ISPRS Journal of Photogrammetry and Remote Sensing 167 (2020) 363–374

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

GESAC: Robust graph enhanced sample consensus for point cloud registration

Jiayuan Li [a,*], Qingwu Hu [a], Mingyao Ai [a,b]

[a] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China
[b] State Key Laboratory of Information Engineering, Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

ARTICLE INFO

Keywords:
Point cloud registration
Coarse registration
Feature correspondence
RANSAC
Robust cost

ABSTRACT

Pairwise point cloud registration (PCR) is a crucial problem in photogrammetry, which aims to find a rigid transformation that registers a pair of point clouds. Typically, PCR is performed in a coarse-to-fine manner. Coarse registration provides good initial transformations for fine registration, which determines whether the PCR can succeed. RANSAC-based correspondence registration is the most popular technique for coarse registration. However, the outlier rate of feature correspondences extracted from point clouds is generally very high. Current RANSAC-variants require a huge number of trials to achieve satisfactory results at high outlier rates. This paper proposes a fast and robust RANSAC-variant for PCR, called graph enhanced sample consensus (GESAC). GESAC improves classic RANSAC-family in both sampling and model fitting steps. In the sampling, GESAC generates a much larger subset instead of a minimal subset for model fitting. RANSAC-variants treat a subset as a good one only if the correspondences in the subset are all inliers. In contrast to RANSAC-variants, GESAC allows outliers in the subset and only requires three inliers in the case of point cloud coregistration. Hence, the probability to obtain good subsets of GESAC is much larger than the ones of classic RANSAC-variants. GESAC uses an equal-length constraint to filter "degraded" subsets and expresses a subset as a graph. Then, a max-pooling graph matching strategy is applied to remove potential outliers in the subset. In the model fitting, GESAC introduces a shape-annealing robust estimate instead of classic least-squares for rigid transformation estimation. Hence, even if the subset cleaned by graph matching still contains outliers, GESAC is able to recover a correct solution for PCR. Both simulated and real experiments demonstrate the power of GESAC, i.e., it can tolerate up to more than 99% outliers and is 4000 + times faster than RANSAC at outlier rates above 99% (Note that the running time of feature extraction is not included).

1. Introduction

Point cloud registration (PCR) is a very important step in 3D scanning, which has been widely applied in the fields of photogrammetry, surveying and mapping, computer vision, and robotics. For instance, PCR is substantial to generate the 3D panorama of an object scene; it can be used to register a structure-from-motion point cloud and a LiDAR point cloud; it is also a main step in simultaneous localization and mapping (SLAM), which calculates the trajectory of a robot. Due to occlusions and limited field-of-view, a single point cloud captured by a 3D scanner generally covers only a part of the scene. Thus, similar to image-based reconstruction, it is necessary to measure a sequence of 3D scans from different perspectives to cover the whole scene. These individual 3D scans can be merged using pairwise PCR, which finds a rigid transformation that registers a pair of 3D scans into a common reference system (Li et al., 2020).

Currently, one of the most popular strategies for PCR is in a coarse-to-fine manner (Dong et al., 2018). First, a coarse registration method is applied to provide an initial guess of the rigid transformation; then, a fine registration algorithm such as iterative closest point (ICP) proposed by Besl and McKay (1992) refines the initial guess and provides an accurate result. For example, the processing software of the RIEGL VZ-1000[1] uses artificial markers for coarse registration and ICP for fine registration. ICP is a milestone and the de facto standard for PCR. However, it only finds a local minimum of the transformation and hence is very sensitive to the initial guesses. If the initial parameters are not

* Corresponding author.
E-mail address: ljy_whu_2012@whu.edu.cn (J. Li).
[1] http://www.riegl.com/products/.

https://doi.org/10.1016/j.isprsjprs.2020.07.012
Received 15 October 2019; Received in revised form 9 June 2020; Accepted 22 July 2020
Available online 3 August 2020

sufficiently good, ICP may fail to register the point clouds. Therefore, coarse registration determines whether the PCR can succeed. In coarse registration, feature-based methods are flexible and inexpensive. So, this paper only focuses on feature-based coarse registration.

RANSAC-based (Fischler and Bolles, 1981) correspondence registration is the most popular technique in coarse registration (Dong et al., 2018). First, initial feature correspondences are extracted and matched from point clouds based on feature detectors and descriptors. Then, a RANSAC-type algorithm is adapted to simultaneously remove outliers and fit the rigid transformation model. Because of lack of texture, uneven point densities, and noise, 3D feature matching is much less accurate compared to its 2D counterparts like scale-invariant feature transform (SIFT) algorithm (Lowe, 2004) and radiation-variation insensitive feature transform (RIFT) algorithm (Li et al., 2019b), which results in much higher outlier rates. As pointed out by Yang and Carlone (2019), it is common to have 95% of outliers in the initial 3D correspondence set. Current RANSAC-variants require a huge number of trials to achieve correct results at such high outlier rates, which largely decrease their practicability. For example, RANSAC theoretically requires more than 4.6 million trials (see Section 3.2.4 for details) to generate at least one good sample (the confidence is 0.99) at an outlier rate of 99%. Thus, RANSAC variants usually limit the maximum number of trials in realistic applications. However, this will decrease their robustness.

Graph matching can also be used for correspondence-based PCR. Generally, feature points are represented as graphs, and graph matching seeks reliable correspondences by minimizing the distortions of the two graphs. Then, the rigid transformation is estimated based on the reliable correspondences. The major drawback of graph matching lies in its NP-hard nature, since it is essentially a quadratic assignment problem. Although many algorithms have been proposed, the huge computational costs still limit their usages in realistic PCR task.

In this paper, we propose a new RANSAC-variant called graph enhanced sample consensus (GESAC) for PCR, which is fast and robust. GESAC improves both steps of classic RANSAC-type methods, i.e., random sampling and model fitting. In the random sampling, RANSAC-variants use a minimal subset with three feature correspondences for rigid transformation estimation. They treat the sampled subset as a good one only if the three correspondences are all inliers. In contrast, GESAC samples a much larger subset (in our experiments, the size of a subset is 32). It allows outliers in the subset and only requires three inliers in the case of point cloud coregistration. Therefore, the probability to obtain good subsets of GESAC is much larger than the ones of classic RANSAC-variants. In theory, GESAC only requires 1500 trials (see Section 3.2.4 for details) to generate at least one good subset sample (the confidence is 0.99) at an outlier rate of 99%, which is only 1/4000 of the one of RANSAC. Specifically, we observe that the two lines of a correct line correspondence detected from light detection and ranging (LiDAR) point clouds should have approximately the same length, since the length of a line is invariant to rotations and translations. Each pair of point correspondences can construct a line correspondence. Thus, if both the point correspondences are correct, the lengths of the two constructed lines are approximately the same. We regard this as an equal-length constraint and use it to filter "degraded" subsets. If a subset contains more than two correspondences that satisfy the equal-length constraint, it is regarded as a "non-degraded" subset; otherwise, it is a "degraded" one. Then, GESAC uses a graph to express a subset and removes potential outliers by a graph matching algorithm. GESAC adapts a max-pooling strategy to solve this graph matching problem. Since the size of the graph is very small, the graph matching can be performed very efficiently.

In the model fitting step, GESAC applies a shape-controlling cost (Barron, 2019) instead of the classic least-squares cost and presents a shape-annealing robust estimate strategy in the iteratively reweighted least squares (IRLS) algorithm (Holland and Welsch, 1977) to optimize this cost. Thus, even if the subset cleaned by graph matching still contains outliers, GESAC can recover a correct solution. The hypothesize-and-verify framework, equal-length constraint, subset graph matching, and shape-annealing estimate together guarantee that GESAC is able to tolerate extremely high outlier rates. Extensive simulated and real experiments show that GESAC largely outperforms other compared state-of-the-art methods. The contributions of this paper lie in three aspects:

- We propose a new RANSAC variant for PCR that does not follow the basic idea of sampling in current RANSAC-type methods. To our best knowledge, the proposed GESAC is the only method that directly extends RANSAC by using higher than MSSs and allowing outliers in the samples for model fitting.
- We design a two-stage filtering strategy to identify good subsets. We propose an equal-length constraint to filter "degraded" subsets. We also integrate a graph matching algorithm in a RANSAC-based framework.
- We propose a coarse-to-fine optimization in the IRLS called shape-annealing robust estimate. Compared with M-estimates, the shape-annealing estimate can largely alleviate that the solver gets stuck in local minima.

2. Related work

2.1. 3D Keypoint matching

3D keypoint matching consists of three steps, which is the same as its 2D counterparts. First, keypoints are detected from point clouds based on feature detectors (e.g., local surface patches (LSP) detector (Chen and Bhanu, 2007), intrinsic shape signatures (ISS) detector (Zhong, 2009), MeshDoG detector (Zaharescu et al., 2009), or KeypointNet (Suwajanakorn et al., 2018)), which analyse the local distribution of points to identify highly distinctive points; then, these keypoints are encoded to feature vectors via descriptors (e.g., spin image descriptor (Johnson and Hebert, 1999), fast point feature histogram (FPFH) descriptor (Rusu et al., 2009), signature of histogram of orientations (SHOT) descriptor (Salti et al., 2014), or 3DSmoothNet (Gojcic et al., 2019)), so that the similarities between different keypoints can be easily computed; finally, matching scores between two sets of features are calculated and "one-to-one" corresponding relationship is established via various matching strategies such as nearest neighbor distance ratio (Lowe, 2004) or a chi-square test (Zhong, 2009). More comprehensive studies of 3D feature detectors and descriptors can be found in (Tombari et al., 2013; Guo et al., 2016).

2.2. RANSAC-based registration

Robust estimation aims to recover the geometric model from outlier contaminated observations. A large number of robust estimation algorithms have been applied to PCR problems, such as M-estimators (Zhou et al., 2016), least trimmed squares (Chetverikov et al., 2005), q-norm estimation (Li et al., 2016), truncated least squares (Yang and Carlone, 2019), weighted q-norm estimation (Li et al., 2020) and RANSAC-family (Fischler and Bolles, 1981), to name a few. Among these methods, perhaps the RANSAC and its variants are the most widely used and have become a de facto standard for correspondence-based coarse registration. RANSAC is a hypothesize-and-verify technique, which alternately performs random sampling and model fitting until the stopping criterion is reached. Specifically, a minimal required "non-degraded" subset of the input observations (e.g., three correspondences in the case of PCR) is first randomly selected; then, a rigid transformation model is fitted based on this subset and the size of consensus is calculated. These two steps are repeatedly proceeded until the stopping criterion is reached and the model with the largest consensus size is accepted as the satisfactory solution.

RANSAC has many variants. MLESAC (Torr and Zisserman, 2000) is a generalization of RANSAC, which maximizes the likelihood rather than the size of consensus. Tordoff and Murray (2005) used prior

probabilities of matches (e.g., matching scores) to guide MLESAC. Li et al. (2017, 2019a) introduced a sample checking strategy into RANSAC based on normalized barycentric coordinates. Locally optimized RANSAC (LO-RANSAC) method (Chum et al., 2003) and fixed locally optimized RANSAC (FLO-RANSAC) method (Lebeda et al., 2012) add an inner loop into RANSAC to locally refine the fitted model based on its consensus set. Randomized RANSAC (R-RANSAC) algorithm (Matas and Chum, 2004) introduces a statistical test (called $T_{(d,d)}$ test) on randomly picked observations to reduce the time complexity. However, if the outlier rate is extremely high (e.g., 90%~99%), $T_{(d,d)}$ test may reject good subset samples, which, in turn, increases the number of sampling trials. StaRSaC (Choi and Medioni, 2009) relaxes the requirement of the inlier threshold by performing standard RANSAC multiple times with different inlier thresholds. Raguram et al. (2012) proposed a universal framework for RANSAC-variants, called universal RANSAC (USAC), which incorporates many practical and computational tricks, including tricks on sampling, sample check, model check, model verification, and model refinement. Deep learning can also be used to improve the sampling of RANSAC. For example, Brachmann et al. (2017) proposed a differentiable sample consensus (DSAC) algorithm that uses a probabilistic-based hypothesis selection to learn good samples.

The goal of RANSAC-type methods is to select outlier-free samples for model fitting. Because the probability of drawing a bad sample increases exponentially with the size of the sample, all current RANSAC-variants use a minimal required size. The speed of RANSAC-variants mainly depends on two aspects, i.e., the number of sampling trials and the size of the observations. Although many efforts have been made to improve the random sampling stage, it is still very time-consuming at extremely high outliers. For example, at an outlier rate of 99%, current RANSAC-variants require more than 4.6 million trials to generate at least one good sample with a confidence of 0.99.

2.3. Graph matching

Graphs have been widely used in feature matching (Cho et al., 2010), point cloud segmentation (Xu et al., 2017), model fitting (Araújo and Oliveira, 2020; Yu et al., 2010), and object recognition (Berner et al., 2008; Wu et al., 2013), etc. Graph matching is a powerful technique that seeks the optimal correspondence between two graphs maximizing the affinities of their nodes and edges. Because the combinatorial nature of graph matching makes the global solution hardly available, a variety of approximate solutions have been presented. For example, Leordeanu and Hebert (2005) proposed an efficient approximation called spectral matching (SM) to correspondence problems based on spectral relaxation. Cour et al. (2007) proposed an extended SM named spectral matching with affine constraint (SMAC). Cho et al. (2010) introduced a random walk view for graph matching and presented a re-weighted random walk matching (RRWM) method. Similar to SM, RRWM formulates the matching problem as a node selection problem on an association graph. In another work (Cho et al., 2014), they presented a max-pooling strategy instead of traditional sum-pooling to graph matching. Max-pooling strategy can effectively suppress most of the noisy scores from outliers. Zass and Shashua (2008) proposed a new framework for graph matching from a probabilistic perspective. They formalized a soft matching criterion based on a probabilistic interpretation and induced an algebraic relation between the desired probabilistic matching and the edge matrix. Egozi et al. (2012) merged the two ideas of spectral relaxation and probabilistic framework. They interpreted the SM scheme as a maximum likelihood estimate of the assignment probabilities and proposed a new probabilistic solution based on some assumptions. Leordeanu et al. (2009) proposed an integer projected fixed-point method for objective function optimization in the integer domain. Wang et al. (2017) utilized path following strategy to improve graph matching accuracy. Due to the combinatorial nature of graph matching, the computational costs of these methods are very high, which greatly reduce their competitiveness.

2.4. Higher than minimal sampling

According to Tennakoon et al. (2015), there are two group of methods that use higher than minimal subset samples (MSSs) to solve the model-fitting problem.

The first group of methods are clustering-based. Different from traditional clustering methods that are based on a pairwise distance measure defined by two points (the size of MSSs is 2), these methods use higher than MSSs to generate affinity relations between those points and formulate the clustering as a hyper-graph partitioning problem. For example, Agarwal et al. (2005) proposed a two-step method for solving the hyper-graph partitioning problem. Liu et al. (2010) introduced a method to directly partition the hyper-graph without a pairwise graph conversion step. In their another work (Liu and Yan, 2012), they integrated a hypothesize-and-verify strategy with MSSs to approximately construct the hyper-graph. These methods are different from the proposed GESAC. The higher than MSSs here are used for computing affinities of the hyper-graph while the ones in GESAC are used for model estimation. Essentially, the meaning of the MSS in clustering is different from the one in RANSAC-type methods. The MSS here is used for distance measure computation and its size p is 2 (Agarwal et al., 2005). However, the MSS in RANSAC is model-dependent, e.g., the sizes of the MSS for a rigid model and a Homography model are 3 and 4, respectively.

Another class of methods are based on a hypothesize-and-verify technique, which use higher than MSSs to improve the quality of the hypothesis. In LO-RANSAC and FLO-RANSAC, they first use a MSS to fit a model and extract inliers that are consistent with the model; then, they generate a higher than MSS from the inliers to refine the model. Actually, they follow the basic idea of the classic RANSAC that uses MSSs to generate hypotheses. Pham et al. (2014) proposed random cluster models to generate hypotheses. They first used an adjacency graph to extract inlier structures from points and then sampled higher than MSS from the inlier structures. However, it is impractical to directly seek inlier structures in cases with high outlier rates. Moreover, the aforementioned methods cannot allow outliers in the samples. Perhaps the most relevant to our work is (Tennakoon et al., 2015). This method uses higher than MSSs to generate hypotheses and allows outliers in the samples. Unlike GESAC, it is not a RANSAC-variant. Moreover, the size of samples in this method is small, i.e., $p + 2$, and the least k-th order statistics (LkOS) cost is sensitive to high outlier rates (Chin and Suter, 2017).

As pointed out by Tennakoon et al. (2015), all these two groups of methods do not directly extend RANSAC-like methods by using higher than MSSs for hypothesis generation. The reasons are twofold: (1) Higher than MSSs increase the computational complexity due to the decreasing probability of selecting a clean sample. (2) It is difficult to know that a good hypothesis has been reached. This paper addresses these two problems and proposes a new RANSAC-variant called GESAC. For the first problem, GESAC allows outliers in the higher than MSSs, which increases probability of selecting a useful sample. For the second problem, GESAC presents a two-stage filtering strategy and a shape-annealing estimate to identify good hypotheses. GESAC also follows the stopping criterion of RANSAC.

3. Methodology

In this section, we first give the problem formulation of correspondence-based PCR. Then, we describe several important steps involved in the proposed GESAC algorithm, including equal-length constraint, point set graph matching, shape-annealing robust estimate, and stopping criterion. Finally, the computational complexity is analysed.

3.1. Problem formulation

Given a 3D correspondence set $\mathcal{M} = (\mathcal{X}, \mathcal{Y})$, where $\mathcal{X} = \{\boldsymbol{x}_i\}_1^n$ and $\mathcal{Y} = \{\boldsymbol{y}_i\}_1^n$ ($\boldsymbol{x}_i, \boldsymbol{y}_i \in \mathbb{R}^3$), such that:

$$\boldsymbol{y}_i = \mathbf{R}\boldsymbol{x}_i + \boldsymbol{t} + \boldsymbol{v}_i + \boldsymbol{\varepsilon}_i \tag{1}$$

where $\mathbf{R} \in SO(3)$ is a 3D orthogonal rotation matrix; $\boldsymbol{t} \in \mathbb{R}^3$ is a 3D translation vector; $\boldsymbol{\varepsilon}_i$ is a noise term; and $\boldsymbol{v}_i$ is a zero vector for an inlier, or an arbitrary vector for an outlier. In other words, if a correspondence $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ is an inlier, we have $\boldsymbol{y}_i = \mathbf{R}\boldsymbol{x}_i + \boldsymbol{t} + \boldsymbol{\varepsilon}_i$; otherwise, the relation between $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ can be arbitrary. The goal of registration is to estimate the optimal rigid transformation between inliers. Note that the proposed GESAC is designed for registration of LiDAR point clouds that have the same scale. Therefore, it is not suitable for cases with unknown scales such as photogrammetric point clouds.

Registration without outliers. Suppose correspondences in the set $\mathcal{M}$ are all inliers, i.e., $\boldsymbol{v}_i = \boldsymbol{0}$, $\forall i$, and $\boldsymbol{\varepsilon}_i$ is a Gaussian noise, the problem described in the above becomes a least-squares problem. Then, the optimal closed-form solution of $(\mathbf{R}, \boldsymbol{t})$ can be easily computed by using Horn's orthonormal matrix method (Horn et al., 1988).

Registration with outliers. In real applications, correspondences obtained by keypoint matching generally contain a large fraction of spurious data (outliers). However, the least-squares cost is very sensitive to outliers. A single outlier in the observations can result in arbitrarily wrong estimation. Hence, a good way is to wrap elegant closed-form solutions such as Horn's method (Horn et al., 1988) within a RANSAC-like scheme. As analysed, current RANSAC variants are not suitable for extremely high outlier rates. This paper proposes a new variant, i.e., GESAC, to deal with such a problem.

3.2. GESAC

Given a point cloud pair $(\mathcal{P}, \mathcal{Q})$, the initial correspondence set $\mathcal{M}$ is extracted by using the ISS detector (Zhong, 2009) and the FPFH descriptor (Rusu et al., 2009). Then, a hypothesize-and-verify scheme is adapted to estimate a rigid transformation model. In the first step, we repeatedly generate a subset $\mathcal{S}_k \subset \mathcal{M}$ with size $|\mathcal{S}_k| = m$ $(3 < m \ll n)$, where k is an iteration counter. An equal-length constraint is used to filter "degraded" subsets. If a subset $\mathcal{S}_k$ is "non-degraded", only correspondences that satisfy the constraint are retained, obtaining a subset $\mathcal{S}_k'$. Then, a point set graph matching algorithm is applied to obtain a cleaned subset $\mathcal{S}_k''$. In the second step, we use a shape-annealing robust estimate to estimate a rigid model $(\mathbf{R}_k, \boldsymbol{t}_k)$ for $\mathcal{S}_k''$ and compute the consensus set $C_k \subseteq \mathcal{M}$ based on an inlier threshold λ that is consistent with the model. The model $(\mathbf{R}^*, \boldsymbol{t}^*)$ with the largest consensus set is accepted as the optimal solution.

3.2.1. Equal-length constraint

Definition: Given two correct correspondences $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and $(\boldsymbol{x}_j, \boldsymbol{y}_j)$, then, the length of vector $\boldsymbol{y}_{ij} = \boldsymbol{y}_j - \boldsymbol{y}_i$ is approximately equal to the one of vector $\boldsymbol{x}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i$, i.e.,

$$(d_{ij}^{y} = \|\boldsymbol{y}_j - \boldsymbol{y}_i\|_2) \approx (d_{ij}^{x} = \|\boldsymbol{x}_j - \boldsymbol{x}_i\|_2) \tag{2}$$

where d_{ij}^{y} and d_{ij}^{x} are the lengths of $\boldsymbol{y}_{ij}$ and $\boldsymbol{x}_{ij}$, respectively.

So, if a pair of correspondences are inliers, they must satisfy the equal-length constraint. As known, at least three non-collinear inliers are required to estimate a rigid transformation in 3D space. These inliers can construct at least 3 pairs of correspondences that satisfy the constraint. In GESAC, we first construct $m(m-1)/2$ pairs of correspondences for a subset $\mathcal{S}_k$; then, the equal-length constraint is checked for each pair $\{(\boldsymbol{x}_i, \boldsymbol{y}_i), (\boldsymbol{x}_j, \boldsymbol{y}_j)\}$ based on $|d_{ij}^{y} - d_{ij}^{x}| \leqslant \lambda$. If the number of pairs inside $\mathcal{S}_k$ that satisfy the constraint is smaller than 3, the subset is regarded as a "degraded" one. For a "non-degraded" subset, only correspondences that satisfy the constraint are retained, obtaining $\mathcal{S}_k'$. $\mathcal{S}_k'$ can not be directly used for least-squares estimation such as Horn's method, since a pair of outlier correspondences may also have the same length. $\mathcal{S}_k'$ usually contains outliers. Hence, the outlier filtering strategy (point set graph matching) and robust estimation (shape-annealing estimate) are necessary.

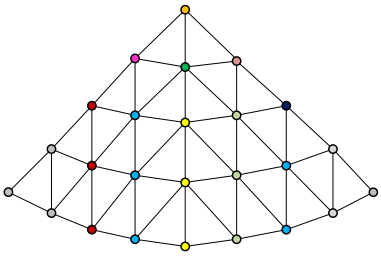


Fig. 1. A graph constructed by a point set, where the nodes represent points and the edges represent relations between points.

3.2.2. Point set graph matching

A graph is a useful data structure for the representation of point sets, which has been widely used in computer vision and photogrammetry. Fig. 1 shows a graph, where the nodes represent points and the edges describe relations between points. Graphs have some important properties, i.e., they are invariant to translation, rotation, and scale changes, etc. Thus, graphs are often used in correspondence problems. Graph matching refers to a problem that seeks the optimal correspondence between two graphs to maximize the affinities of their nodes and edges.

Graph matching problem: Given two graphs $G = (V, E)$ and $G' = (V', E')$, where V (V') is a set of node vertices and E (E') is an edge set, graph matching aims to find a subset of potential correct matches represented by a binary assignment matrix $\mathbf{Z} \in \{0,1\}^{n_G \times n_{G'}}$ that maximizes the similarity between G and G', where n_G and n_G' are the sizes of node sets V and V', respectively. If a pair of correspondence $(v_i \in V, v_a' \in V')$ is matched, then $z_{ia} = 1$; otherwise, $z_{ia} = 0$. (z_{ia} is an element in the i-th row and a-th column of $\mathbf{Z}$) A widely used objective function that measures the similarity is defined as,

$$\begin{aligned} \arg\max_{z} f_z &= \arg\max_{z} \sum_{z_{ia}=1} c_V(v_i, v_a') + \sum_{\substack{z_{ia}=1 \\ z_{jb}=1}} c_E(e_{ij}, e_{ab}') \\ &= \arg\max_{z} \boldsymbol{z}^T \mathbf{A} \boldsymbol{z} \end{aligned} \tag{3a}$$

$$s.t. \quad \boldsymbol{z} \in \{0,1\}^{n_G n_{G'}}, \ \sum_{i=1}^{n_G} z_{ia} \leqslant 1, \ \sum_{a=1}^{n_{G'}} z_{ia} \leqslant 1 \tag{3b}$$

where $c_V(v_i, v_a')$ is a node similarity term and $c_E(e_{ij}, e_{ab}')$ is an edge similarity term; $e_{ij} \in E$ ($e_{ab}' \in E'$); $\boldsymbol{z}$ is a $n_G n_{G'} \times 1$ vectorized replica of matrix $\mathbf{Z}$; $\mathbf{A}$ is a symmetric affinity matrix, whose diagonal elements are assigned node similarities and non-diagonal elements are assigned edge similarities; Eq. (3b) enforces a one-to-one matching constraint. Based

on the Taylor expansion, function f_z can be approximated by,

$$f_z = f_{z_{k'}} + (z - z_{k'})^T \mathbf{A} z_{k'} \tag{4}$$

where k' is an iteration counter. Then, z can be optimized via an iteration procedure,

$$z_{k'+1} := \frac{\mathbf{A} z_{k'}}{\|\mathbf{A} z_{k'}\|_2} \tag{5}$$

Algorithm 1. Point set graph matching

Input: A correspondence set: $\mathcal{S}' = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_1^{m'}$
Output: a cleaned correspondence set $\mathcal{S}'' = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_1^{m''}$
Build graphs G and G', and construct the affinity matrix $\mathbf{A}$;
Initialize $k' := 0$ and initial score vector z_0 as uniform;
while z *not converged* **do**
  for *each candidate match* (i, a) **do**
    Compute its confidence $(z_{k'})_{ia}$ based on Eq. (8);
  end
  Update the confidence based on $z_{k'+1} := z_{k'}/\|z_{k'}\|_2$;
  $k' := k' + 1$;
end
Reshape z to a matrix $\mathbf{Z}$;
Binarize $\mathbf{Z}$ via adaptive thresholding;
Compute the sum of each row $\sum_i z_{ia}$ or column $\sum_a z_{ia}$;
if $\sum_i z_{ia} > 1$ *or* $\sum_a z_{ia} > 1$ **then**
  The element with maximum score $\rightarrow$ 1, others $\rightarrow$ 0;
end
Extract a "correct" match set $\mathcal{S}''$ according to $\mathbf{Z}$;

Point set matching: After obtaining a subset $\mathcal{S}'$ based on the equal-length constraint, we build graphs G and G', where the correspondences belonging to $\mathcal{S}'$ are regarded as nodes and the linkages are edges. The edge similarity $c_E(e_{ij}, e'_{ab})$ is defined as the exponential length difference,

$$c_E(e_{ij}, e'_{ab}) = \exp\left(-\left(\|v_i - v_j\|_2 - \|v'_a - v'_b\|_2\right)^2\right) \tag{6}$$

The node similarity is difficult to measure and is not reliable. Hence, we set $c_V(v_i, v'_a) = 0$. The affinity matrix $\mathbf{A}$ is then constructed. Each element z_{ia} assigns a confidence score for a candidate match (v_i, v'_a). The score z_{ia} is updated by Eq. (5) where $\mathbf{A}z$ accumulates the weighted affinities (sum-pooling strategy). Here, we use a max-pooling strategy (Cho et al., 2014) instead of the sum-pooling strategy to update the confidence scores,

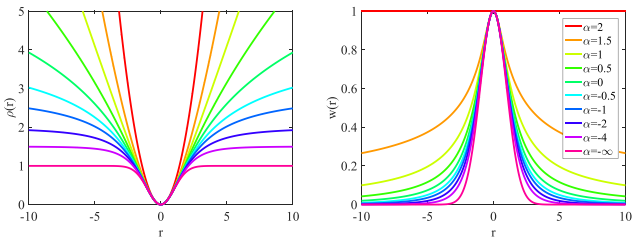
$$z_{k'+1} := \frac{\mathbf{A} \otimes z_{k'}}{\|\mathbf{A} \otimes z_{k'}\|_2} \tag{7}$$

where

$$(\mathbf{A} \otimes z_{k'})_{ia} = \sum_{j \in \mathcal{N}_i} \max_{b \in \mathcal{N}_a} z_{jb} C_E(e_{ij}, e'_{ab}) \tag{8}$$

$\mathcal{N}_i$ and $\mathcal{N}_a$ are sets of neighbors of i and a, respectively. The max-pooling strategy, in essence, is to assign the best supporting matches for each candidate correspondence. Therefore, the strategy is able to suppress the noise arise from outliers. To prevent one-to-many or many-to-one matching, a final discretization step should be performed to enforce the one-to-one matching constraint. Specifically, we first binarize the confidence score matrix $\mathbf{Z}$ by using an adaptive thresholding method; then, each row and each column of $\mathbf{Z}$ are summed. If the sum of a row or a column is larger than 1, only the correspondence with the maximum score in this row or column is accepted as correct. The details of point set graph matching are summarized in **Algorithm 1**.

There are two reasons why we choose graph matching rather than other methods for outlier filtering. First, graph matching can establish an optimal matching relationship and is not sensitive to outlier rates. In practice, graph matching is not very popular since its computational complexity is large. Fortunately, we adapt the graph matching in our GESAC. The node size of the graph is very small, i.e., $m' = |\mathcal{S}'_k| \leqslant m \ll n$. Thus, the graph can be efficiently optimized. Second, the equal-length constraint can be implicitly involved in the objective function. In rigid point set matching, the length between points is a relatively reliable metric.



(a) (b)

Fig. 2. Visualization of the generalized robust cost (a) and its weight function (b) with different shape controlling parameters. ($c = 1$).

3.2.3. Shape-annealing robust estimate

RANSAC and its variants use the least-squares cost for rigid transformation estimation, which is extremely sensitive to outliers. This paper aims to solve the PCR problem with extreme amounts of outliers. The subset $\mathcal{S}''$ cleaned by graph matching may still contain outliers. Once a good subset is generated, we want to ensure that the correct solution can be estimated since the number of required iterations to get a good subset is large. Our goal is to minimize the distances between matching points, while disabling spurious correspondences. The objective function is shown in Eq. (9),

$$\sum_{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{S}''} \rho(\|\boldsymbol{y}_i - (\mathbf{R}\boldsymbol{x}_i + \boldsymbol{t})\|_2) \tag{9}$$

where $r_i = \|\boldsymbol{y}_i - (\mathbf{R}\boldsymbol{x}_i + \boldsymbol{t})\|_2 = \|\boldsymbol{v}_i + \boldsymbol{\varepsilon}_i\|_2$ is the residual of $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and $\rho(r_i)$ is a robust penalty function. The cost $\rho(r_i)$ reflects the amount of influence of a residual r_i towards the optimization energy. For inliers, $r_i = \|\boldsymbol{\varepsilon}_i\|_2$, which is very small compared with the residuals of outliers. Thus, the optimization energy is generally dominated by outliers. It is critical to adapt an appropriate robust penalty. In our GESAC, we use a robust cost with shape controlling (Barron, 2019),

$$\rho(r_i, \alpha) = \begin{cases} \frac{1}{2}(r_i/c)^2 & \text{if } \alpha = 2 \\ \log\left(\frac{1}{2}(r_i/c)^2 + 1\right) & \text{if } \alpha = 0 \\ 1 - \exp\left(-\frac{1}{2}(r_i/c)^2\right) & \text{if } \alpha = -\infty \\ \frac{|\alpha - 2|}{\alpha}\left(\left(\frac{(r_i/c)^2}{|\alpha - 2|} + 1\right)^{\alpha/2} - 1\right) & \text{if } \alpha = \frac{1}{4}h \end{cases} \tag{10}$$

where α is a shape parameter; c is a constant scale parameter; and $h \in \mathbb{Z}\ \&\ h \neq 0\ \&\ h \neq 8$. This robust cost is a generalization of traditional M-estimates such as least-squares cost, Cauchy cost, Welsch cost, etc. For example, if $\alpha = 2$, it is a least-squares cost; if $\alpha = 0$, it is a

Cauchy estimate; if $\alpha = -2$, it is a Geman-McClure estimate; and if $\alpha = -\infty$, it becomes a Welsch estimate. The generalized cost is visualized in Fig. 2(a).

Like M-estimators, solving Eq. (9) is equivalent to solve a weighted least-squares (WLS) problem, minimizing the form of $\sum_i w_i r_i^2$. w_i is a weight function, which is generally defined by $w_i = \frac{\partial \rho}{\partial r_i}/r_i$, where $\frac{\partial \rho}{\partial r_i}$ is the derivative of $\rho$with respect to the residual r_i. Thus, the weight function of the generalized cost is,

$$w(r_i,\alpha) = \begin{cases} \dfrac{1}{c^2} & \text{if } \alpha = 2 \\ \dfrac{2}{r_i^2 + 2c^2} & \text{if } \alpha = 0 \\ \dfrac{1}{c^2}\exp\left(-\dfrac{1}{2}(r_i/c)^2\right) & \text{if } \alpha = -\infty \\ \dfrac{1}{c^2}\left(\dfrac{(r_i/c)^2}{|\alpha-2|}+1\right)^{(\alpha/2-1)} & \text{if } \alpha = \dfrac{1}{4}h \end{cases} \tag{11}$$

Algorithm 2. Shape-annealing rigid model estimation

Input: A correspondence set: $\mathcal{S}'' = \{(x_i,\ y_i)\}_1^{m''}$
Output: a rigid transformation $(\mathbf{R}, \boldsymbol{t})$
Initialize $k'' := 0$, $\{w_i\} := 1$, and $\alpha := 2$;
while $(\mathbf{R}, \boldsymbol{t})$ *not converged* **do**
  Construct a weighted least squares problem;
  Find a close-form solution $(\mathbf{R}_{k''},\ \boldsymbol{t}_{k''})$ via Horn's method;
  Compute the residuals $\{r_i\}_1^{m''}$ for $\{(x_i,\ y_i)\}_1^{m''}$;
  Update the weights $\{w_i\}_1^{m''}$ according to Eq. (11);
  if $(k'' \bmod t) = 0$ **then**
    Anneal the shape parameter α;
  end
  $k'' := k'' + 1$;
end
Compute $(\mathbf{R}, \boldsymbol{t})$ via Horns method with the newest weights;

Fig. 1(b) plots the weight function with different shape controlling parameters. As shown, least-squares cost gives uniform weights to all observations (both inliers and outliers). Robust costs (such as $\alpha = 0, \alpha = -2$, and $\alpha = -\infty$) give large weights (close to 1) to observations with small residuals while giving small weights (close to 0) to observations with large residuals. Thus, robust costs can largely discount the effect of outliers towards to the total energy.

Generally, the WLS problem is optimized by IRLS. In our optimization, we do not fix a value for α. We anneal the shape of the weight function in the iteration procedure of IRLS. We call this strategy as shape-annealing robust estimate. From Fig. 2(b), we can see that large values of αmake the weight function smoother and more observations are allowed to take part in optimization. In contrast, the curve of the weight function becomes more pointy as αdecreases, which makes the parameter estimation more precise. Specifically, $\alpha$is first set to 2 and is annealed every $t = 3$ iterations over the values $[2,\ 1,1/2,\ 1/4,0,\ -1/4,\ -1/2,\ -1,\ -2,\ -4,\ -8,\ -16,\ -32,\ -\infty]$, where t is an interval of iterations to anneal the parameter α. The shape-annealing estimate is more robust than M-estimates, since the coarse-to-fine strategy can largely alleviate that the solver gets stuck in local minima. The shape-annealing robust estimate for rigid transformation estimation is summarized in **Algorithm 2**.

Algorithm 3. GESAC

Input: A pair of point clouds $(\mathcal{P}, \mathcal{Q})$
Output: The optimal rigid transformation $(\mathbf{R}^*, \boldsymbol{t}^*)$
Extract a correspondence set $\mathcal{M}$ via ISS and FPFH from $(\mathcal{P}, \mathcal{Q})$;
Initialize $k := 0$, $|C^*| := 0$, $p_0 = 0.99$, $m = 32$, $N := 1e5$;
while $k < N$ **do**
  (1) Sampling-step:
  Randomly select a subset $\mathcal{S}_k \subset \mathcal{M}$, $|\mathcal{S}_k| = m$;
  Perform equal-length constraint check on $\mathcal{S}_k$;
  if *"non-degraded"* **then**
    find a set $\mathcal{S}'_k \subseteq \mathcal{S}_k$ that satisfies the constraint;
  else
    return to line 5, $k := k + 1$;
  end
  Perform point set graph matching on $\mathcal{S}'_k$ (**Algorithm 1**);
  if *"non-degraded"* **then**
    find a cleaned set $\mathcal{S}''_k \subseteq \mathcal{S}'_k$;
  else
    return to line 5, $k := k + 1$;
  end
  (2) Model estimation-step:
  Fit $(\mathbf{R}_k,\ \boldsymbol{t}_k)$ for $\mathcal{S}''_k$ via shape-annealing estimate (**Algorithm 2**);
  Compute a set $C_k \subseteq \mathcal{M}$ that is consistent with $(\mathbf{R}_k,\ \boldsymbol{t}_k)$;
  if $|C_k| > |C^*|$ **then**
    Perform a least-squares on C_k, obtaining $(\mathbf{R}^*_k,\ \boldsymbol{t}^*_k)$;
    Compute a refined consensus set C^*_k ;
    $C^* := C^*_k$, $\mathbf{R}^* := \mathbf{R}^*_k$, $\boldsymbol{t}^* := \boldsymbol{t}^*_k$;
    Update number of required trials N according to Eq. (13);
  end
  $k := k + 1$;
end
return $(\mathbf{R}^*, \boldsymbol{t}^*)$;

3.2.4. Stopping criterion

In RANSAC and its variants, the standard stopping criterion is based on the minimum number of required trials to ensure with confidence p_0 that at least one good subset (outlier-free) can be obtained. The proposed GESAC also follows a similar criterion. In contrast to RANSAC, GESAC regards a subset with at least 3 inliers as good instead of outlier-free. Hence, a bad subset includes three cases, i.e., (1) 2 inliers and $m-2$ outliers, (2) 1 inliers and $m-1$ outliers, (3) 0 inliers and m outliers. Suppose the true inlier rate of the initial correspondence set is η, then the probability to select a bad subset is,

$$p = \frac{m(m-1)}{2}\eta^2(1-\eta)^{m-2} + m\eta(1-\eta)^{m-1} + (1-\eta)^m \tag{12}$$

and the probability of drawing N bad subsets is p^N. To ensure that at least one good subset is obtained, the probability p^N should fall below a threshold $(1-p_0)$,

$$p^N \leqslant (1-p_0) \Leftrightarrow N \geqslant \frac{\log(1-p_0)}{\log p} \tag{13}$$

In practice, confidence p_0 is often set to 0.99 and the true inlier rate $\eta$is unknown. Generally, $\eta$is approximately computed based on the size of current maximum consensus set and updated as the iteration procedure progresses. Table 1 gives the minimum number of required trials N with respect to different sizes of m at an inlier rate of 0.01. Current RANSAC and its variants use a minimal subset for rigid transformation estimation ($m = 3$). They require at least $N = 4605168$ trials to obtain a good sample with confidence $p_0 = 0.99$. Our GESAC with $m = 32$ only

Table 1
Number of required trials N with respect to different m.

m	3	8	14	20	26	32	38
N	4605168	85385	13739	4587	2102	1151	707

Table 2
Detailed settings of the compared state-of-the-art algorithms.

Method	Parameters	Implementation	Input
RANSAC	Subset size m: 3; confidence p_0: 0.99; maximum iterations: 10^5.	MATLAB code; single thread https://www.peterkovesi.com/matlabfns/index.html#robust	Correspondences
R-RANSAC	Subset size m: 3; confidence p_0: 0.99; $T_{(d,d)}$ test: $d = 10$; maximum iterations: 10^5.	MATLAB code; single threadhttps://www.peterkovesi.com/matlabfns/index.html#robust	Correspondences
LO-RANSAC	Subset size m: 3; confidence p_0: 0.99; maximum iterations: 10^5; local optimized size: $\|C_k\|/2$; local optimized iterations: 10.	MATLAB code; single thread https://zhipengcai.github.io/	Correspondences
FLO-RANSAC	Subset size m: 3; confidence p_0: 0.99; maximum iterations: 10^5; local optimized subset size: 21; local optimized iterations: 50.	MATLAB code; single thread https://zhipengcai.github.io/	Correspondences
FGR	Annealing rate: 1.2.	C++ code; single thread https://github.com/intel-isl/FastGlobalRegistration	Correspondences
S4PCS	Overlapping ratio: 50%; subset size: 5000; registration accuracy: 0.1 m; maximum running time: 1000s.	C++ code; single thread https://github.com/nmellado/Super4PCS	Point clouds
K4PCS	Overlapping ratio: 50%; score threshold: 0.001; registration accuracy: 0.1 m; maximum running time: 1000s.	C++ code; single thread http://pointclouds.org/	ISS keypoints
PointNetLK	Training set: ModelNet40; Number of epochs: 200 Maximum number of points per scan: 40 K.	Python; GPU (GTX 1080Ti 11 GB) https://github.com/hmgoforth/PointNetLK	Downsampled points
Our GESAC	Subset size m: 32; confidence p_0: 0.99; maximum iterations: 10^5; maximum iterations of GM: 50; maximum iterations of IRLS: 100.	MATLAB code; single thread https://www.researchgate.net/profile/Jiayuan_Li2	Correspondences

requires $N = 1151$ trials if $p_0 = 0.99$, which is only 1/4000 of the ones of RANSAC variants.

3.3. Computational complexity

GESAC is a procedure with N iterations, whose inner loop consists of three main stages, including equal-length constraint, point set graph matching, and shape-annealing estimate. The details of GESAC are summarized in **Algorithm 3**. The equal-length constraint requires the construction of $m(m-1)/2$ correspondence pairs for subset $\mathcal{S}_k$. Its time complexity is $O(m^2)$. The computational complexity of point set graph matching is $O(k^{'}m^{'2}l^2)$, where $k^{'}$ is the average required number of iterations that the assignment vector z converges, $m^{'}$ is the size of subset $\mathcal{S}_k^{'}$, and l is the size of neighbor sets $\mathcal{N}_i$ and $\mathcal{N}_a$. In the shape-annealing estimate, the complexity of Horn's method, weight update, and residual computation is $O(m'')$, where m'' is the size of subset $\mathcal{S}_k''$. Let the average required number of iterations be k'', then the total time complexity of the shape-annealing estimate is $O(k''m'')$. The time complexity of consensus set computation in GESAC is $O(n)$. In GESAC, the equal-length constraint is applied N times, while point set graph matching and shape-annealing estimate are performed much less. Let the execution times of point set graph matching and shape-annealing estimation be $N^{'}$ and N'', the total complexity of GESAC is $O\left(m^2N + k^{'}m^{'2}l^2N^{'} + k''m''N'' + nN''\right)$. Since $k''m''N'' \ll m^2N$, the complexity can be simplified as $O\left(m^2N + k^{'}m^{'2}l^2N^{'} + nN''\right)$. In the above, we have found that large values of m make the required number of trials N smaller. However, the value of m is not the larger the better. From the time complexity, although large m can reduce the number of trials N, it increases the value of m^2 and the complexity of point set graph matching.

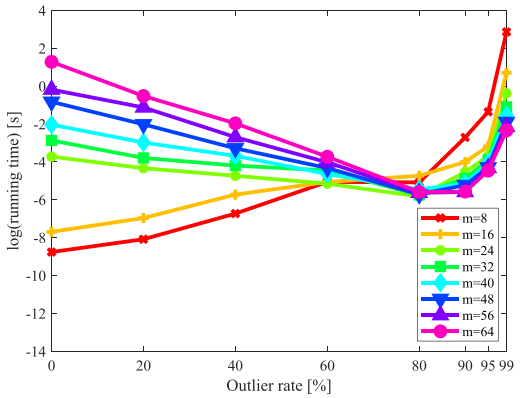
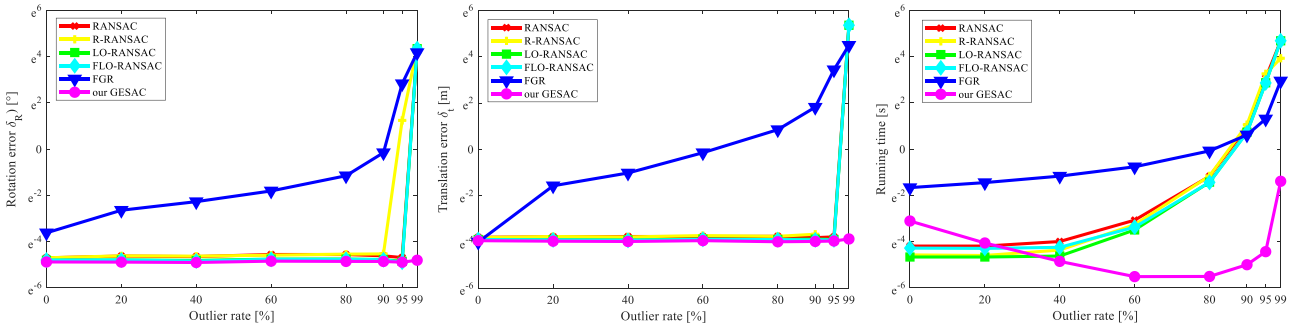


Fig. 3. The runtime results with respect to different values of m.

4. Experiments and evaluations

In this section, the proposed GESAC is comprehensively evaluated on both synthetic data and real point cloud datasets. First, a simulated coarse registration experiment is conducted to learn a reasonable value for the parameter m. Then, our GESAC is compared with seven other state-of-the-art algorithms, including RANSAC (Fischler and Bolles, 1981), R-RANSAC (Matas and Chum, 2004), LO-RANSAC (Chum et al., 2003), FLO-RANSAC (Lebeda et al., 2012), FGR (Zhou et al., 2016), super 4-points congruent sets (S4PCS) proposed by Mellado et al. (2014), Keypoint-based 4-points congruent sets (K4PCS) proposed by Theiler et al. (2014), and PointNetLK (Aoki et al., 2019). In our experiments, the inlier threshold $\lambda$is set to triple of the standard deviation of noise (or triple of the average point cloud resolution). For the PointNetLK, we use a pretrained model that was trained on the ModelNet40 dataset.[2] The experiments of PointNetLK are performed on a desktop PC with a GeForce GTX 1080Ti @ 11 GB GPU. PointNetLK can not deal with point clouds with large sizes. For example, it can only tolerate up to 40 K points per scan on a GPU with 10 GB memory. Thus, we downsample each scan of the ETH dataset to 40 K points based on a random sampling method. The detailed information (parameters, implementations, and inputs) of each algorithm is summarized in Table 2. Two widely used metrics for PCR problem are chosen for quantitative evaluation, i.e., rotation error $\delta_{\mathbf{R}}$ and translation error δ_t,

[2] https://modelnet.cs.princeton.edu/.



(a) rotation error $\delta_{\mathbf{R}}$ (b) translation error δ_t (c) running time

Fig. 4. Comparisons on the simulated data. Subfigures (a), (b), and (c) plot the rotation errors, translation errors, and runtime, respectively. The horizontal axis is the outlier rate (0–99%) and the vertical axis is the accuracy metrics or runtime.

Table 3
Detailed information about the ETH dataset.

Info	Arch	Courtyard	Facade	Office	Trees
Overlap	30–40%	40–70%	60%~70%	>80%	≈50%
$\mathcal{N}_{pair}$	8	28	21	8	10
$\mathcal{N}_{point}$	27423 K	12194 K	17343 K	10721 K	20248 K
$\mathcal{N}_{key}$	22281	15795	8175	4906	15858
$\lvert\mathcal{M}\rvert$	15975	33991	15545	11628	22264

$$\begin{cases} \delta_t = \|\boldsymbol{t}^t - \boldsymbol{t}^e\|_2 \\ \delta_{\mathbf{R}} = \arccos\dfrac{tr(\mathbf{R}^t(\mathbf{R}^e)^T) - 1}{2} \end{cases} \tag{14}$$

where $\mathbf{R}^t$ and $\mathbf{R}^e$ are the true rotation matrix and the estimated one, respectively; $tr(\cdot)$ is the trace of a matrix; $\boldsymbol{t}^t$ and $\boldsymbol{t}^e$ are the true translation vector and the estimated one, respectively. $\delta_{\mathbf{R}}$ measures the angular distance between $\mathbf{R}^t$ and $\mathbf{R}^e$, and δ_t is the Euclidean distance between $\boldsymbol{t}^t$ and $\boldsymbol{t}^e$. All the experiments except the PointNetLK are performed on a laptop with single CPU Core i7-8550U @ 1.8 GHz, and 8 GB of RAM. The source code will be made publicly available.[3]

4.1. Parameter m study

In GESAC, the choice of the parameter m is very important, influencing the number of required trials and running time. Here, we select a reasonable value for m based on a synthetic PCR experiment. Specifically, the input outlier contaminated correspondences $\mathcal{M} = (\mathcal{X}, \mathcal{Y})$ are generated as follows: Given an outlier rate r_{out} and the number of inliers n_{in}, the number of initial correspondences is computed by $n = n_{in}/(1 - r_{out})$. First, we randomly generate n_{in} points $\mathcal{X}_{in} = \{\boldsymbol{x}_i\}_1^{n_{in}}$ in $\mathbb{R}^3$ using a normal distribution $\mathcal{N}(0, 100^2)$. These points are transformed to another local coordinate system via transformation $\boldsymbol{y}_i^t = \mathbf{R}^t\boldsymbol{x}_i + \boldsymbol{t}^t$, obtaining their correspondences $\mathcal{Y}_{in}^t = \{\boldsymbol{y}_i^t\}_1^{n_{in}}$, where the ground truth rotation matrix $\mathbf{R}^t$ is a randomly generated 3×3 Rodrigues matrix (rotation angles are within $[-\pi/2, \pi/2]$) and the ground truth translation $\boldsymbol{t}^t$ is a randomly generated 3×1 vector whose elements are distributed in [-100, 100]. To make the simulation more realistic, Gaussian noise $\mathcal{N}(0, 0.1^2)$ is then added to $\mathcal{Y}_{in}^t$, obtaining the inlier correspondence set $\mathcal{M}_{in} = (\mathcal{X}_{in}, \mathcal{Y}_{in})$. Finally, two sets of 3D points with size $n - n_{in}$ are randomly generated using a normal distribution $\mathcal{N}(0, 100^2)$, obtaining outlier correspondence set $\mathcal{M}_{out} = (\mathcal{X}_{out}, \mathcal{Y}_{out})$. The inlier set $\mathcal{M}_{in}$ and outlier set $\mathcal{M}_{out}$ are merged to obtain the initial correspondence set $\mathcal{M}$. In our experiment, we fix the number of inliers to 80 and sequentially increase the outlier rate r_{out} ([0, 20%, 40%, 60%, 80%, 85%, 90%, 95%, 99%]).

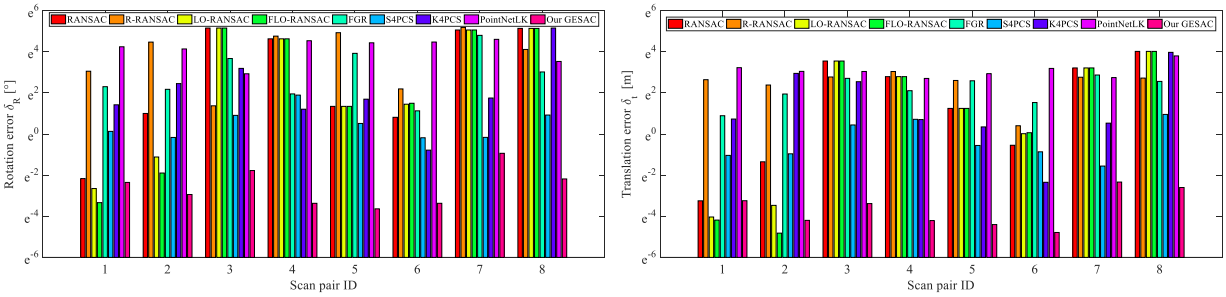
Values above 3 for parameter m have almost no influence on the registration accuracy of our GESAC. The value of m influences the value of N and affects the time complexity. Thus, we study the running time of the proposed GESAC with respect to different values of m. The results are plotted in Fig. 3, where each reported value is the average of 1000 individual tests. As shown, when the outlier rate is low (lower than 60%), the smaller the value of m, the faster the running speed; however, when the outlier rate is high (higher than 60%), the results are completely opposite. Therefore, neither a very large value nor a very small value of m is a good choice. This conclusion can also be inferred from the expression of the computational complexity. When the outlier rate is low, the required number of trials N is small. In this case, increasing the value of m will largely increase the complexity of $O\left(k'm'^2l^2N'\right)$. In contrast, when the outlier rate is high, the term $O(m^2N)$ is the dominant one. Based on the analysis in the above, parameter m is set to 32 in all the following experiments.

4.2. Evaluations on simulated data

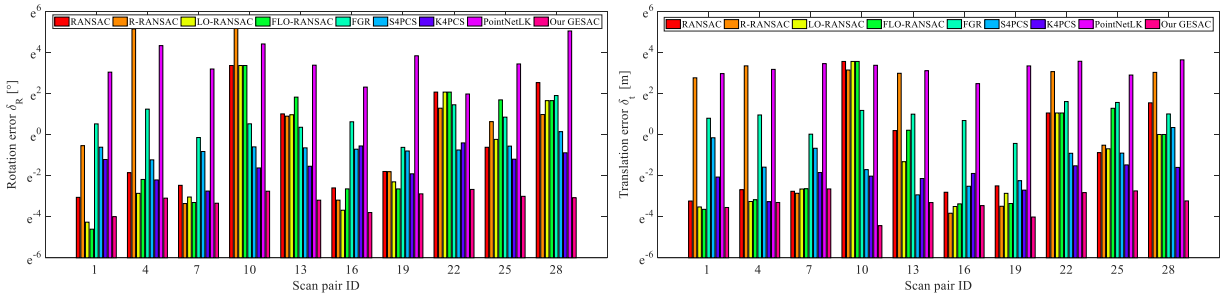
The simulation process is the same as for the study of parameter m. Since we simulate correspondence set $\mathcal{M} = (\mathcal{X}, \mathcal{Y})$ instead of point clouds as input, this section only compares the proposed GESAC with correspondence-based registration algorithms in Table 2, including RANSAC, R-RANSAC, LO-RANSAC, FLO-RANSAC, and FGR. An orthonormal matrix algorithm is used in RANSAC-type methods for close-form rigid transformation estimation. Fig. 4 reports the average quantitative results.

As shown in Fig. 4 and Fig. 4(b), RANSAC-type methods achieve similar registration accuracies. They perform very well before the outlier rate reaches 95%. However, if the outlier rate is extremely high, they often fail to find a correct solution within 10^5 trials. We regard a registration test with $\delta_{\mathbf{R}} < 1°$ and $\delta_t < 0.5$ m as a successful one. Then, at an outlier rate of 99%, the success rates of RANSAC (9.6%), R-RANSAC (2.1%), LO-RANSAC (14.8%), and FLO-RANSAC (14.8%) are all lower than 20%. FGR performs the worst, whose registration accuracy is almost always the lowest. The model fitting in FGR is sensitive to

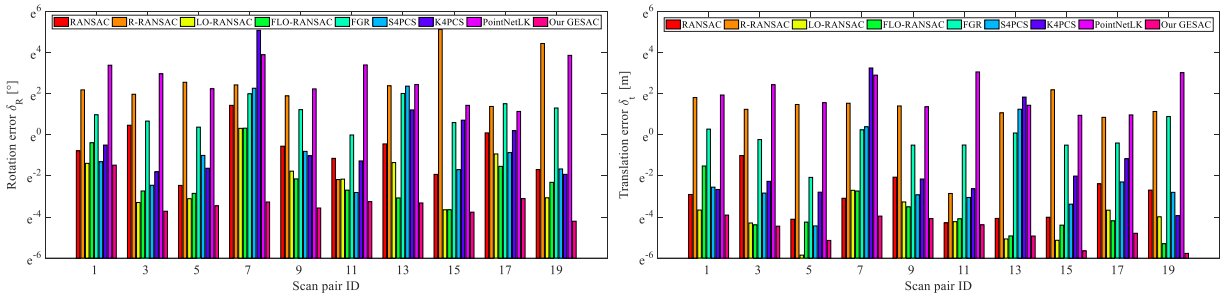
[3] http://www.escience.cn/people/lijiayuan/index.html.



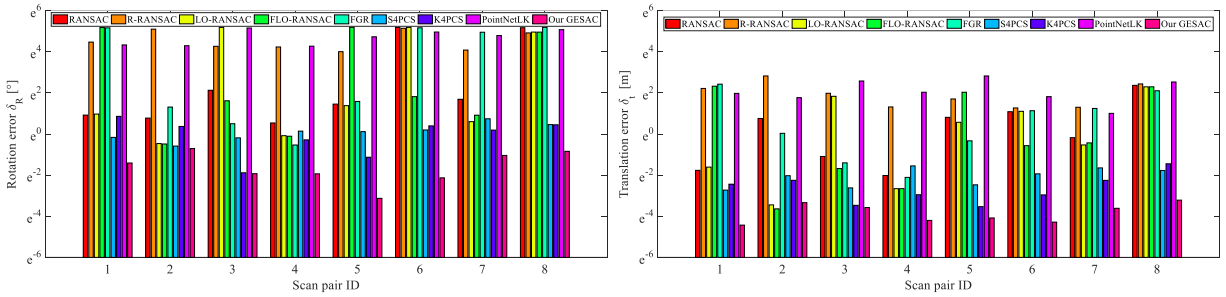
(a) Arch



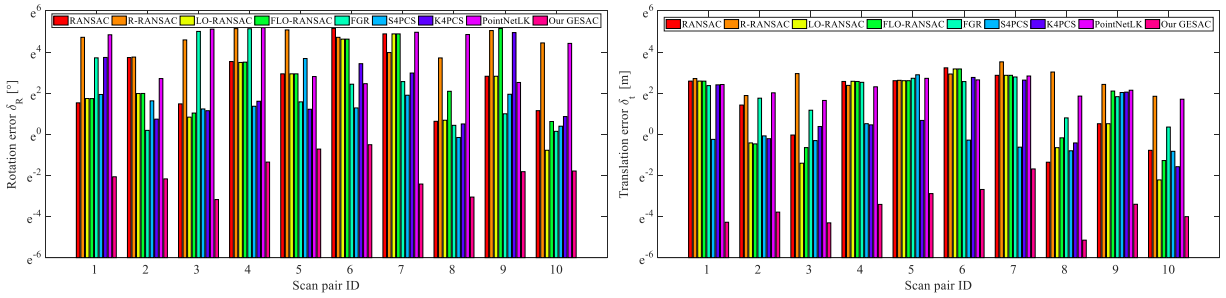
(b) Courtyard



(c) Facade



(d) Office



(e) Trees

Fig. 5. Evaluations on the challenging ETH dataset. Subfigures (a)–(e) plot the comparison results on the Arch, Courtyard, Facade, Office, and Trees, respectively. The rotation error $\delta_{\mathbf{R}}$ is reported in the left and the translation error δ_t is displayed in the right. The scan pair ID is an identifier for each LiDAR scan pair.

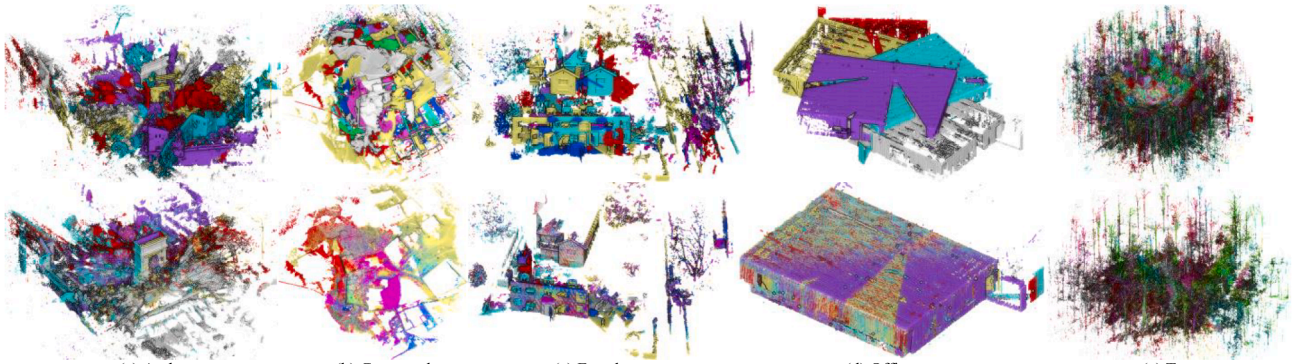


Fig. 6. Complete registration results. First row is the input LiDAR scans, where Arch contains 5 scans, Courtyard contains 8 scans, Facade contains 7 scans, Office contains 5 scans, and Trees contains 6 scans. Different scans are represented by different colors. As shown, the input point clouds are messy. Second row shows the registration results of GESAC. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Average rotation error $\delta_{\mathbf{R}}$ [°] results on the ETH dataset.

Method	Arch	Courtyard	Facade	Office	Trees
RANSAC	76.89	4.45	0.64	47.65	43.22
R-RANSAC	77.37	13.18	18.88	102.11	103.24
LO-RANSAC	76.84	3.56	0.24	64.01	32.15
FLO-RANSAC	76.84	3.84	0.25	64.45	48.48
FGR	32.71	2.32	2.97	85.52	39.65
S4PCS	2.13	0.46	1.32	1.18	7.89
K4PCS	28.67	0.74	16.73	1.17	25.13
PointNetLK	69.14	32.60	23.70	116.82	88.31
Our GESAC	**0.12**	**0.04**	**0.05**	**0.25**	**0.21**

outliers, i.e., the fitting accuracy severely decreases as the outlier increases. Therefore, FGR is not suitable for cases with extreme amounts of outliers. In contrast, the proposed GESAC performs the best among these six methods, which can tolerate extremely high outlier rates. Its success rate is 100% at an outlier rate of 99%, and its $\delta_{\mathbf{R}}$ and δ_t are 0.008° and 0.018 m on average, respectively. Although GESAC is also a RANSAC variant, its registration accuracy is slightly better than the compared RANSAC-type methods. The reason may be twofold: First, GESAC uses large subsets for model estimation. There are many redundant observations, which improve the robustness to noise. Second, GESAC uses a robust cost instead of least-squares cost. The robust cost can give large weights to correspondences with small residuals while giving small weights to the ones with large residuals.

Fig. 4(c) shows the results of runtime. As shown, GESAC is comparable with RANSAC-type methods at low outlier rates. However, if the outlier rate is high, RANSAC-type methods become much slower than the proposed method. For example, GESAC is 420 + times faster than RANSAC when the outlier rate is 99%. If we set the minimum number of required trials to 4.6×10^6, GESAC will be 4000 + times faster than RANSAC. Although FGR is implemented by C++ code, it is still 60 + times slower than the proposed GESAC.

4.3. Evaluations on challenging real data

GESAC is also evaluated on a large-scale challenging real world dataset, i.e., the ETH LiDAR dataset[4]. This dataset consists of five categories, including Arch, Courtyard, Facade, Office, and Trees, each of which contains multiple LiDAR scan pairs. The ISS (Zhong, 2009) detector is used to extract features and the FPFH (Rusu et al., 2009) is used as a feature descriptor[5]. We use a strategy that is similar to (Cai et al., 2019; Li et al., 2020) for the initial correspondence set $\mathcal{M} = (\mathcal{X}, \mathcal{Y})$ establishment. Specifically, a correspondence $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ is selected into $\mathcal{M}$ only if $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ are one of the top-5 best matches to each other. The detailed information about the ETH dataset is summarized in Table 3, including overlapping ratios, number of scan pairs $\mathcal{N}_{pair}$, average number of points in a scan $\mathcal{N}_{point}$, average number of keypoints $\mathcal{N}_{key}$, and average number of correspondences $|\mathcal{M}|$. The ETH dataset contains ground truth, where each scan pair is associated with a true rigid transformation $(\mathbf{R}^t, \boldsymbol{t}^t)$. A correspondence is an inlier only if it satisfies $\|\boldsymbol{y}_i^t - (\mathbf{R}^t\boldsymbol{x}_i + \boldsymbol{t}^t)\|_2 < \lambda$. The inlier threshold $\lambda$is set to three times the average resolution. It is very challenging to register the ETH dataset, since the outlier rates of the correspondence sets are extremely high. The outlier rates of the Arch, Courtyard, Facade, Office, and Trees datasets are 99.53%, 98.01%, 98.50%, 99.14%, and 99.64%, respectively. In this dataset, point clouds are available. Hence, we add three more state-of-the-art algorithms for comparison, i.e., S4PCS (Mellado et al., 2014), K4PCS (Theiler et al., 2014), and PointNetLK (Aoki et al., 2019), which are not based on correspondences.

Fig. 5 shows the quantitative result of each LiDAR scan pair (the Courtyard and Facade datasets contain many pairs and only partial of them are displayed). As shown, RANSAC-type methods except R-RANSAC can obtain good results on most of scan pairs of the Courtyard and Facade datasets. However, RANSAC-type methods are not suitable for cases with extreme amounts of outliers. None of them perform well on the Arch and Trees datasets. Their rotation errors are even larger than 100°. Again, the model estimation of FGR is sensitive to heavy outliers. S4PCS is comparable with RANSAC-variants. It is even better than LO-RANSAC and FLO-RANSAC on the Arch and Office datasets. Essentially, 4PCS and its variants are also RANSAC-based, since their main framework is based on a similar hypothesize-and-verify technique used in RANSAC. K4PCS performs worse than S4PCS. S4PCS uses point clouds as input while K4PCS uses ISS keypoints. Thus, the reason may be that the keypoints are too sparse to find accurate and reliable 4PCS bases. PointNetLK gets the worst results on the ETH dataset, which cannot

[4] http://www.prs.igp.ethz.ch/research/completed_projects/automatic_registration_of_point_clouds.html

[5] ISS and FPFH are implemented based on PCL: http://pointclouds.org/

Table 5
Average translation error δ_t [m] results on the ETH dataset.

Method	Arch	Courtyard	Facade	Office	Trees
RANSAC	17.07	2.31	0.06	2.44	9.05
R-RANSAC	13.53	12.44	3.36	7.67	15.66
LO-RANSAC	17.09	1.90	0.03	2.73	8.50
FLO-RANSAC	17.10	2.05	0.03	3.67	9.21
FGR	10.18	2.30	0.91	3.53	8.67
S4PCS	1.02	0.34	0.32	0.13	3.21
K4PCS	11.57	0.59	3.05	0.09	5.54
PointNetLK	23.29	19.64	10.89	9.06	10.09
Our GESAC	**0.04**	**0.03**	**0.01**	**0.03**	**0.05**

Table 6
Average runtime [s] results on the ETH dataset.

Method	Arch	Courtyard	Facade	Office	Trees
RANSAC	239.82	317.72	194.72	352.96	314.93
R-RANSAC	201.82	127.23	112.68	95.47	204.62
LO-RANSAC	226.93	318.12	197.11	364.69	302.92
FLO-RANSAC	232.91	317.96	197.70	360.90	316.19
FGR	47.93	91.72	38.35	44.39	55.22
S4PCS	813.05	705.58	737.55	1000	1000
K4PCS	775.66	388.03	1000	89.95	1000
PointNetLK	**0.58**	0.56	0.50	**0.52**	**0.58**
Our GESAC	1.71	**0.11**	**0.25**	0.71	1.77

successfully register any scan pairs. The reason is that the generalization ability of PointNetLK is poor. The model trained on the ModelNet40 can not be adapted in the ETH dataset, because the ModelNet40 is a small-scale dataset (single object in a scan) while the ETH dataset is captured in large-scale outdoor scenes. In contrast, our proposed GESAC achieves the best performance, i.e., both the rotation errors and translation errors are the lowest on most of the pairs. Even in extremely difficult cases (e.g., the Arch and Trees datasets), GESAC still obtains very good registration accuracy. The largest rotation error and translation error of GESAC are only 0.60° and 0.19 m, respectively.

In GESAC, we design a two-stage outlier filtering strategy (equal-length constraint and point set graph matching), which is able to identify good subsets. The hypothesize-and-verify framework and robust cost ensure that satisfactory solutions can be found. These make the proposed GESAC is robust. In addition, GESAC adapts a shape-annealing estimate, which gives large weights to small residuals while giving small weights to large residuals. It is thus less sensitive to noise, especially non-Gaussian noise. Thus, it has a higher model fitting accuracy than traditional least-squares that is used in RANSAC. Fig. 6 shows the complete registration result of GESAC on each scan category.

Table 4 and Table 5 summarize the average rotation error and translation error of each category, respectively. The rotation errors of RANSAC-type methods except R-RANSAC on the Facade dataset is less than 1°. S4PCS and K4PCS perform better than RANSAC on the Courtyard and Office datasets. The proposed GESAC can almost be directly adopted in practical applications without any post-processing such as fine registration. Even on the extremely difficult cases with 99.64% of outliers (the Trees dataset), GESAC still achieves a high registration accuracy, i.e., 0.21° and 0.05 m, which is much superior to others. The success rate (the criterion is $\delta_{\mathbf{R}} < 2^\circ$ and $\delta_t < 1\text{m}$) of GESAC (100%) is much higher than the ones of RANSAC (54.67%), R-RANSAC (22.67%), LO-RANSAC (64%), FLO-RANSAC (60%), FGR (20%), S4PCS (78.67%), K4PCS (66.67%), and PointNetLK (0%).

The running times are reported in Table 6. Note that the reported values do not contain the running time of feature extraction. From the table, we can see that the proposed GESAC is very efficient. It is the fastest among the CPU based methods. It is 2800 + and 770 + times faster than RANSAC on the Courtyard and Facade, respectively. Even on the Arch, our method is still 139 + times faster than RANSAC. Although FGR, S4PCS, and K4PCS are implemented by C++, GESAC is still two to four orders of magnitude faster than them. PointNetLK is the fastest since it is a GPU implementation. We also test the CPU implementation of PointNetLK, which costs about 10 s on the ETH dataset with a much better CPU (CPU Core i7-7800X @ 3.50 GHz). Actually, the computation of GESAC can be sped up several orders of magnitude with a GPU implementation as it is highly parallelizable.

4.4. Limitations

The limitations of the proposed GESAC are twofold: (1) the equal-length constraint holds only if point clouds do not have scale differences, i.e., the proposed two-stage filtering strategy is only suitable for cases with rotation and translation changes. Therefore, the proposed GESAC is not suitable for registration of point clouds with unknown scales or non-rigid transformations. For example, our GESAC can not be directly applied to register a photogrammetric point cloud and a LiDAR point cloud. The photogrammetric point cloud should be scaled to have the same metric with the LiDAR point cloud before registration. (2) our GESAC is designed specifically for PCR problem. It is not a general method like RANSAC which can also be applied to other robust estimation problems, such as line fitting. Moreover, if the number of true inliers in initial correspondence set is too low (such as < 10), our GESAC may fail.

5. Conclusions

In this paper, we develop an accurate, fast, and robust correspondence-based coarse registration method for point cloud registration (PCR). The proposed method named GESAC can be regarded as a RANSAC variant. Compared with current RANSAC-type methods, we use a much larger subset instead of a minimal subset for model estimation and allow outliers in the subset. This is substantially different from current RANSAC variants, since they only regard outlier-free subsets as good ones. GESAC has a much higher probability to obtain good subsets than RANSAC variants. It only requires 1/4000 of number of trials that is required by RANSAC. We design a two-stage filtering strategy to identify good subsets. Specifically, an equal-length constraint is first presented to filter "degraded" subsets; then, a max-pooling graph matching method is adapted to remove potential outliers in the "non-degraded" subsets. Once a potential good subset is obtained, GESAC uses

a shape-annealing robust estimate instead of classic least-squares for rigid transformation estimation. These three improvements ensure the high robustness of GESAC. Extensive simulated and real experiments demonstrate that GESAC is much superior to current methods, i.e., it is able to tolerate extremely high outlier rates (e.g., higher than 99%) and is several orders of magnitude faster than RANSAC-variants at high outlier rates.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) (No. 41901398), Natural Science Foundation of Hubei Province (No. 2019CFB167), State Key Laboratory of Rail Transit Engineering Informatization(FSDI) (No. SKLK19-06), and China Postdoctoral Science Foundation (No. 2018M640734).

References

Agarwal, S., Lim, J., Zelnik-Manor, L., Perona, P., Kriegman, D., Belongie, S., 2005. Beyond pairwise clustering. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 2, pp. 838–845.
Aoki, Y., Goforth, H., Srivatsan, R.A., Lucey, S., 2019. Pointnetlk: Robust & efficient point cloud registration using pointnet. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7163–7172.
Araújo, A.M., Oliveira, M.M., 2020. A robust statistics approach for plane detection in unorganized point clouds. Pattern Recogn. 100, 107115.
Barron, J.T., 2019. A general and adaptive robust loss function. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4331–4339.
Berner, A., Bokeloh, M., Wand, M., Schilling, A., Seidel, H.-P., 2008. A graph-based approach to symmetry detection. In: Volume Graphics, volume 40, pp. 1–8.
Besl, P., McKay, N.D., 1992. A method for registration of 3-d shapes. IEEE Trans. Pattern Anal. Mach. Intell. 14, 239–256.
Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., Rother, C., 2017. Dsac-differentiable ransac for camera localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6684–6692.
Cai, Z., Chin, T.-J., Bustos, A.P., Schindler, K., 2019. Practical optimal registration of terrestrial lidar scan pairs. ISPRS J. Photogram. Remote Sens. 147, 118–131.
Chen, H., Bhanu, B., 2007. 3d free-form object recognition in range images using local surface patches. Pattern Recogn. Lett. 28, 1252–1262.
Chetverikov, D., Stepanov, D., Krsek, P., 2005. Robust euclidean alignment of 3d point sets: the trimmed iterative closest point algorithm. Image Vis. Comput. 23, 299–309.
Chin, T.-J., Suter, D., 2017. The maximum consensus problem: recent algorithmic advances. Synth. Lect. Comput. Vision 7, 1–194.
Cho, M., Lee, J., Lee, K.M., 2010. Reweighted random walks for graph matching. In: European conference on Computer vision, pp. 492–505.
Cho, M., Sun, J., Duchenne, O., Ponce, J., 2014. Finding matches in a haystack: A max-pooling strategy for graph matching in the presence of outliers. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2083–2090.
Cho, M., Sun, J., Duchenne, O., Ponce, J., 2014. Finding matches in a haystack: A max-pooling strategy for graph matching in the presence of outliers. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2091–2098.
Choi, J., Medioni, G., 2009. Starsac: Stable random sample consensus for parameter estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 675–682.
Chum, O., Matas, J., Kittler, J., 2003. Locally optimized ransac. In: Joint Pattern Recognition Symposium, pp. 236–243.
Cour, T., Srinivasan, P., Shi, J., 2007. Balanced graph matching. In: Advances in Neural Information Processing Systems, pp. 313–320.
Dong, Z., Yang, B., Liang, F., Huang, R., Scherer, S., 2018. Hierarchical registration of unordered tls point clouds based on binary shape context descriptor. ISPRS J. Photogram. Remote Sens. 144, 61–79.
Egozi, A., Keller, Y., Guterman, H., 2012. A probabilistic approach to spectral graph matching. IEEE Trans. Pattern Anal. Mach. Intell. 35, 18–27.
Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24, 381–395.
Gojcic, Z., Zhou, C., Wegner, J.D., Wieser, A., 2019. The perfect match: 3d point cloud matching with smoothed densities. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5545–5554.
Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., Kwok, N.M., 2016. A comprehensive performance evaluation of 3d local feature descriptors. Int. J. Comput. Vision 116, 66–89.
Holland, P.W., Welsch, R.E., 1977. Robust regression using iteratively reweighted least-squares. Commun. Stat. - Theory and Methods 6, 813–827.
Horn, B.K.P., Hilden, H.M., Negahdaripour, S., 1988. Closed-form solution of absolute orientation using orthonormal matrices. J. Opt. Soc. Am. A 5, 1127–1135.
Johnson, A.E., Hebert, M., 1999. Using spin images for efficient object recognition in cluttered 3d scenes. IEEE Trans. Pattern Anal. Mach. Intell. 21, 433–449.
Lebeda, K., Matas, J., Chum, O., 2012. Fixing the locally optimized ransac–full experimental evaluation. In: British Machine Vision Conference, pp. 1–11.
Leordeanu, M., Hebert, M., 2005. A spectral technique for correspondence problems using pairwise constraints. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1482–1489.
Leordeanu, M., Hebert, M., Sukthankar, R., 2009. An integer projected fixed point method for graph matching and map inference. In: Advances in Neural Information Processing Systems, pp. 1114–1122.
Li, J., Hu, Q., Ai, M., 2017. Robust feature matching for geospatial images via an affine-invariant coordinate system. Photogram. Rec. 32, 317–331.
Li, J., Hu, Q., Ai, M., 2019a. Lam: Locality affine-invariant feature matching. ISPRS J. Photogram. Remote Sens. 154, 28–40.
Li, J., Hu, Q., Ai, M., 2019b. Rift: Multi-modal image matching based on radiation-variation insensitive feature transform. IEEE Trans. Image Process.
Li, J., Hu, Q., Ai, M., 2020. Robust geometric model estimation based on scaled welsch q-norm. IEEE Trans. Geosci. Remote Sens.
Li, J., Zhong, R., Hu, Q., Ai, M., 2016. Feature-based laser scan matching and its application for indoor mapping. Sensors 16, 1265.
Liu, H., Latecki, L.J., Yan, S., 2010. Robust clustering as ensembles of affinity relations. In: Advances in Neural Information Processing Systems, pp. 1414–1422.
Liu, H., Yan, S., 2012. Efficient structure detection via random consensus graph. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp. 574–581.
Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60, 91–110.
Matas, J., Chum, O., 2004. Randomized ransac with td, d test. Image Vis. Comput. 22, 837–842.
Mellado, N., Aiger, D., Mitra, N.J., 2014. Super 4pcs fast global pointcloud registration via smart indexing. Comput. Graph. Forum 33, 205–215.
Pham, T.T., Chin, T.-J., Yu, J., Suter, D., 2014. The random cluster model for robust geometric fitting. IEEE Trans. Pattern Anal. Mach. Intell. 36, 1658–1671.
Raguram, R., Chum, O., Pollefeys, M., Matas, J., Frahm, J.-M., 2012. Usac: a universal framework for random sample consensus. IEEE Trans. Pattern Anal. Mach. Intell. 35, 2022–2038.
Rusu, R.B., Blodow, N., Beetz, M., 2009. Fast point feature histograms (fpfh) for 3d registration. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 3212–3217.
Salti, S., Tombari, F., Di Stefano, L., 2014. Shot: Unique signatures of histograms for surface and texture description. Comput. Vis. Image Underst. 125, 251–264.
Suwajanakorn, S., Snavely, N., Tompson, J.J., Norouzi, M., 2018. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In: Advances in Neural Information Processing Systems, pp. 2059–2070.
Tennakoon, R.B., Bab-Hadiashar, A., Cao, Z., Hoseinnezhad, R., Suter, D., 2015. Robust model fitting using higher than minimal subset sampling. IEEE Trans. Pattern Anal. Mach. Intell. 38, 350–362.
Theiler, P.W., Wegner, J.D., Schindler, K., 2014. Keypoint-based 4-points congruent sets–automated marker-less registration of laser scans. ISPRS J. Photogram. Remote Sens. 96, 149–163.
Tombari, F., Salti, S., Di Stefano, L., 2013. Performance evaluation of 3d keypoint detectors. Int. J. Comput. Vision 102, 198–220.
Tordoff, B.J., Murray, D.W., 2005. Guided-mlesac: Faster image transform estimation by using matching priors. IEEE Trans. Pattern Anal. Mach. Intell. 27, 1523–1535.
Torr, P.H., Zisserman, A., 2000. Mlesac: A new robust estimator with application to estimating image geometry. Comput. Vis. Image Underst. 78, 138–156.
Wang, T., Ling, H., Lang, C., Feng, S., 2017. Graph matching with adaptive and branching path following. IEEE Trans. Pattern Anal. Mach. Intell. 40, 2853–2867.
Wu, J., Shen, H., Li, Y.-D., Xiao, Z.-B., Lu, M.-Y., Wang, C.-L., 2013. Learning a hybrid similarity measure for image retrieval. Pattern Recogn. 46, 2927–2939.
Xu, Y., Hoegner, L., Tuttas, S., Stilla, U., 2017. Voxel- and graph-based point cloud segmentation of 3d scenes using perceptual grouping laws. ISPRS Annals Photogram., Remote Sens. Spatial Inform. Sci., 4.
Yang, H., Carlone, L., 2019. A polynomial-time solution for robust registration with extreme outlier rates. arXiv preprint arXiv:1903.08588.
Yu, J., Zheng, H., Kulkarni, S.R., Poor, H., 2010. Two-stage outlier elimination for robust curve and surface fitting. EURASIP J. Adv. Signal Process. 2010, 154891.
Zaharescu, A., Boyer, E., Varanasi, K., Horaud, R., 2009. Surface feature detection and description with applications to mesh matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 373–380.
Zass, R., Shashua, A., 2008. Probabilistic graph and hypergraph matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.
Zhong, Y., 2009. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 689–696.
Zhou, Q.-Y., Park, J., Koltun, V., 2016. Fast global registration. In: European Conference on Computer Vision, pp. 766–782.