[3] Sameer Agarwal, Noah Snavely, and Steven M Seitz. Fast algorithms for l problems in multiview geometry. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[4] Dror Aiger, Niloy J Mitra, and Daniel Cohen-Or. 4-points congruent sets for robust pairwise surface registration. In *ACM Transactions on Graphics (TOG)*, volume 27, page 85. ACM, 2008.

[5] Aitor Aldoma, Zoltan-Csaba Marton, Federico Tombari, Walter Wohlkinger, Christian Potthast, Bernhard Zeisl, Radu Bogdan Rusu, Suat Gedikli, and Markus Vincze. Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation. *IEEE Robotics & Automation Magazine*, 19(3):80–91, 2012.

[6] Per Bergström and Ove Edlund. Robust registration of point sets using iteratively reweighted least squares. *Computational Optimization and Applications*, 58(3):543–561, 2014.

[7] Per Bergström and Ove Edlund. Robust registration of surfaces using a refined iterative closest point algorithm with a trust region approach. *Numerical Algorithms*, 74(3):755–779, 2017.

[8] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–607. International Society for Optics and Photonics, 1992.

[9] Thomas M Breuel. Implementation techniques for geometric branch-and-bound matching methods. *Computer Vision and Image Understanding*, 90(3):258–294, 2003.

[10] Rainer E Burkard and Eranda Cela. Linear assignment problems and extensions. In *Handbook of combinatorial optimization*, pages 75–149. Springer, 1999.

[11] Álvaro Parra Bustos, Tat-Jun Chin, Anders Eriksson, Hongdong Li, and David Suter. Fast rotation search with stereographic projections for 3d registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2227–2240, 2016.

[12] Zhipeng Cai, Tat-Jun Chin, Huu Le, and David Suter. Deterministic consensus maximization with biconvex programming. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 685–700, 2018.

[13] Dmitry Chetverikov, Dmitry Svirko, Dmitry Stepanov, and Pavel Krsek. The trimmed iterative closest point algorithm. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 3, pages 545–548. IEEE, 2002.

[14] Sungjoon Choi, Qian-Yi Zhou, and Vladlen Koltun. Robust reconstruction of indoor scenes. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 5556–5565. IEEE, 2015.

[15] Chin Seng Chua and Ray Jarvis. Point signatures: A new representation for 3d object recognition. *International Journal of Computer Vision*, 25(1):63–85, 1997.

[16] Ondrej Chum and Jiri Matas. Matching with prosac-progressive sample consensus. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 220–226. IEEE, 2005.

[17] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *DAGM*. Springer, 2003.

[18] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 998–1005. Ieee, 2010.

[19] Nadav Dym, Haggai Maron, and Yaron Lipman. Ds++: a flexible, scalable and provably tight relaxation for matching problems. *ACM Transactions on Graphics (TOG)*, 36(6):184, 2017.

[20] David W Eggert, Adele Lorusso, and Robert B Fisher. Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine vision and applications*, 9(5-6):272–290, 1997.

[21] Olof Enqvist, Klas Josephson, and Fredrik Kahl. Optimal correspondences from pairwise constraints. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1295–1302. IEEE, 2009.

[22] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[23] Fajwel Fogel, Rodolphe Jenatton, Francis Bach, and Alexandre d'Aspremont. Convex relaxations for permutation problems. In *Advances in Neural Information Processing Systems*, pages 1016–1024, 2013.

[24] David Forsyth. *Probability and Statistics for Computer Science*. Springer, 2018.

[25] Ran Gal and Daniel Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Transactions on Graphics (TOG)*, 25(1):130–150, 2006.

[26] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[27] Peter J Huber et al. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[28] M Hullin, R Klein, T Schultz, and A Yao. Efficient lifted relaxations of the quadratic assignment problem. 2017.

[29] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.

[30] Itay Kezurer, Shahar Z Kovalsky, Ronen Basri, and Yaron Lipman. Tight relaxation of quadratic matching. In *Computer Graphics Forum*, volume 34, pages 115–128. Wiley Online Library, 2015.

[31] Huu Le, Tat-Jun Chin, and David Suter. An exact penalty method for locally convergent maximum consensus. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017.

[32] Huu Le, Tat-Jun Chin, and David Suter. Ratsac-random tree sampling for maximum consensus estimation. In *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2017.

[33] Huu Le, Anders Eriksson, Michael Milford, Thanh-Toan Do, Tat-Jun Chin, and David Suter. Non-smooth m-estimator for maximum consensus estimation. 2018.

[34] D Khuê Lê-Huu and Nikos Paragios. Alternating direction graph matching. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4914–4922. IEEE, 2017.

[35] Marius Leordeanu and Martial Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1482–1489. IEEE, 2005.

[36] Hongdong Li and Richard Hartley. The 3d-3d registration problem revisited. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[37] Feng Lu and Evangelos Milios. Globally consistent range scan alignment for environment mapping. *Autonomous robots*, 4(4):333–349, 1997.

[38] Nicolas Mellado, Dror Aiger, and Niloy J Mitra. Super 4pcs fast global pointcloud registration via smart indexing. In *Computer Graphics Forum*, volume 33, pages 205–215. Wiley Online Library, 2014.

[39] NS Mendelsohn and AL Dulmage. The convex hull of sub-permutation matrices. *Proceedings of the American Mathematical Society*, 9(2):253–254, 1958.

[40] Jeff M Phillips, Ran Liu, and Carlo Tomasi. Outlier robust icp for minimizing fractional rmsd. In *3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, pages 427–434. IEEE, 2007.

[41] Pulak Purkait, Christopher Zach, and Anders Eriksson. Maximum consensus parameter estimation by reweighted l1 methods. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 312–327. Springer, 2017.

[42] Rahul Raguram, Ondrej Chum, Marc Pollefeys, Jiri Matas, and Jan-Michael Frahm. Usac: a universal framework for random sample consensus. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):2022–2038, 2013.

[43] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.

[44] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[45] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 357–360. ACM, 2007.

[46] Ben J Tordoff and David W Murray. Guided-mlesac: Faster image transform estimation by using matching priors. *IEEE transactions on pattern analysis and machine intelligence*, 27(10):1523–1535, 2005.

[47] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 78(1):138–156, 2000.

[48] Ngoc-Trung Tran, Dang-Khoa Le Tan, Anh-Dzung Doan, Thanh-Toan Do, Tuan-Anh Bui, Mengxuan Tan, and Ngai-Man Cheung. On-device scalable image-based localization via prioritized cascade search and fast one-many ransac. *IEEE Transactions on Image Processing*, 28(4):1675–1690, 2019.

[49] Quoc Huy Tran, Tat-Jun Chin, Wojciech Chojnacki, and David Suter. Sampling minimal subsets with large spans for robust estimation. *International journal of computer vision*, 106(1):93–112, 2014.

[50] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: a globally optimal solution to 3d icp point-set registration. *IEEE transactions on pattern analysis and machine intelligence*, 38(11):2241–2254, 2016.

[51] Liuqin Yang, Defeng Sun, and Kim-Chuan Toh. Sdpnal + +: a majorized semismooth newton-cg augmented lagrangian method for semidefinite programming with non-negative constraints. *Mathematical Programming Computation*, 7(3):331–366, 2015.

[52] Hao Zhang, Wenjun Zhou, Christopher Reardon, and Lynne E Parker. Simplex-based 3d spatio-temporal feature description for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2059–2066, 2014.