ICOS: Efficient and Highly Robust Rotation Search and Point Cloud Registration with Correspondences

Lei Sun

Abstract—Rotation search and point cloud registration are two fundamental problems in robotics and computer vision, which aim to estimate the rotation and the transformation between the 3D vector sets and point clouds, respectively. Due to the presence of outliers, probably in very large numbers, among the putative vector or point correspondences in real-world applications, robust estimation is of great importance. In this paper, we present ICOS (Inlier searching using COnpatible Structures), a novel, efficient and highly robust solver for both the correspondence-based rotation search and point cloud registration problems. Specifically, we (i) propose and construct a series of compatible structures for the two problems where various invariants can be established, and (ii) design three time-efficient frameworks, the first for rotation search, the second for known-scale registration and the third for unknown-scale registration, to filter out outliers and seek inliers from the invariant-constrained random sampling based on the compatible structures proposed. In this manner, even with extreme outlier ratios, inliers can be sifted out and collected for solving the optimal rotation and transformation effectively, leading to our robust solver ICOS. Through plentiful experiments over standard datasets, we demonstrate that: (i) our solver ICOS is fast, accurate, robust against over 95% outliers with nearly 100% recall ratio of inliers for rotation search and both known-scale and unknown-scale registration, outperforming other state-of-the-art methods, and (ii) ICOS is practical for use in multiple real-world applications.

Index Terms—Rotation Search, point cloud registration, robust estimation, invariants, compatibility

I. INTRODUCTION

Rotation search and point cloud registration are two cornerstones in robotics and computer vision. The former has found broad applications in attitude estimation [46], image stitching [6], etc, while the latter has been widely applied in 3D scene reconstruction [9], [12], [23], object localization and recognition [16], [20], [37], medical imaging [3], Simultaneous Localization and Mapping (SLAM) [52], etc.

A. Problems Formulation

Rotation search aims to find the best rotation that aligns two 3D vector sets in different coordinate frames, which can be formulated as follows.

Given two 3D vector sets \( U = \{ u_i \}_{i=1}^N \) and \( V = \{ v_i \}_{i=1}^N \) where we consider tuple \( (u_i, v_i) \) as a vector correspondence, the following relation is satisfied:

\[
RU_i + \epsilon_i = v_i,
\]

where \( R \in SO(3) \) [10], [45] denotes the rotation matrix, and \( \epsilon_i \in \mathbb{R}^3 \) represents the measurement of Gaussian noise with isotropic covariance \( \sigma^2 I_3 \) (\( \sigma \) is the standard deviation and \( I_3 \in \mathbb{R}^{3 \times 3} \) is the identity matrix) and mean \( \mu = 0 \) w.r.t. the \( i_th \) vector correspondence. Thus, the rotation search problem can be mathematically defined as:

\[
\min_{R \in SO(3)} \sum_{i=1}^{N} \|RU_i - v_i\|^2,
\]

where globally optimal solution \( \hat{R} \) is the best rotation desired.

Point cloud registration consists in estimating the transformation (including scale, rotation and translation) between two given 3D point clouds. Now we introduce the correspondence-based point cloud registration problem.
With two 3D point clouds $\mathcal{P} = \{ P_i \}_{i=1}^N$ and $\mathcal{Q} = \{ Q_i \}_{i=1}^N$, where tuple $(P_i, Q_i)$ represents a point correspondence, we can have the following relation:

$$sR P_i + t + \varepsilon_i = Q_i, \quad (3)$$

where $s > 0$ is the scale, $R \in SO(3)$ is the rotation matrix, $t \in \mathbb{R}^3$ is the translation vector, and $\varepsilon_i \in \mathbb{R}^3$ represents the measurement of Gaussian noise similar to that in Problem 1. The generalized point cloud registration problem can be defined as follows.

**Problem 2 (Generalized Point Cloud Registration):** Solving the generalized point cloud registration problem under the perturbation of noise is equivalent to the following minimization problem:

$$\min_{s>0, R \in SO(3), t \in \mathbb{R}^3} \sum_{i=1}^{N} \|sR P_i + t - Q_i\|^2, \quad (4)$$

where global minimizers $(\hat{s}, \hat{R}, \hat{t})$ are the optimal solutions to this registration problem.

When the scale aligning the two point clouds is fixed, which means scale $s = 1$, we can directly omit $s$ in Problem 2 called the known-scale registration problem. However, the scale could vary, which, for instance, exists in the cross-source point cloud registration problems. In this case, the scale needs to be estimated additionally, and these problems are called unknown-scale registration.

Our work intends to solve both the known-scale and the unknown-scale registration in the presence of both noise and a great number of outliers among the putative correspondences, providing a highly robust backend for real applications.

### B. Motivation and Contributions

**Motivation.** The correspondence-based rotation search and point cloud registration approaches are often confronted with one difficulty: in many real-world scenes, the input correspondences are very likely to be corrupted by outliers; for example, the feature matchers (e.g. SIFT [30] and SURF [3] for 2D images and FPFH [41] for 3D point clouds) may generate mismatched or spurious results owing to noise, repetitive patterns, low-textured scenes, etc. Therefore, in real applications, robust estimation solvers should be of paramount significance.

When it comes to robust estimation, RANSAC [18] is a common robust heuristic [1], but it has a critical disadvantage: its runtime increases exponentially with the outlier ratio among the correspondences. Thus, RANSAC could be impractical for use when encountering the extreme outlier ratios (e.g. more than 90%). Branch-and-Bound (BnB) is able to search in the space of transformation ($SE(3)$) group to globally optimize the objective function, whereas its computational cost grows exponentially with the correspondence number; hence, it may be too slow for large-sized problems.

In addition, the outlier removal methods GORE [11], [38] as well as the non-minimal solvers FGR [34], QUASAR [49] and TEASER/TEASER++ [48], [50] have been proposed to deal with the two problems with over 90% outliers. However, for point cloud registration, these solvers can only succeed when the scale between the two point clouds is known and fixed. When the scale varies (e.g. point clouds formed with different sources), none of these methods could tolerate more than 90% outliers, which indeed necessitates a general solver capable of solving not only the known-scale problems but the unknown-scale ones. Besides, for handling rotation search problems with high outlier ratios, though GORE [38] is efficient when the problem size is small, it requires more time to remove outliers as the correspondence number increases. Hence, proposing a high-outlier-ratio-tolerant solver that can be efficient regardless of the correspondence number seemsrather imperative.

**Contributions.** Motivated by these facts, in this paper, we propose a novel highly robust paradigm, named ICOS (Inlier Compatible Structures) for addressing rotation search and both known-scale and unknown-scale point cloud registration with extreme outlier ratios efficiently.

To be specific, ICOS aims to seek the correct inliers through the random sample flow (rather than pruning outliers as done in the previous works [11], [42]), but significantly different from the RANSAC-family (e.g. [14], [18], [28]), the sampling and filtering process of ICOS is built upon and also facilitated by the constraints derived from the compatible structures that we define based on the theory of invariance.

First, we introduce and construct a series of compatible structures for the rotation search and generalized point cloud registration problems, with which we are able to ‘customize’ special invariant-based constraints that can be used for effectively differentiating inliers from outliers with simple Boolean conditions (Section IV).

Moreover, we design three sampling-based frameworks that can fit with the compatible structures to seek and preserve eligible inliers out of the putative correspondences for the two problems, in which different sampling strategies are employed to make the proposed solvers more time-efficient (Section V).

Finally, we validate ICOS in a variety of experiments over different datasets and demonstrate that: (i) ICOS is precise and robust even when over 95% of the correspondences are outliers in the rotation search, and both the known-scale and unknown-scale registration problems (robust against up to 99% outliers), (ii) ICOS runs fast, yields accurate estimates and recalls approximately 100% of the inliers, and (iii) it is also applicable to practical application problems over real datasets (Section VI).

II. RELATED WORK

In this section, we provide some reviews on: (i) methods for naive and robust rotation search, (ii) methods for naive and robust point cloud registration, (iii) classic robust heuristics for outlier rejection.

**Rotation Search.** The rotation search (‘Wahba’) problem was first proposed by Wahba in [46], in order to estimate the attitude of satellite from the observation of vectors. And it can be solved in closed form based on either rotation matrix [2], [19], [25], [31] or the quaternion-based parametrization [24], [32], although they usually fail in the existence of outliers.

It is known to all that in the process of building the vector correspondences using the feature descriptors [5], [41] based
on 2D or 3D keypoints, it is likely to obtain mismatches, that is, the so-called outliers. Thus, robust solvers are required.

The BnB framework is effective in finding the optimal rotation from the outlier-corrupted corresponding vectors, and the typical examples include the first globally optimal method searching over the rotation space \cite{22}, the L2-distance-based approach \cite{7}, and the speeded angular-distance-based solution \cite{39}. In addition to BnB, guaranteed outlier removal \cite{38} is another way to estimate the rotation robustly, and is also much faster than the BnB solvers. Some other robust rotation search solvers are also the subroutines for point cloud registration problems (e.g. \cite{11}, \cite{50}, \cite{54}), which will be later discussed in the review on point cloud registration.

Point Cloud Registration. To solve the point cloud registration problem, Iterative Closest Point (ICP) \cite{8} is a common local-minimum approach, but it is too dependent on the initial guess of the transformation, thus prone to fail without a good initialization. Therefore, an initialization-free way to address the registration problem—to build correspondences between point clouds from 3D keypoints \cite{44} using feature descriptors (e.g. FPFH \cite{41}, ISS \cite{53} and FCGF \cite{13}), has grown increasingly popular. Many correspondence-based methods have been proposed to address the registration problem in the presence of noise. For example, it can be solved using closed-form solutions \cite{2}, \cite{24} by decoupling the estimation of rotation, scale and translation. Besides, Branch-and-Bound (BnB) has also been applied to solve the registration problem \cite{35} in a globally optimal way. More recently, the Lagrangian Dual Relaxation is developed to produce a globally optimal generalized registration solver \cite{10} for handling point-to-point, point-to-line and point-to-plane registration with optimality guarantees.

However, the 3D feature descriptors generally have lower accuracy and robustness compared with 2D ones \cite{5}, \cite{30}, which could result in relatively high or even extreme outlier ratios in some real-world scenes, as has been discussed in \cite{11}. And the above-mentioned solutions are fairly sensitive to wrong correspondences and will be fragile once there exist outliers among the correspondences, which consequently demands the following robust registration methods.

The minimal closed-form solutions \cite{2}, \cite{24} can be fit into the RANSAC-family frameworks \cite{14}, \cite{18}, \cite{28} to deal with outliers, though may be slow when the outlier ratio is high. Another idea to become robust is that: if outliers can be guaranteed to be removed or eliminated as many as possible, then what still remain should be mostly inliers. GORE \cite{11} is such a preprocessing outlier removal approach that guarantees that the correspondences removed must not be inliers. In terms of the non-minimal robust solvers, FGR \cite{54} is the first method using Graduated Non-Convexity (GNC) to reject outliers, which estimates the rigid transformation matrix with a Gauss-Newton method. Though not globally optimal, FGR is efficient for known-scale registration. Currently, TEASER/TEASER++ \cite{48}, \cite{50} are the more robust non-minimal certifiably globally optimal solvers that can dispose of 99% outliers in the known-scale situation. Another recent solution \cite{29} utilizes RANSAC for scale and translation estimation, and also adopt the GNC framework to solve the rotation. Unfortunately, all these solvers are either incapable of estimating the scale or unable to tolerate over 90% outliers for unknown-scale point cloud registration (as will be shown in Fig. 6 of Section VI).

Robust Heuristics. Apart from the classic hypothesize-and-test robust heuristic RANSAC that runs with minimal solvers to obtain the best consensus set through random sampling, M-estimation is useful for outlier rejection \cite{26}, \cite{33}, \cite{35}, \cite{54}. When the naive solver is non-minimal, the special M-estimator GNC \cite{47} is a general-purpose heuristic operated by alternating optimization and adopts robust cost functions like Geman-McClure (GM) or Truncated Least Squares (TLS), which can typically handle 70-80% outliers in many robotic or computer vision problems. In addition, invariants or graph theory can be also applied for robust estimation. The methodologies mainly include: (i) to first prune outliers and then hand the rest of the correspondences over to other heuristics or solvers (\cite{42}, \cite{48}, \cite{50}), and (ii) to first seek a portion of inliers, make an estimate, and then use the optimal solution to find the complete inlier set (our previous work \cite{43}).

III. NOTATION AND PRELIMINARIES
A. Invariants

We first define the concept of invariants in the two problems by adopting the notation and definition in \cite{42}.

Definition 1 (Invariants): In one geometric problem, assume we obtain \(N\) putative correspondences whose indices are put in set \(\mathcal{C} = \{1, 2, \ldots, N\}\). Let \(\mathcal{S} \subset \mathcal{C}\) denote a subset of \(\mathcal{C}\) with a fixed size and \(\bar{x}\) denote the vector stacking all variables. Then let \(y_S\) represent the measurements w.r.t. subset \(\mathcal{S}\), \(\eta_S\) represent the corresponding noise measurements which are independent of \(x\), and \(h_S(x, \eta_S)\) represent the corresponding model that is used to express the relation between \(x\) and \(y_S\), given by \(y_S = h_S(x, \eta_S)\).

If we find a function \(f(y_S)\) where \(f(y_S) = f(h_S(x, \eta_S))\) is always satisfied and also independent of \(x\) no matter which \(\mathcal{S} \subset \mathcal{C}\) is selected, we can consider \(f\) as an invariant function, also abbreviated as an invariant.

B. Naive Solvers

In our solver ICOS, the computation of the scale, rotation and translation without outliers is realized by introducing the fast non-minimal naive solvers below.

Assume we have \(M\) correspondences (inliers), we first need to acquire corresponding 3D vectors, which, in rotation search, are just the original vector correspondences themselves such that

\[
\tilde{m}_i = u_i, \quad \tilde{n}_i = v_i,
\]

while in point cloud registration, can be represented with the centroid-based translation-free technique \cite{2} such that

\[
\tilde{m}_i = P_i - \bar{P}, \quad \tilde{n}_i = Q_i - \bar{Q},
\]

where \(\bar{P}\) and \(\bar{Q}\) are the centroids in the two frames given by

\[
\bar{P} = \frac{1}{M} \sum_{i=1}^{M} P_i, \quad \bar{Q} = \frac{1}{M} \sum_{i=1}^{M} Q_i.
\]
Finding adequate 3-COS

Finding adequate 4-COS

Fig. 2. Overview of the inlier searching process of ICOS. We start by finding eligible α-COS across the two point sets, then seek new correspondences that can establish β-COS with the existing α-COS, and eventually stop sampling when we collect a sufficient number of eligible β-COS.

where \( I_{12}^1 \) represents the length-based invariant w.r.t. this pair of correspondences, and \( \epsilon_{12} \) denotes the noise measurement (perturbation) on this invariant.

Intuitively, invariant \( I_{12}^1 \) represents the length difference between the third sides of the two triangles formed by the two vector correspondence pairs over the two frames. Without noise perturbation, \( I_{12}^1 \) should be strictly equal to 0 as long as this pair of correspondences are both inliers. But with noise, we can also impose a constraint on \( I_{12}^1 \) to examine whether this correspondence pair contain outliers.

**Proposition I (Constraints on \( I_{12}^1 \))**: For the pairwise invariant \( I_{ij}^1 \) in the 2-COS defined above, we can derive that

\[
I_{12}^1 \leq L,
\]

where \( L \) denotes the bound w.r.t. noise satisfying the inequality such that \( L^* + \|\epsilon\| \cdot (\|\frac{1}{v_1}\| + \|\frac{1}{v_2}\|) \leq L \), in which \( L^* = \|\frac{u_1}{|v_1|} - \frac{u_2}{|v_2|} - \frac{u_1}{|u_1|} + \frac{u_2}{|u_2|}\| \).

**Proof I**: Based on triangular inequality and the properties of \( R \in SO(3) \), we can always have:

\[
\begin{align*}
| |u_1|^2 - |u_2|^2| - |v_1|^2 - |v_2|^2| & = | |u_1|^2 - |u_2|^2| - |v_1|^2 - |v_2|^2| + |u_1|^2 - |u_2|^2| \\
n & \leq L^* + \|R^T\| \cdot \|\frac{\epsilon_1}{|v_1|} - \frac{\epsilon_2}{|v_2|}\| \\
n & \leq L^* + \|\frac{\epsilon_1}{|v_1|}\| + \|\frac{\epsilon_2}{|v_2|}\| \leq L,
\end{align*}
\]

where the inequality \([13]\) can be proved.

If we have obtained an eligible 2-COS, we can compute the rotation w.r.t. this pair of correspondences using the Horn’s triad-based closed-form transformation \([24]\) such that

\[
\begin{align*}
O_1 & = \begin{bmatrix} u_1 & (u_1 \times v_1) \end{bmatrix}, \\
O_2 & = \begin{bmatrix} u_2 & (u_2 \times v_2) \end{bmatrix}, \\
R_{12} & = O_2 O_1^T = R_{9} \exp(\rho_{12}),
\end{align*}
\]
where \( \text{Exp} \) is the exponential map \([4]\) to define the effect of noise on this rotation, which can be represented as \( \text{Exp}(\rho_{ij}) \) and \( \rho_{ij} \) denotes the noise measurement, and \( R_{gt} \) means the ground-truth rotation. Then, we can further try to add a new correspondence to this 2-COS for building a 3-COS, based on which new constraints can be imposed.

**Proposition 2 (3-COS, Invariants and Constraints):** Assume that now we obtain a new correspondence, say the 3\(_{rd} \) correspondence, in addition to the 2-COS we have, as shown in Fig.\([2]\) we are able to construct a 3-COS with the 3\(_{rd} \) invariant such that

\[
I_{13}^{\text{th}} = ||u_i^* - u_3^*|| - ||v_i^* - v_3^*|| = \epsilon_{13},
\]

and two additional length-based constraints are satisfied:

\[
I_{13}^{\text{th}} \leq L, \tag{17}
\]

where \( i \in \{1,2\} \). Apart from simply constraining the length-based rigidity, we also supplement a residual-based constraint on \((u_3,v_3)\) such that

\[
||R_{12}u_3 - v_3|| \leq G, \tag{18}
\]

where \( G \) is the bound for the residual error w.r.t. the 3\(_{rd} \) correspondence. Besides, for every single 2-COS in this 3-COS, we can compute a raw rotation, represented as \( R_{ij} \) with the correspondence tuple \((i,j) \in \{(1,2),(2,3),(1,3)\}\). In this fashion, we are able to impose new constraints on the mutual compatibility between the raw rotations w.r.t. different correspondence tuples selected from this 3-COS such that

\[
\angle(R_{ij}, R_{ab}) \leq F, \tag{19}
\]

where \( \angle \) is the geodesic distance (error) \([21]\) between two rotations, written as

\[
\angle(R_{ij}, R_{ab}) = \left| \arccos\left( \frac{\text{tr}\left(R_{ij}^T R_{ab}\right) - 1}{2} \right) \right|, \tag{20}
\]

where \( F \) is the bound of noise for the geodesic error with \( 2\angle(\text{Exp}(\rho), I_3) \leq F \), and \( I_3 \) means the \( 3 \times 3 \) identity matrix.

**Proof 2:** The proof of constraints \((17)\) is fairly apparent, thus omitted here. For constraints \((19)\), on the basis of the triangular inequality of the geodesic distance in \([21]\), similar to \([42]\), we can derive:

\[
\angle(R_{ij}, R_{ab}) = \angle(R_{ij}, R_{ab}) \leq \left| \arccos\left( \frac{\text{tr}\left(\text{Exp}(\rho_{ij})^T R_{ij} R_{ab} \text{Exp}(\rho_{ab})\right) - 1}{2} \right) \right| = \left| \arccos\left( \frac{\text{tr}\left(\text{Exp}(\rho_{ij})^T R_{ij} R_{ab} \text{Exp}(\rho_{ab})\right) - 1}{2} \right) \right| \tag{21}
\]

\[
\leq \left| \arccos\left( \frac{\text{tr}\left(\text{Exp}(\rho_{ij})^T R_{ij} R_{ab} \text{Exp}(\rho_{ab})\right) - 1}{2} \right) \right| \leq \angle(\text{Exp}(\rho_{ij}), I_3) + \angle(\text{Exp}(\rho_{ab}), I_3) \leq F.
\]

As a result, in rotation search, the constraints over the 2-COS and 3-COS have been established, which will be later transformed as a series of Boolean conditions for the sampling and filtering frameworks of ICOS in Section \([V]\).

### B. COS in Point Cloud Registration

The basic structure in point cloud registration starts from the COS with 3 correspondences.

**Definition 3 (3-COS and Pairwise Scale-based Invariants):** Assume now we obtain 3 non-collinear point correspondences \((P_i, Q_i), i = 1, 2, 3\), defined as one 3-COS. We can derive the following pairwise scale-based invariants:

\[
I_{ij}^s = \frac{||Q_i - Q_j||}{||P_i - P_j||} = s_{gt} + \epsilon_{ij}, \tag{22}
\]

where \((i,j) \in \{(1,2), (1,3), (2,3)\}\), \(I_{ij}^s\) represents the scale-based invariant w.r.t. the \(i_{th}\) and \(j_{th}\) correspondences, and \(s_{gt}\) denotes the ground-truth scale, and \(\epsilon_{ij}\) denotes the noise measurement on the scale.

Then we can further endow any pair of scale-based invariants \(I_{ij}^s\) with the constraint given as follows.

**Proposition 3 (Constraints on \(I_{ij}^s\)):** For any two invariants \(I_{ij}^s\) in the 3-COS above, we can derive that

\[
|I_{ij}^s - I_{jk}^s| \leq A \left( \frac{1}{||P_i - P_j||} + \frac{1}{||P_j - P_k||} \right), \tag{23}
\]

where \((i,j,k) \in \{(1,2,3), (2,3,1), (3,1,2)\}\) (here \(I_{ij}^s\) and \(I_{ji}^s\) denote the same invariant), and \(A\) is the noise bound, satisfying that \(2|\epsilon| \leq A\). In addition, when the scale is known \((s_{gt} = 1)\), we can further have that

\[
|I_{ij}^s - 1| \leq A \left( \frac{1}{||P_i - P_j||} \right). \tag{24}
\]

**Proof 3:** It is easy to have that

\[
s_{gt} R_{gt} P_{ij} + \epsilon_{ij} = Q_{ij}, \Rightarrow ||s_{gt} R_{gt} P_{ij}|| = ||Q_{ij} - \epsilon_{ij}||, \tag{25}
\]

where \(P_{ij} = P_i - P_j, Q_{ij} = Q_i - Q_j,\) and \(\epsilon_{ij} = \epsilon_i - \epsilon_j\). And using the triangular inequality, we can derive that

\[
||Q_{ij}|| - ||\epsilon_{ij}|| \leq ||s_{gt} R_{gt} P_{ij}|| = s_{gt} ||P_{ij}|| \leq ||Q_{ij}|| + ||\epsilon_{ij}||, \tag{26}
\]

so that

\[
s_{gt} - \frac{||\epsilon_{ij}||}{||P_{ij}||} \leq \frac{||Q_{ij}||}{||P_{ij}||} \leq s_{gt} + \frac{||\epsilon_{ij}||}{||P_{ij}||}. \tag{27}
\]

Thus, inequalities \((23)\) and \((24)\) can be derived as long as we have: \(||\epsilon_{ij}|| \leq ||\epsilon_i|| + ||\epsilon_j|| = 2|\epsilon| \leq A\).

Subsequently, we roughly calculate the raw scale \(s_{123}\) as

\[
s_{123} = \frac{1}{\sum \omega_i} \sum \omega_i \frac{||Q_{ij}||}{||P_{ij}||}, \tag{28}
\]

and rotation \(R_{123}\) that aligns the 3-COS w.r.t. the two point sets using Horn’s 3-point minimal solver \([24]\) similar to \([15]\).

Then we define new invariants based on the translation.

**Definition 4 (Translation-based Invariants):** In addition to \((22)\), we can derive the translation-based invariants:

\[
I_t^i = Q_i - s_{123} R_{123} P_i = t_{gt} + \zeta_i, \tag{29}
\]

where \(i \in \{1, 2, 3\}\). \(I_t^i\) represents the translation-based invariant w.r.t. the \(i_{th}\) correspondences, and \(\zeta_i\) describes its noise measurement. In known-scale cases, we directly set \(s_{123} = 1\).
Algorithm 1: ICOS for Rotation Search

Input: correspondences $\mathcal{N} = \{(u_i, v_i)\}_{i=1}^N$; minimum number of the inlier set size $X$;
Output: optimal $\hat{R}$; inlier set $\mathcal{N}^{inn}$;

while true do
\begin{enumerate}
\item $\mathcal{N}^{inn} \leftarrow \emptyset$, pass $\leftarrow 0$;
\item for $itr_1 = 1$: maxIt$_{itr_1}$ do
\begin{enumerate}
\item Randomly select $\mathcal{R} = \{(u_i, v_i)\}_{i=1}^N \subset \mathcal{N}$;
\item if $\mathcal{R}$ satisfies constraint (13) then
\begin{enumerate}
\item Solve $R_{12}$ with $\mathcal{R}$ using (15);
\item pass $\leftarrow 1$;
\end{enumerate}
\item break
\end{enumerate}
end
\item if pass $= 1$ then
\begin{enumerate}
\item for $itr_3 = 1$: maxIt$_{itr_3}$ do
\begin{enumerate}
\item checkSampling;
\item Randomly select $(u_k, v_k) \in \mathcal{N}$ and construct a 3-COS with $\mathcal{R}$;
\item if this 3-COS satisfies (17) then
\begin{enumerate}
\item $\text{count} \leftarrow \text{count} + 1$;
\item $\mathcal{N}^{inn} \leftarrow \mathcal{N}^{inn} \cup \{(u_k, v_k)\}$;
\end{enumerate}
\item end
\item if $\text{count} \geq X$ then
\begin{enumerate}
\item $\mathcal{N}^{inn} \leftarrow \mathcal{N}^{inn} \cup \mathcal{R}$;
\item break
\end{enumerate}
\end{enumerate}
end
\end{enumerate}
end
\item Solve $\hat{R}$ with $\mathcal{N}^{inn}$ using (8);
\item Compute residuals $r_i$ ($i = 1, 2, \ldots, N$) for all the correspondences in $\mathcal{N}^{inn}$; if $r_i \leq 5.2\sigma$, add it to $\mathcal{N}^{inn}$;
\item Solve optimal $\hat{R}$ with $\mathcal{N}^{inn}$ using (8);
\item return optimal $\hat{R}$ and the inlier set $\mathcal{N}^{inn}$;
\end{enumerate}

Proposition 4 (Constraints on $I_4^s$): For invariants $I_4^s$ in the 3-COS above, we can derive that
\begin{equation}
\|I_4^t - I_4^s\| \leq B \tag{30}
\end{equation}
where $(i, j) \in \{(1, 2), (2, 3), (3, 1)\}$, and $B$ is the noise bound, satisfying $2\|\xi\| \leq B$.

Proof 4: The proof here is trivial.

Up to now, we have explored all the possible invariants as well as their constraints according to the rigid compatibility of the 3-COS. On the basis of this 3-COS, whenever we obtain a new correspondence as the $4_{th}$ node, defined as a 4-COS, additional constraints can be further imposed as follows.

Proposition 5 (4-COS, Invariants and Constraints): If we have the $4_{th}$ correspondence in addition to the 3-COS we have, we can build 3 more scale-based invariants such that
\begin{equation}
I_{44}^s = \frac{\|Q_4 - Q_4\|}{\|P_4 - P_4\|} = s_{gt} + \epsilon_{i4}, \tag{31}
\end{equation}
where $i \in \{1, 2, 3\}$, with the following constraints:
\begin{equation}
|I_{44}^s - I_{44}^t| \leq A \left(\frac{1}{\|P_4 - P_4\|} + \frac{1}{\|P_4 - P_4\|}\right), \tag{32}
\end{equation}
where $i, j \in \{1, 2, 3\}, i \neq j$. Similar to Proposition 2, we also supplement a residual-based constraint on $(P_4, Q_4)$ such that
\begin{equation}
\|s_{123}R_{123}P_4 + t_{123} - Q_4\| \leq C, \tag{33}
\end{equation}
where $t_{123}$ can be computed from the 3-COS as
\begin{equation}
t_{123} = \frac{I_4^t + I_4^t + I_4^t}{3}, \tag{34}
\end{equation}
and $C$ is the bound for this residual error. For the known-scale cases, $I_{123}^t = 1$ is adopted. Moreover, for every single 3-COS in this 4-COS, we can compute a raw rotation, represented as $R_{ijk}$ with the correspondence tuple $(i, j, k)$. Thus, we can establish new constraints on the mutual compatibility of the different rotations as in Proposition 2 such that
\begin{equation}
\angle (R_{ijk}, R_{abc}) \leq D, \tag{35}
\end{equation}
where `$\angle$' is also the geodesic error between two rotations, expressed as
\begin{equation}
\angle (R_{ijk}, R_{abc}) = \arccos \left(\frac{\text{tr} (R_{ijk} R_{abc}) - 1}{2}\right), \tag{36}
\end{equation}
$(i, j, k), (a, b, c) \in \{(1, 2, 3), (1, 2, 4), (1, 3, 4), (2, 3, 4)\}$, and $D$ is the noise bound satisfying $2\angle(\text{Exp}(\rho), I_3) \leq D$.

Proof 5: The proof is omitted here, because the derivations of constraints (34) is fairly apparent based on the that of Proposition 2 and 4 while for the derivation of constraint (35), readers can refer to Proposition 2.

So far, we have derived the invariant-based constraints from both the 3-COS and 4-COS for the point cloud registration problem, getting us prepared for the resulting robust solver.

V. Our Solver: ICOS

In this section, we present our solver ICOS for rotation search and both the known-scale and the unknown-scale point cloud registration problems.

We first provide the insights and the complete overview of the proposed solver. As intuitively illustrated in Fig. 2, we begin with finding eligible $a$-COS across the two point sets by roughly checking the first layer of invariant-based constraints, where $a = 2$ for rotation search and $a = 3$ for point cloud registration, as discussed in Section IV.

Once a $a$-COS satisfies the required constraints, we solve the raw solutions (rotation, scale or translation) with it, which is for further examining whether the second layer of constraints can be satisfied. Holding one eligible $a$-COS, we then pick random correspondences one after another with testing both their residual-based and invariant-based constraints simultaneously, in order to construct an eligible $b$-COS, where $b = a + 1$, on the basis of this $a$-COS.

The process of finding $b$-COS continues until we are able to obtain a sufficient number of such $b$-COS within reasonable times of the single-correspondence random sampling, which also serves as a hidden condition aiming to determine whether the $a$-COS and its corresponding $b$-COS can be all true inliers.

The following subsections provide more details on formulating the pseudocode of ICOS for the specific problems.
Algorithm 2: ICOS for Known-scale Registration

Input: correspondences $\mathcal{N} = \{(P_i, Q_i)\}^n_{i=1}$; minimum number of the inlier set size $X$;
Output: optimal $(\hat{R}, \hat{t})$; inlier set $\mathcal{N}^{ln}$;

1. while true do
   2. $\mathcal{N}^{ln} \leftarrow \emptyset$;
   3. for $itr = 1 : maxItr$ do
      4. Randomly select $\mathcal{R} = \{(P_i, Q_i)\}^n_{i=1} \subset \mathcal{N}$;
      5. if $\mathcal{R}$ satisfies constraint (24) then
         6. for $itr' = 1 : maxItr_2$ do
            7. Randomly select $(P_3, Q_3) \in \mathcal{N}$ and construct a 3-COS set $\mathcal{O}$ with $\mathcal{R}$;
            8. if $\mathcal{O}$ satisfies constraints (24) then
               9. break
            end
         end
      end
   end
   10. if $\mathcal{O}$ satisfies constraints (23, 24) then
      11. Solve $s_{123}, R_{123}$ with $\mathcal{O}$ using (28) and (15);
      12. if $\mathcal{O}$ satisfies constraints (30) then
         13. for $itr_3 = 1 : maxItr_3$ do
            14. Randomly select $(P_k, Q_k) \in \mathcal{N}$ and construct a 4-COS with $\mathcal{O}$;
            15. if this 4-COS satisfies (32, 35) then
               16. count $\leftarrow$ count $+ 1$;
               17. $\mathcal{N}^{ln} \leftarrow \mathcal{N}^{ln} \cup \{(P_k, Q_k)\}$;
            end
         end
      end
   end
   17. Solve $\mathcal{R}$ with set $\mathcal{N}^{ln}$ using (8) and (10);
   18. Compute residuals $r_i$ ($i = 1, 2, \ldots, N$) for all the correspondences in $\mathcal{N}$; if $r_i \leq 5.2\sigma$, add it to $\mathcal{N}^{ln}$;
   19. Solve optimal $(\hat{R}, \hat{t})$ with $\mathcal{N}^{ln}$ using (8) and (10);
   20. return optimal $(\hat{R}, \hat{t})$ and the inlier set $\mathcal{N}^{ln}$;

A. ICOS for Rotation Search

For addressing the rotation search problem, we render the pseudocode of ICOS in Algorithm 2 as well as the description for each part as follows.

Lines 2-10: ICOS initiates with random sampling. But different from ordinary random sampling procedures, we impose constraints on each random sample selected. When picking one random subset $\mathcal{R}$ of 2 correspondences, we immediately check the length-based invariant [13]. If it can be satisfied, we solve the raw rotation with this 2-COS $\mathcal{R}$ and then allow it to proceed to the next step, which is operated by setting $pass = 1$; if not, we start over from the sampling process.

Lines 11-25: If we manage to obtain an eligible 2-COS, say $\mathcal{R}$, we can then seek new correspondences to build a 3-COS, also by constrained sampling, where we select and test the correspondence samples one at a time. If one correspondence can satisfy constraints [17, 19], which indicates it can form an eligible 3-COS with $\mathcal{R}$, we add it to the inlier set and make $count$ increase by 1. Once we collect enough correspondences ($count \geq X$), we stop sampling and return the $X+2$ inliers.

During this process, we supplement a subroutine: checkSampling, in order to boost time-efficiency, which is designed to timely break from the sampling if the obtained 2-COS $\mathcal{R}$ does not seem to be inliers, and this operation could be realized by regularly checking the number of 4-COS formed according to the iteration number, as demonstrated in Algorithm 2. To be specific, as long as the correspondences in 2-COS $\mathcal{R}$ are both inliers, at least one inlier correspondence ought to be sampled after a sufficient number of iterations, which can be computed from [37] by setting a high confidence; conversely, if the latter condition cannot be satisfied, we have every reason to assume that set $\mathcal{R}$ contains at least one outlier.

Lines 26-28: We first compute the rotation with the $X+2$ inliers and consider all correspondences whose residual errors satisfy $r_i \leq 5.2\sigma$ as the ultimate inliers. Finally, we are able to solve the optimal rotation with all the inliers found among the
Algorithm 4: checkSampling (Subroutine of ICOS)

Input: iteration number \(itr_3\), and count;
Output: break or do nothing;
\[
\begin{align*}
1 & \text{ if } itr_3 \geq maxItr_4 \text{ and count} < 1 \text{ then} \\
2 & \quad \text{ break} \\
3 & \text{ end} \\
4 & \text{ if } itr_3 \geq 2maxItr_4 \text{ and count} < 2 \text{ then} \\
5 & \quad \text{ break} \\
6 & \text{ end} \\
7 & \text{ if } itr_3 \geq 3maxItr_4 \text{ and count} < 3 \text{ then} \\
8 & \quad \text{ break} \\
9 & \text{ end}
\end{align*}
\]

B. ICOS for Known-scale Registration

The pseudocode of ICOS for point cloud registration with known scale \((s = 1)\) is given in Algorithm 2.

Lines 2-14: We design a decoupled sampling method for the known-scale registration. First, we select a random subset \(\mathcal{R}\) of 2 correspondences, rather than 3, and check if their scale-based invariant \(I_{23}\) is close enough to 1 using constraint (24). If it is, we can then proceed to select random samples for the remaining one correspondence, with which we try to form an eligible 3-COS that can satisfy constraints (24) based on \(\mathcal{R}\); if not, we go back to the sampling process for seeking \(\mathcal{R}\).

Lines 15-32: Once a qualified 3-COS (set \(\mathcal{O}\)) is obtained, we check their constraints (23). If satisfied, we then solve the raw rotation with \(\mathcal{O}\), and determine whether constraints (30) are fulfilled in \(\mathcal{O}\). If yes, we then move on to sampling once again, for building one 4-COS based on set \(\mathcal{O}\). We take random correspondences one at a time, and feed them sequentially to \(\mathcal{O}\) to check constraints (32-35). After collecting enough correspondences, say \(X\), which can construct qualified 4-COS with \(\mathcal{O}\), we jump out and return the inliers.

Lines 33-35: Similar to Algorithm 1, we check the residual error w.r.t. each correspondence to gather the entire inlier set, which is then used to compute optimal solutions with the non-minimal solvers [9], [8] and [10].

C. ICOS for Unknown-scale Registration

When the scale is unknown, we design a different framework for ICOS, and its pseudocode is shown in Algorithm 3. Algorithm 3 employs the methodology similar to lines 15-35 of Algorithm 2. The differences lie in that: (i) we are unable to impose constraints (24) on the 3-COS using the decoupled sampling method as operated in lines 3-14, since the scale is not fixed, which may result in more runtime than the known-scale problems generally, and (ii) the raw scale and the scale-based invariants are not set to 1, but need to be computed as in (22) and (31) instead.

D. Discussion on the Maximum Iteration Numbers

Note that in Algorithm 1, a series of maximum iteration numbers are involved during the sampling process, so now we discuss their proper choice of values.

E. Discussion on the Performance of ICOS

Since the main body of our solver ICOS consists in simply judging a series of Boolean conditions (true or false) according to the constraints over the invariants w.r.t. different variables, it can be fast to implement. Besides, ICOS abandons the strategy of making an estimate for every random sample selected (as operated in RANSAC), and smartly solves the transformation only when the scale-based constraints (22, 24) allow it to. Also, the residual errors w.r.t. all the correspondences only require to be solved once (just for finding the whole inlier set), saving much time for realistic implementation, especially when the point cloud correspondences are in huge numbers, which is not uncommon in practice.

VI. EXPERIMENTS

In this section, we conduct a series of experiments based on multiple datasets to evaluate the proposed solver ICOS for point cloud registration, also in comparison with the existing state-of-the-art solvers. All the experiments are implemented in Matlab on a laptop with an i7-7700HQ CPU and 16GB RAM, and no parallelism programming is ever used. And the explicit parameter setup for ICOS is given in Table I.

A. Benchmarking of Rotation Search on Synthetic Data

Experimental Setup. We test our solver ICOS for rotation search over synthetic experiments. We first randomly generate \(N = \{100, 500, 1000\}\) vectors \(U = \{u_i\}^N_{i=1}\) with unit-norm, and rotate \(U\) with a random rotation \(R \in SO(3)\) to obtain a
with constantly demonstrates the lowest rotation errors, and (iv) ICOS is the most accurate solver for all time since it breaks at 98%, while ICOS is robust against 95-96% with FGR and RANSAC fail at 90% and RANSAC is also not adopted due to its fairly long runtime. 

To create outliers in clutter, a portion of the points in the transformed point set, as the point set is the same as (38), where subscripts ‘gt’ indicates the ground-truth value.

For benchmarking, we test our solver ICOS (Algorithm 1) against FGR [54], GORE [38], RANSAC and BnB [39]. In terms of FGR, we merely solve the rotation rather than the whole transformation using the Gauss-Newton method. As for RANSAC, we adopt Horn’s minimal method [24] to solve the rotation with the confidence set to 0.995, where we design two stop conditions: (i) RANSAC with at most 100 iterations, named RANSAC(100), and (ii) RANSAC with at most 1000 iterations, named RANSAC(1000). When $N = 1000$, we exclude BnB as it takes tens of minutes per run. QUASAR [49] is also not adopted due to its fairly long runtime. 

Results. Fig. 3 displays the benchmarking results on both accuracy and runtime. It is not hard to observe that: (i) both FGR and RANSAC(100) fail at 90% and RANSAC(1000) breaks at 98%, while ICOS is robust against 95-96% with $N = 100$, 98% with $N = 500$, and 99% with $N = 1000$, (iii) ICOS is the most accurate solver for all time since it constantly demonstrates the lowest rotation errors, and (iv) with $N = \{500, 1000\}$, ICOS is the fastest solver when the outlier ratio is below 95%, and with $N = 100$, it has similar speed as GORE when the outlier ratio is no greater than 60%.

B. Benchmarking of Point Cloud Registration on Real Data

Experimental Setup. We then evaluate ICOS against other state-of-the-art methods over the ‘bunny’ and the ‘Armadillo’ point clouds from the Stanford 3D Scanning Repository [15]. We first randomly downsample the point cloud to 1000 points and resize it to be placed in a cube of $[-0.5, 0.5]^3$ meters, which can be regarded as the initial point set $P = \{P_i\}_{i=1}^{1000}$. Then, we generate a random rigid transformation: $(s, R, t)$, in which scale $s \in [1, 5]$ meters, rotation $R \in SO(3)$ and translation $||t|| \leq 3$ meters, to transform point set $P$. We add random Gaussian noise with $\sigma = 0.01$ and $\mu = 0$ to the transformed point set, as the point set $Q = \{Q_i\}_{i=1}^{1000}$. To create outliers in clutter, a portion of the points in $Q$ are substituted by random points inside a sphere of diameter $\sqrt{3}$ meters according to the outlier ratio (from 0% up to 99%), as exemplified in Fig. 4. All the experimental results are obtained over 50 Monte Carlo runs in the same environment.

The estimation errors in all experiments are computed as:

$$E_s = ||\hat{s} - s_{gt}||,$$

$$E_t = ||\hat{t} - t_{gt}||,$$

and $E_R$ is the same as (38), where subscripts ‘gt’ all indicate the ground-truth.

Known-scale problems: We adopt FGR, GORE, RANSAC, TEASER and ICOS for the known-scale point cloud registration problems. As for RANSAC, we use Horn’s minimal (3-point) method [24] to solve the rigid transformation and the confidence is set to 0.995. There are two stop conditions for RANSAC: (i) RANSAC with at most 1000 iterations, named
Fig. 4. Benchmarking of point cloud registration over the ‘bunny’ [15]. Left-top: Examples of a known-scale and an unknown-scale point cloud registration problem with 98% outliers. (a) Known-scale registration results w.r.t. increasing outlier ratio (0-99%). (b) Unknown-scale registration results w.r.t. increasing outlier ratio (0-99%).

Fig. 5. Inlier recall ratio of ICOS in rotation search, and both known-scale and unknown-scale point cloud registration w.r.t. increasing outlier ratio (0-99%).

Fig. 6. Benchmarking of point cloud registration on scale estimation. Left: Scale errors w.r.t. increasing outlier ratio (0-99%). Right: Runtime w.r.t. increasing outlier ratio (0-99%).

RANSAC(1000), and (ii) RANSAC with at most 1 minute of runtime, named RANSAC(1min). In terms of TEASER, we adopt the GNC heuristic version as in [50], which uses GNC-TLS to solve the large robust SDP problem, rather than using convex relaxation, but differently: (i) we do not apply parallelism programming in order to ensure the fairness in runtime evaluation for all the tested solvers, and (ii) TEASER is completely implemented in Matlab, not in C++, using the fast maximum clique solver [17].

Unknown-scale problems: For unknown-scale registration, we only compare ICOS with RANSAC because: (i) FGR and GORE are not designed for scale estimation originally, and (ii) the scale estimator Adaptive Voting in TEASER is too slow for handling 1000 correspondences (requiring tens of minutes per run) and is also unable to tolerate over 90% outliers (as will be shown in Fig. 6), thus no need to be compared.

Results. The main benchmarking results are shown in Fig. 4 (over the ‘bunny’ point cloud) and Fig. 7 (over the ‘Armadillo’ point cloud). We can observe that: (i) in the known-scale registration problem, though GORE, TEASER and our ICOS are all robust against 99% outliers, TEASER and ICOS are generally more accurate, (ii) RANSAC(1000) fails at 90%, while FGR and RANSAC(1min) fail at over 95%, not as robust as ICOS, (iii) in the unknown-scale registration problems, ICOS can still address 99% outliers highly robustly, and (iv) ICOS is the fastest solver when the outlier ratio is below 98% (Note that though FGR seems faster at over 96%, it already becomes hardly robust).
C. Evaluation on the Inlier Recall Ratio

We test the capability of ICOS for recalling correct inliers from the outlier-corrupted putative correspondence set in rotation search and known-scale and unknown-scale registration with the same experimental setup as in Section VI-A and VI-B setting $N = 1000$ for all the problems, and the results are shown in Fig. 5. The recall ratio is defined as the ratio of the number of true inliers found by ICOS to the ground-truth number of inliers among the original correspondences. We can observe that ICOS is able to recall approximately 100% (no less than 99%) of the inliers from the correspondences with outlier ratio ranging from 0% to 99%, which underlies its high estimation accuracy against noise in the benchmarking above.

D. Benchmarking on Scale Estimation in Registration

We benchmark ICOS for scale estimation alone, compared with the Adaptive Voting in TEASER/TEASER++ and the 1-point RANSAC in [29]. Following the setup in Section VI-B we further downsample the ‘bunny’ to only 100 points because solver Adaptive Voting will run in tens of minutes with 1000 correspondences. In each run, the scale is randomly generated within $(1, 5)$ meters. Fig. 6 shows the scale errors and the runtime w.r.t. increasing outlier ratio. We can see that both Adaptive Voting and 1-point RANSAC break at 90% outliers, while ICOS can stably tolerate 90% outliers and is faster than the other two competitors. In fact, ICOS can be robust against up to 99% outliers, as has been shown in Section VI-B.

E. Qualitative Registration Results over Real Datasets

In addition to the quantitative point cloud registration results rendered above, we provide qualitative results of the registration problems over multiple real point clouds, including the ‘bunny’ and ‘Armadillo’ from [15], the ‘Mario’ and ‘Squirrel’ from the SHOT dataset [51] and the ‘city’ and ‘castle’ from the RGB-D Scans dataset [51], for concrete evaluation.

These experiments are all set up as follows: (i) we first transform a certain point cloud with a random transformation (with both known scale and unknown scale), (ii) we then adopt the FPFH 3D feature descriptor [40] to match and build putative correspondences between the transformed point cloud (colored in magenta) and the initial one (colored in blue), in which outliers are bound to exist, and (iii) we apply ICOS to solve the transformation and then reproject the transformed point cloud back to its initial place with the transformation solved. The qualitative results are shown in Fig. 8. Note that after reprojection, the better the magenta point cloud overlaps with the blue one, the smaller the registration errors can be.

From Fig. 8 we can see that ICOS yields rather satisfactory robust registration results.
We supplement more experiments on rotation search and unknown-scale point cloud registration with high noise ($\sigma = 0.1$). Since the ‘bunny’ point cloud is placed in a $[-0.5, 0.5]^3$ box, adding a noise with $\sigma = 0.1$ can be extreme, as demonstrated in Fig. 9 where the shape of the ‘bunny’ is hardly seen. We test ICOS against other solvers over the synthetic data with $N = 100$ and the ‘bunny’ with $N = 1000$. The results are shown in Fig. 9 (in unknown-scale registration, the translation errors are similar to rotation errors, thus omitted).

We can observe that ICOS is still robust against at least 80% outliers in both of the tested problems with such high noise while keeping time-efficient, outperforming other competitors.

G. Application 1: Image Stitching

To validate ICOS in the real-world application w.r.t. rotation search, we adopt ICOS to solve the image stitching problem over the LunchRoom image sequence from the PASSTA dataset [34]. We use 2D feature descriptor SURF [5] to detect and match the 2D keypoints across a certain pair of images with overlapping scenes in them. Subsequently, we apply the inverse of the intrinsic matrix $K^{-1}$ of the camera to create the vector correspondences $U = \{u_i\}_{i=1}^N$ and $V = \{v_i\}_{i=1}^N$ over the two different frames. Eventually, we employ Algorithm 1 to solve the rotation $\hat{R}$ aligning the two vector sets (coordinate frames) and then obtain the homography matrix using matrix transformation $H = K\hat{R}K^{-1}$ to stitch the image pair together. The correspondences and experimental results are shown in Fig. 10. We find that ICOS always finds the correct inliers among correspondences and stitches the images successfully.

H. Application 2: 3D Object Localization

We further test ICOS over the RGB-D scenes datasets [27] for dealing with the 3D object localization (pose estimation) problems. We first extract the 3D point cloud of the targeted object (cereal box, cap, and table) from the scene according to the labels provided and impose a random rigid transformation (with both known scale and unknown scale) as well as random noise with $\sigma = 0.01$ and $\mu = 0$ on the target object. Then, we adopt the FPFH 3D feature descriptor [40] to build putative correspondences between the scene and the transformed object. We find that the correspondences constructed by FPFH contain over 80% outliers in all the scenes tested, and what is worse, the outlier ratios even exceed 90% in most scenes.

For each object-scene registration problem, we apply ICOS, RANSAC(1000) and RANSAC(1min) to estimate the transformation (relative pose) of the object w.r.t. the scene, and then reproject the object back to the scene with the transformation estimated by each solver, respectively. The qualitative results are demonstrated in Fig. 11 and 12 and the quantitative results are specified in Table. II and III in which the unit for $E_R$ is degree ($^\circ$), and the unit for $E_t$ is meter, computed as in (39). In each table, the best results are labeled in the **bold** font.
Fig. 10. Image stitching results over the LunchRoom dataset [34]. Column 1 to column 3 display the raw correspondences matched by SURF [5] with the image indices and outlier ratios given on top, the inliers found by ICOS (the inliers found are labeled in green, the others are outliers, labeled in red), and the image stitching results using the rotation estimated by ICOS, respectively.
Fig. 11. Qualitative results of 3D object localization (cereal box) using ICOS, RANSAC(1000) and RANSAC(1min). From left to right, we show the correspondences matched by FPFH (inliers are in green lines and outliers are in red lines), and the registration (reprojection) results with ICOS, RANSAC(1000), and RANSAC(1min), respectively. Note that in all scenes the outlier ratios are over 80%, and our ICOS can always yield very promising registration results.
Fig. 12. Qualitative results of 3D object localization (cap, table) using ICOS, RANSAC(1000) and RANSAC(1min). From left to right, we show the correspondences matched by FPFH (inliers are in green lines and outliers are in red lines), and the registration (reprojection) results with ICOS, RANSAC(1000), and RANSAC(1min), respectively. Note that in all scenes the outlier ratios are over 80%, and our ICOS can always yield very promising registration results.


According to the results, ICOS is capable of estimating the object poses robustly and accurately even with more than 95% outliers among the 619 correspondences (e.g. Scene-09, $s \in (1, 5)$); however, the two RANSAC solvers may fail in nearly a half of the cases. Hence, in addition to its superior performance on point cloud registration problems as shown in standard benchmarking, ICOS also demonstrates strong practicality for tackling real-world problems.

### VII. Conclusion

In this paper, we present a novel, fast and highly robust paradigm, ICOS, applicable to the rotation search and both the known-scale and unknown-scale point cloud registration problems with correspondences. We define special compatible structures for the two problems above to generate invariants with mathematical constraints. Then, we are able to render a sequence of well-designed invariant-constrained random sampling and inlier seeking frameworks for these problems, which can eliminate outliers by means of fast Boolean conditions and search for qualified $\alpha$-COS as well as their corresponding $b$-COS as the inliers, giving birth to our solver ICOS. Generally, ICOS can automatically stop sampling, and eventually return nearly 100% of the inliers out of the putative correspondences.

We benchmark ICOS against other state-of-the-art rotation search and point cloud registration solvers in multiple experiments. We show that: (i) ICOS can be the most state-of-the-art, or at least one of the most stat-of-the-art, solver for the two problems, since it is robust against over 95% outliers (at most up to 99%), has the best estimation accuracy and runs fast in most cases, and (ii) ICOS demonstrates the similar robustness (robust against 99% outliers) and accuracy regardless of the scale situation, suitable for generalized point cloud registration, and (iii) ICOS has proved to be practical and efficient for real-world applications problems including image stitching and 3D object localization.

Future works may include extending the framework of ICOS or its inspirations to other geometric vision or robotics problems, as a general-purpose solver.

### References

1. Alex M Andrew. Multiple view geometry in computer vision. *Kynrens*, 2001.
2. K Somani Arun, Thomas S Huang, and Steven D Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on pattern analysis and machine intelligence*, (5):698–700, 1987.
3. Michel A Audette, Frank P Ferrrie, and Terry M Peters. An algorithmic overview of surface registration techniques for medical imaging. *Medical image analysis*, 4(3):201–217, 2000.
4. Timothy D Barfoot. State estimation for robotics. *Cambridge University Press*, 2017.
5. Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):345–359, 2008.
6. Jean-Charles Buzin, Yongduek Seo, Richard Hartley, and Marc Pollefeys. Globally optimal inlier set maximization with unknown rotation and focal length. In *European Conference on Computer Vision*, pages 803–817. Springer, 2014.
7. Jean-Charles Buzin, Yongduek Seo, and Marc Pollefeys. Globally optimal consensus set maximization through rotation search. In *Asian Conference on Computer Vision*, pages 539–551. Springer, 2012.
8. PJ Besl and Neil D McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
9. Gérard Blais and Martin D. Levine. Registering multiview range data to create 3D computer objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):820–824, 1995.
10. Jesus Briales and Javier Gonzalez-Jimenez. Convex global 3d registration with lagrangian duality. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4960–4969, 2017.
11. Alvaro Parra Bustos and Ta-Jun Chin. Guaranteed outlier removal for point cloud registration with correspondences. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2868–2882, 2017.
