2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

ROBIN: a Graph-Theoretic Approach to Reject Outliers in Robust Estimation using Invariants

Jingnan Shi, Heng Yang, Luca Carlone

Abstract— **Many estimation problems in robotics, computer vision, and learning require estimating unknown quantities in the face of outliers. Outliers are typically the result of incorrect data association or feature matching, and it is not uncommon to have problems where more than 90% of the measurements used for estimation are outliers. While current approaches for robust estimation (*e.g.,* RANSAC or graduated non-convexity) are able to deal with moderate amounts of outliers, they fail to produce accurate estimates in the presence of many outliers. This paper develops an approach to prune outliers. First, we develop a theory of *invariance* that allows us to quickly check if a subset of measurements are mutually compatible without explicitly solving the corresponding estimation problem. Second, we develop a graph-theoretic framework, where measurements are modeled as vertices and mutual compatibility is captured by edges in a graph. We generalize existing results showing that the inliers form a clique in this *compatibility graph* and typically belong to the *maximum clique*. We also show that in practice the *maximum k-core* of the compatibility graph provides an approximation of the maximum clique, while being much faster to compute in large problems. The combination of these two contributions leads to ROBIN, our approach to Reject Outliers Based on INvariants, which allows us to quickly prune outliers in generic estimation problems. We demonstrate ROBIN in four geometric perception problems and show it boosts robustness of existing solvers (making them robust to more than 95% outliers), while running in milliseconds in large problems.**

I. Introduction

Robust estimation is concerned with estimating an unknown variable $\boldsymbol{x}$ given noisy and potentially corrupted measurements. For instance, in 3D perception problems, the $\boldsymbol{x}$ may be the pose of an unknown object (*e.g.,* a car) while the measurements are features of the object (*e.g.,* wheels, headlights) detected in the sensor data (*e.g.,* images, lidar scans) [60], [45]. In Simultaneous Localization and Mapping (SLAM), $\boldsymbol{x}$ might denote the trajectory of the robot and the location of landmarks. In machine learning, the $\boldsymbol{x}$ can be a description of a model regressing the data. In all these problems, one has to estimate the state $\boldsymbol{x} \in \mathbb{X}$ (where $\mathbb{X}$ is the domain of $\boldsymbol{x}$, for instance the set of 3D poses) given measurements $\boldsymbol{y}_i$ and a measurement model $\boldsymbol{h}_i$ that describes how the measurements are related to the unknown state $\boldsymbol{x}$:

$$\boldsymbol{y}_i = \boldsymbol{h}_i(\boldsymbol{x}, \boldsymbol{\epsilon}_i), \qquad i = 1, \dots, N \tag{1}$$

where $\boldsymbol{\epsilon}_i$ denotes the measurement noise. In practical applications, $\boldsymbol{y}_i$'s are the result of the preprocessing of raw sensor data: for instance, in 3D perception problems, they are the output of image-based or lidar-based feature detection and matching. This preprocessing, often referred to as the *perception front-end*, is prone to producing many corrupted measurements (*i.e., outliers*) for which the error $\boldsymbol{\epsilon}_i$ is so large that $\boldsymbol{y}_i$ becomes uninformative towards the estimation of $\boldsymbol{x}$. For instance, traditional and deep learning techniques for feature matching in object pose estimation problems [43], [62] are not uncommon to produce more than 90% outliers.

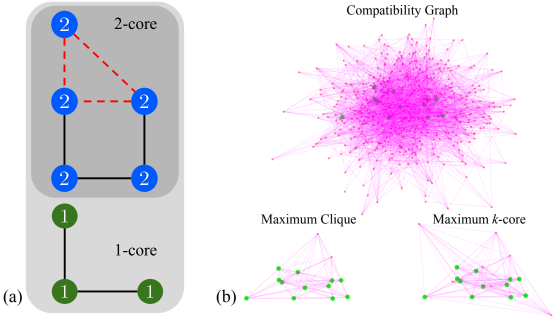

Fig. 1. We propose ROBIN, a graph-theoretic approach to reject outliers in robust estimation problems. We use *invariants* to check compatibility among subsets of measurements and describe the results of these checks using a *compatibility graph* $\mathcal{G}$. ROBIN rejects outliers by computing a maximum clique or a maximum k-core of $\mathcal{G}$. (a) Maximum clique (dotted red edges) and k-core ($k = 1, 2$) in a toy graph (numbers denote the *core number* of each vertex). (b) Compatibility graph for a single rotation averaging problem with 500 measurements and only 12 inliers (green dots).

Research in robust estimation across robotics, computer vision, and learning has made substantial progress towards solving estimation problems with outliers [31], [12], [17], [54], [9]. Despite this progress, the literature is still divided between *fast heuristics* and *globally-optimally solvers*. Fast heuristics, such as RANSAC [26] and variants [53], [20], [21], [6], [55], [47], graduated non-convexity (GNC) [57], [10], [11], or iterative local solvers for M-estimation [1], [51] are fast but prone to fail in the presence of many outliers [43], [62], [48]. On the other hand, globally-optimally solvers, such as Branch and Bound [32], [63], [43], and combinatorial approaches for *maximum consensus* [19] can tolerate extreme outlier rates, but run in worst-case exponential time and are slow in practice. This dichotomy has been resolved only in specific problems, *e.g.,* point cloud registration, where recent approaches like TEASER++ [62] are robust to more than 99% outliers and run in milliseconds. The goal of this paper is to provide a general framework (which indeed generalizes our approach in [62]) to boost the performance of existing techniques, including RANSAC and GNC, to be robust to more than 95% outliers while preserving their real-time operation.

Contribution. We address the following key question: *Can we quickly detect and discard many outliers in the set of measurements* (1), *while preserving all the inliers?*

J. Shi, H. Yang, and L. Carlone are with the Laboratory for Information & Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA 02139, USA, Email: {jnshi,hankyang,lcarlone}@mit.edu

This work was partially funded by ARL DCIST CRA W911NF-17-2-0181, ONR RAIDER N00014-18-1-2828, Lincoln Laboratory's Resilient Perception in Degraded Environments program, and NSF CAREER award "Certifiable Perception for Autonomous Cyber-Physical Systems".

978-1-7281-9077-8/21/$31.00 ©2021 IEEE

Intuitively, we want to bring down the number of outliers into a range where existing solvers (RANSAC, GNC) work well (*e.g.,* below 60%). Towards this goal, we provide two contributions. The first contribution, presented in Section III, is a theory of *invariance* that allows us to check if a subset of $n \ll N$ measurements are mutually compatible; contrary to RANSAC, our check does not require computing an estimate for $\boldsymbol{x}$ so it is extremely fast. The second contribution, in Section IV, is a graph-theoretic framework, where vertices represent measurements and edges represent mutual compatibility. We generalize existing results [39], [62], [25] showing that, even for $n > 2$ (related work focuses on pairwise compatibility, where $n = 2$), the inliers form a clique in this *compatibility graph* and can be typically retrieved by computing the *maximum clique*. Despite the availability of fast algorithms, finding a maximum clique is NP-hard and its runtime is often incompatible with real-time robotics applications. Therefore, we propose a faster alternative by computing the maximum *k-core* of the graph. The combination of these contributions leads to ROBIN, our approach to Reject Outliers Based on INvariants. We summarize ROBIN in Section V, while in Section VI we demonstrate it in four perception problems (single rotation averaging, point cloud registration, point-with-normal registration, and 2D-3D camera pose estimation) and show that (i) ROBIN boosts robustness of existing robust solvers, making them robust to more than 95% outliers, and (ii) it runs in milliseconds in large-scale simulated and real datasets.

Extra results, visualizations, and proofs are given in the supplementary material [52] and the video attachment.

II. Related Work

Consensus Maximization is a framework for robust estimation that aims to find the largest set of measurements with errors below a user-defined threshold. Consensus maximization is NP-hard [2], [17], hence the community has resorted to randomized approaches, such as RANSAC [26]. RANSAC repeatedly draws a minimal subset of measurements to compute an estimate, and stops after finding an estimate that agrees with a large set of measurements. While RANSAC works well for problems where the minimal set is small and there are not many outliers, the number of iterations increases exponentially with the percentage of outliers [48], making it impractical for problems with many outliers. On the other hand, global solvers, such as branch-and-bound (BnB) [38] and tree search [18], exist but scale poorly with the problem size, with BnB being exponential in the size of $\boldsymbol{x}$, and tree search being exponential in the number of outliers [15].

M-estimation performs estimation by optimizing a robust cost function that reduces the influence of the outliers. The resulting problems are typically optimized using iterative local solvers. MacTavish and Barfoot [54] compare several robust cost functions for visual-inertial localization using iterative re-weighted least squares solvers. The downside of local solvers is that they need a good initial guess, which is often unavailable in practice. A popular approach to circumvent the need for an initial guess is *Graduated Non-Convexity* (GNC) [11], [10]. Zhou *et al.* [65] use GNC for point cloud registration. Yang *et al.* [57] and Antonante *et al.* [2] combine GNC with global non-minimal solvers and show their general applicability to problems with up to 80% outliers.

For certain low-dimensional geometric problems, fast global solvers exist. Enqvist *et al.* [24] use a *truncated least squares* (TLS) cost to solve triangulation problems in polynomial time in the number of measurements, but exponential time in the dimension of $\boldsymbol{x}$. Ask *et al.* [4] use a TLS cost for image registration. Recently, *certifiably-robust* globally-optimal solvers based on *convex relaxations* have been used for *M-estimation* [35], [62], [59]. Unfortunately, due to the poor scalability of current SDP solvers, such methods are mostly viable to *check* optimality [62], [61].

Graph algorithms have been extensively used in robot perception and vision, from SLAM [14], [23], [49] to human-pose estimation [13]. Neira and Tardós propose compatibility tests and BnB for data association in SLAM [41]. Bailey *et al.* [5] propose a Maximum Common Subgraph algorithm for feature matching in lidar scans. Graph algorithms, such as graph-cut, have been used in RANSAC to improve performance against noise and outliers [7]. In [46], graph cliques are used to segment objects under rigid body motion. Leordeanu and Hebert [37] establish image matches by finding strongly-connected clusters in the correspondence graph with an approximate spectral method. Enqvist *et al.* [25] develop an outlier rejection algorithm for 3D-3D and 2D-3D registration based on approximate vertex cover. Recently, progress in graph algorithms (see [50], [44]) have allowed for the emergence of fast graph-based algorithms, such as TEASER++ [62], that are robust to high-noise and outliers.

III. Measurement Invariants

This section develops a theory of invariance that allows checking if a subset of measurements contains outliers without explicitly computing an estimate for $\boldsymbol{x}$. We will then use these checks to select large sets of mutually compatible measurements and reject outliers in Section IV.

A. From Measurements to Invariants

Let us consider the measurements in eq. (1) and denote the indices of the measurements as $\mathcal{Y} \doteq \{1, \dots, N\}$. For a given integer $n \leq N$, let $\mathcal{S} \subset \mathcal{Y}$ be a subset of n indices in $\mathcal{Y}$, and denote with $\mathcal{S}_j$ the j-th element of this subset (with $j = 1, \dots, n$). Then, we use the following notation:

$$\boldsymbol{y}_{\mathcal{S}} = \begin{bmatrix} \boldsymbol{y}_{\mathcal{S}_1} \\ \boldsymbol{y}_{\mathcal{S}_2} \\ \vdots \\ \boldsymbol{y}_{\mathcal{S}_n} \end{bmatrix}, \quad \boldsymbol{h}_{\mathcal{S}} = \begin{bmatrix} \boldsymbol{h}_{\mathcal{S}_1} \\ \boldsymbol{h}_{\mathcal{S}_2} \\ \vdots \\ \boldsymbol{h}_{\mathcal{S}_n} \end{bmatrix}, \quad \boldsymbol{\epsilon}_{\mathcal{S}} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathcal{S}_1} \\ \boldsymbol{\epsilon}_{\mathcal{S}_2} \\ \vdots \\ \boldsymbol{\epsilon}_{\mathcal{S}_n} \end{bmatrix} \tag{2}$$

which is simply stacking together measurements $\boldsymbol{y}_i$, functions $\boldsymbol{h}_i$, and noise $\boldsymbol{\epsilon}_i$ for the subset of measurements $i \in \mathcal{S}$.

We are now ready to define an *n-invariant*.

Definition 1 (n-Invariant): Given measurements (1), a function $\boldsymbol{f}$ is called an *n-invariant* if for all $\mathcal{S} \subset \mathcal{Y}$ of size n: $\boldsymbol{f}(\boldsymbol{y}_{\mathcal{S}}) = \boldsymbol{f}(\boldsymbol{h}_{\mathcal{S}}(\bar{\boldsymbol{x}}, \boldsymbol{\delta}_{\mathcal{S}}))$, $\forall \bar{\boldsymbol{x}} \in \mathbb{X}$, where $\boldsymbol{\delta}$ is a noise term whose statistics do not depend on $\bar{\boldsymbol{x}}$, and $\boldsymbol{\delta} = \mathbf{0}$ whenever $\boldsymbol{\epsilon} = \mathbf{0}$ in (1) (*i.e.,* when measurements are noiseless).

Intuitively, the invariant function $\boldsymbol{f}$ takes a subset of measurements and computes a quantity that no longer depends on $\boldsymbol{x}$ (hence the equation is satisfied for all $\boldsymbol{x} \in \mathbb{X}$). When $n = 2$ in Definition 1, we obtain a *pairwise invariant.*

B. Example of invariants

Here we provide examples of invariants for four estimation problems in robotics and vision: single rotation averaging, point cloud registration, point-with-normal registration, and 2D-3D camera pose estimation.

Single rotation averaging [28] seeks to find an unknown rotation $\boldsymbol{R} \in \mathrm{SO}(3)$ given noisy measurements $\boldsymbol{R}_i \in \mathrm{SO}(3)$:

$$\boldsymbol{R}_i = \boldsymbol{R} \cdot \mathrm{Exp}(\boldsymbol{\epsilon}_i) \tag{3}$$

where $\boldsymbol{\epsilon}_i \in \mathbb{R}^3$ denotes measurement noise and Exp maps a 3D vector to a rotation matrix [8, §7.1.3]. Eq. (3) has the same form of (1), with $\boldsymbol{y}_i \doteq \boldsymbol{R}_i$, $\boldsymbol{x} \doteq \boldsymbol{R}$, and $\boldsymbol{h}_i(\boldsymbol{R}, \boldsymbol{\epsilon}_i) \doteq \boldsymbol{R} \cdot \mathrm{Exp}(\boldsymbol{\epsilon}_i)$. The problem admits a simple pairwise invariant.

Proposition 2 (Pairwise Invariant for Rotation Averaging): The function $\boldsymbol{f}(\boldsymbol{R}_i, \boldsymbol{R}_j) \doteq \boldsymbol{R}_i^{\mathsf{T}} \boldsymbol{R}_j$, where $\boldsymbol{R}_i, \boldsymbol{R}_j$ are a pair of measurements, is an invariant for (3) and satisfies:

$$\boldsymbol{f}(\boldsymbol{R}_i, \boldsymbol{R}_j) \doteq \boldsymbol{R}_i^{\mathsf{T}} \boldsymbol{R}_j = \mathrm{Exp}(\boldsymbol{\delta}_i)^{\mathsf{T}} \mathrm{Exp}(\boldsymbol{\delta}_j) \tag{4}$$

with $\boldsymbol{\delta}_i = \boldsymbol{\epsilon}_i$ and $\boldsymbol{\delta}_j = \boldsymbol{\epsilon}_j$, for any rotation $\boldsymbol{R}$ in (3). □

The proof is trivial and is given in [52] for completeness.

Point cloud registration [3], [30] consists in finding a rigid body transformation $(\boldsymbol{R}, \boldsymbol{t})$ (where $\boldsymbol{R} \in \mathrm{SO}(3)$ and $\boldsymbol{t} \in \mathbb{R}^3$) that aligns two sets of 3D points $\boldsymbol{a}_i \in \mathbb{R}^3$ and $\boldsymbol{b}_i \in \mathbb{R}^3$, with $i = 1, \ldots, N$. The corresponding measurement model can again be phrased in the form of (1):

$$\boldsymbol{b}_i = \boldsymbol{R}\boldsymbol{a}_i + \boldsymbol{t} + \boldsymbol{\epsilon}_i \tag{5}$$

where $\boldsymbol{\epsilon}_i \in \mathbb{R}^3$ is the measurement noise.

Proposition 3 (Pairwise Invariant for Point Registration): Assuming the noise $\boldsymbol{\epsilon}_i$ is isotropic,[1] the function $\boldsymbol{f}(\boldsymbol{b}_i, \boldsymbol{b}_j) \doteq \|\boldsymbol{b}_j - \boldsymbol{b}_i\|$, where $\boldsymbol{b}_i, \boldsymbol{b}_j$ are a pair of measurements, is an invariant for (5) and satisfies:

$$\boldsymbol{f}(\boldsymbol{b}_i, \boldsymbol{b}_j) \doteq \|\boldsymbol{b}_j - \boldsymbol{b}_i\| = \|\boldsymbol{a}_j - \boldsymbol{a}_i + \boldsymbol{\delta}_j - \boldsymbol{\delta}_i\| \tag{6}$$

for any transformation $(\boldsymbol{R}, \boldsymbol{t})$ between the points in (5), where $\boldsymbol{\delta}_i$ (resp. $\boldsymbol{\delta}_j$) has the same distribution of $\boldsymbol{\epsilon}_i$ (resp. $\boldsymbol{\epsilon}_j$). □

The proposition formalizes the intuition that the distance between points in a point cloud is invariant to rigid transformations and has been used to detect outliers in [25], [58], [42]. Note that in this problem (and in the ones below), each measurement encodes an association between two points in the two point clouds and for this reason the measurements are often referred to as *correspondences* or *putative matches*.

Point-with-normal registration is similar to point-cloud registration in that it looks for a rigid body transformation $(\boldsymbol{R}, \boldsymbol{t})$ that aligns 3D points, except the 3D points $\boldsymbol{a}_i$ (resp. $\boldsymbol{b}_i$) have normals $\boldsymbol{n}_i^a$ (resp. $\boldsymbol{n}_i^b$), with $\boldsymbol{n}_i^a, \boldsymbol{n}_i^b \in \mathbb{R}^3$ and $\|\boldsymbol{n}_i^a\| = \|\boldsymbol{n}_i^b\| = 1$. This results in the measurement model:

$$\boldsymbol{b}_i = \boldsymbol{R}\boldsymbol{a}_i + \boldsymbol{t} + \boldsymbol{\epsilon}_i, \qquad \boldsymbol{n}_i^b = \boldsymbol{R} \cdot \mathrm{Exp}(\boldsymbol{\nu}_i) \cdot \boldsymbol{n}_i^a \tag{7}$$

where $\boldsymbol{\nu}_i \in \mathbb{R}^3$ is noise perturbing the measured normals.

Proposition 4 (Pairwise Invariant for P&N Registration): Assuming the noise $\boldsymbol{\epsilon}_i$ is isotropic, the function $\boldsymbol{f}(\boldsymbol{y}_i, \boldsymbol{y}_j) \doteq [\|\boldsymbol{b}_j - \boldsymbol{b}_i\|, (\boldsymbol{n}_i^b)^{\mathsf{T}} \boldsymbol{n}_j^b]^{\mathsf{T}}$, where $\boldsymbol{y}_i \doteq [(\boldsymbol{b}_i)^{\mathsf{T}}, (\boldsymbol{n}_i^b)^{\mathsf{T}}]^{\mathsf{T}}$ and $\boldsymbol{y}_j \doteq [(\boldsymbol{b}_j)^{\mathsf{T}}, (\boldsymbol{n}_j^b)^{\mathsf{T}}]^{\mathsf{T}}$ are a pair of measurements in (7), is an invariant for (7) and satisfies:

$$\boldsymbol{f}(\boldsymbol{y}_i, \boldsymbol{y}_j) \doteq \begin{bmatrix} \|\boldsymbol{b}_j - \boldsymbol{b}_i\| \\ (\boldsymbol{n}_i^b)^{\mathsf{T}} \boldsymbol{n}_j^b \end{bmatrix} = \begin{bmatrix} \|\boldsymbol{a}_j - \boldsymbol{a}_i + \boldsymbol{\delta}_j^p - \boldsymbol{\delta}_i^p\| \\ (\boldsymbol{n}_i^a)^{\mathsf{T}} \mathrm{Exp}(\boldsymbol{\delta}_i^n)^{\mathsf{T}} \mathrm{Exp}(\boldsymbol{\delta}_j^n) \boldsymbol{n}_j^a \end{bmatrix} \tag{8}$$

for any rigid transformation $(\boldsymbol{R}, \boldsymbol{t})$ in (7), where $\boldsymbol{\delta}_i^p$ has the same distribution of $\boldsymbol{\epsilon}_i$, and $\boldsymbol{\delta}_i^n = \boldsymbol{\nu}_i$, $i = 1, \ldots, N$. □

Proposition 4 formalizes the intuition that the angle between normals, measured by the dot product $(\boldsymbol{n}_i^b)^{\mathsf{T}} \boldsymbol{n}_j^b$, is invariant to rigid transformations.

2D-3D camera pose estimation consists in finding the 3D pose of a camera $(\boldsymbol{R}, \boldsymbol{t})$ with respect to a 3D object of known shape from pixel projections of points of the object. Calling the pixel projections $\boldsymbol{y}_i \in \mathbb{R}^2$ and the corresponding 3D points $\boldsymbol{p}_i \in \mathbb{R}^3$, the measurement model becomes:

$$\boldsymbol{y}_i = \pi(\boldsymbol{R}, \boldsymbol{t}, \boldsymbol{p}_i) + \boldsymbol{\epsilon}_i \tag{9}$$

where $\pi(\cdot)$ is the standard perspective projection [29, Section 6.2], and $\boldsymbol{\epsilon}_i$ is the measurement noise. While it is known that there is no invariant for 3D points in generic configuration under perspective projection [40], there exist invariants for specific configurations of points (*e.g.,* collinear or coplanar points). For instance, we can design a 3-invariant capturing that projections of collinear 3D points are also collinear. Here we review another invariant for the case of collinear features, which is known as *cross ratio* in computer vision [29, p. 45]

Proposition 5 (4-invariant for Camera Pose): Given four collinear 3D points $\boldsymbol{p}_i \in \mathbb{R}^3$ $(i = 1, \ldots, 4)$ and the corresponding pixel projections $\boldsymbol{y}_i \in \mathbb{R}^2$, the *cross ratio* [29, p. 45] is a 4-invariant and satisfies:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3, \boldsymbol{y}_4) &\doteq \frac{\|\boldsymbol{y}_1 - \boldsymbol{\delta}_1 - \boldsymbol{y}_2 + \boldsymbol{\delta}_2\| \|\boldsymbol{y}_3 - \boldsymbol{\delta}_3 - \boldsymbol{y}_4 + \boldsymbol{\delta}_4\|}{\|\boldsymbol{y}_1 - \boldsymbol{\delta}_1 - \boldsymbol{y}_3 + \boldsymbol{\delta}_3\| \|\boldsymbol{y}_2 - \boldsymbol{\delta}_2 - \boldsymbol{y}_4 + \boldsymbol{\delta}_4\|} \\ &= \frac{\|(\boldsymbol{p}_1)^{\vee} - (\boldsymbol{p}_2)^{\vee}\| \|(\boldsymbol{p}_3)^{\vee} - (\boldsymbol{p}_4)^{\vee}\|}{\|(\boldsymbol{p}_1)^{\vee} - (\boldsymbol{p}_3)^{\vee}\| \|(\boldsymbol{p}_2)^{\vee} - (\boldsymbol{p}_4)^{\vee}\|} \end{aligned} \tag{10}$$

where $\boldsymbol{\delta}_i = \boldsymbol{\epsilon}_i$ $(i = 1, \ldots, 4)$, and for a 3D point $\boldsymbol{p} = [p_x \; p_y \; p_z]^{\mathsf{T}}$, we denoted $(\boldsymbol{p})^{\vee} \doteq [\frac{p_x}{p_z} \; \frac{p_y}{p_z}]^{\mathsf{T}}$. □

Remark 6 (Other Invariants): Invariants have been studied (typically in a noiseless setup) in the context of pattern recognition in computer vision. Early related work provides an abundance of resources to design invariants [27], that we discuss more broadly in the supplementary material [52]. Similar ideas have been also used in specific robotics applications, *e.g.,* SLAM [39], [16], without a systematic treatment.

C. From Invariants to Compatibility for Outlier Rejection

While the previous section developed invariants without distinguishing inliers from outliers, this section shows that the resulting invariants can be directly used to check if a subset of measurements contains an outlier. Towards this goal, we formalize the notion of inlier and outlier.

Definition 7 (Inliers and Outliers): Given measurements (1) and a threshold $\beta > 0$, a measurement i is called an *inlier* if the corresponding noise satisfies $\|\boldsymbol{\epsilon}_i\| \leq \beta$ and is called an *outlier* otherwise.

[1] Isotropic noise has a rotation-invariant distribution, for instance, a zero-mean Gaussian with covariance matrix equal to $\sigma^2 \mathbf{I}_3$.

Now we note that in the previous section the notion of invariants allowed us to obtain relations that depend on the measurements and the noise, but are independent on $\boldsymbol{x}$, see eqs. (4), (6), (8), (10). Therefore, we can directly use these relations to check if a subset of n measurements contains outliers: if eqs. (4), (6), (8), (10) are not satisfied by any choice of noise smaller than β, then the corresponding subset of measurements *must* contain an outlier. We call the corresponding check a *compatibility test*. In the following, we provide an example of compatibility test for point cloud registration. The reader can find the compatibility tests for the other applications in the supplementary material [52].

Compatibility test for point cloud registration. Eq. (6) states that any pair of measurements satisfies:

$$\|\boldsymbol{b}_j - \boldsymbol{b}_i\| = \|\boldsymbol{a}_j - \boldsymbol{a}_i + \boldsymbol{\delta}_j - \boldsymbol{\delta}_i\| \tag{11}$$

where $\boldsymbol{\delta}_i, \boldsymbol{\delta}_j$ have the same distribution of the measurement noise $\boldsymbol{\epsilon}_i, \boldsymbol{\epsilon}_j$. Therefore, one can check if one of the measurements $\boldsymbol{b}_i, \boldsymbol{b}_j$ is an outlier by checking if (11) can be satisfied by any choice of $\|\boldsymbol{\delta}_i\|, \|\boldsymbol{\delta}_j\| \leq \beta$. If we apply the triangle inequality to the right-hand-side of (11) we obtain:

$$-\|\boldsymbol{\delta}_j - \boldsymbol{\delta}_i\| \leq \|\boldsymbol{b}_j - \boldsymbol{b}_i\| - \|\boldsymbol{a}_j - \boldsymbol{a}_i\| \leq \|\boldsymbol{\delta}_j - \boldsymbol{\delta}_i\| \tag{12}$$

Now for two inliers we have $\|\boldsymbol{\delta}_i\| \leq \beta$ and $\|\boldsymbol{\delta}_j\| \leq \beta$, therefore $\|\boldsymbol{\delta}_j - \boldsymbol{\delta}_i\| \leq 2\beta$. Substituting in (12) we obtain:

$$-2\beta \leq \|\boldsymbol{b}_j - \boldsymbol{b}_i\| - \|\boldsymbol{a}_j - \boldsymbol{a}_i\| \leq 2\beta \tag{13}$$

Therefore, if $\boldsymbol{b}_i, \boldsymbol{b}_j$ are inliers they must satisfy (13), which is easy to check, and intuitively states that inliers associate pairs of points having similar distance in the two point clouds.

Generalizing this example, we obtain the following definition of *compatibility test*.

Definition 8 (Compatibility Test): Given a subset of n measurements and the corresponding n-invariant, a compatibility test is a binary condition involving the invariant, such that if the condition fails the set of measurements *must* contain at least one outlier.

Note that we require the test to be *sound* (*i.e.,* it does not detect outliers when testing a set of inliers), but may not be *complete* (*i.e.,* the test might pass even in the presence of outliers). This property is important since our goal is to prune as many outliers as we can, while preserving the inliers. Also note that the test detects if the set contains outliers, but does not provide information on *which* measurements are outliers. We are going to fill in this gap below.

IV. Compatibility Graph For Outlier Rejection

This section provides a graph-theoretic framework to use the compatibility tests introduced in Section III and point out which measurements are outliers. We start by defining the notion of *compatibility graph* and then show that inliers form a clique in this graph. We then propose the use of maximum clique and maximum k-core to find inliers.

A. From Compatibility Tests to Compatibility Graph

For a problem with an n-invariant, we describe the results of the compatibility tests for all subsets of n measurements using a *compatibility graph.* Formally, we define the compatibility graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ as an undirected graph, where each vertex in the vertex set $\mathcal{V}$ is associated to a measurement in (1) and an edge (i, j) in the edge set $\mathcal{E}$ is present in the graph if measurements i and j were involved in *at least one* of the subsets of measurements that passed the compatibility tests. Building the compatibility graph requires looping over all subsets of n measurements and, whenever the subset passes the compatibility test, adding edges among the corresponding n nodes in the graph. Note that these checks are very fast and easy to parallelize since they only involve checking boolean conditions (*e.g.,* (13)) without computing an estimate (as opposed to RANSAC).

B. Inlier Structures in Compatibility Graphs

This section shows that we can prune many outliers in (1) by computing the maximum clique of the compatibility graph. Let us start with some definitions.

Definition 9 (Maximum Clique): A clique of a graph $\mathcal{G}$ is a set of vertices such that any pair of vertices is connected by an edge in $\mathcal{G}$. The *maximum* clique of $\mathcal{G}$ is the clique with the largest number of vertices. The number of vertices $\omega(\mathcal{G})$ in the maximum clique is called the *clique number* of $\mathcal{G}$.

Given a compatibility graph $\mathcal{G}$, the following result relates the set of inliers in (1) with cliques in $\mathcal{G}$.

Theorem 10 (Inliers and Cliques): Assume we are given measurements (1) and the corresponding n-invariants; call $\mathcal{G}$ the corresponding compatibility graph. Then, assuming there are at least n inliers, the inliers form a clique in $\mathcal{G}$. □

Theorem 10 implies that we can look for inliers by computing cliques in the compatibility graph. Since we expect to have more compatible inliers than outliers, we propose to compute the maximum clique, which is shown to work extremely well in practice in Section VI. On the other hand, when the maximum clique is too expensive to compute, we propose to use the *maximum k-core* (defined below) instead, which empirically approximates the maximum clique, while being much faster (linear-time [22]) to compute.

Definition 11 (k-core, Fig. 1): A subgraph $\mathcal{C}_k$ of $\mathcal{G}$ is called a *k-core* if it is the largest subgraph such that all its vertices have vertex degree (*i.e.,* number of incident edges) at least k within $\mathcal{C}_k$. The *core number* $k(i)$ of a vertex i is defined as the largest k such that i is in $\mathcal{C}_k$. The *degeneracy* $k^\star(\mathcal{G})$ is the largest core number of all vertices in $\mathcal{G}$.

It is well-known that the maximum clique is included in the $(\omega(\mathcal{G})-1)$-core [50]. Although few conditions are known to guarantee that the maximum clique is inside the *maximum* k-core, previous work [56] has observed that this is often the case in practical problems. Indeed, Section VI shows that –in compatibility graphs arising in geometric perception– the maximum k-core provides a great approximation to the maximum clique, while being faster to compute.

While maximum clique has been used for outlier rejection in specific applications [62], [39], our framework generalizes to $n(\geq 2)$-invariants (rather than pairwise) and also provides a linear-time algorithm (based on k-core) for outlier rejection. The interested reader can find a broader discussion in [52].

V. ROBIN: Summary of Our Algorithm

This section summarizes how we combine measurement invariants and graph theory to develop our outlier pruning ap-

Algorithm 1: ROBIN

```
1  Input: set of measurements $\mathcal{Y}$ and model (1); $n$-invariant function
      $\boldsymbol{f}$ (for some $n$); boolean fastMode (use maximum $k$-core if true);
2  Output: subset $\mathcal{Y}^\star \subset \mathcal{Y}$ containing inliers;
3  % Initialize compatibility graph
4  $\mathcal{V} = \mathcal{Y}$;   % each vertex is a measurement
5  $\mathcal{E} = \emptyset$;   % start with empty edge set
6  % Perform compatibility tests
7  for all subsets $\mathcal{S} \subset \mathcal{Y}$ of size $n$ do
8      if testCompatibility($\mathcal{S}, \boldsymbol{f}$) = pass then
9          add edges between any $i, j \in \mathcal{S}$ to $\mathcal{E}$;
10     end
11 end
12 % Find compatible measurements
13 if fastMode = true then
14     $\mathcal{Y}^\star = \text{max\_kcore}(\mathcal{V}, \mathcal{E})$;
15 else
16     $\mathcal{Y}^\star = \text{max\_clique}(\mathcal{V}, \mathcal{E})$;
17 end
18 return: $\mathcal{Y}^\star$.
```

proach, named ROBIN (*Reject Outliers Based on INvariants*). ROBIN's pseudocode is given in Algorithm 1. The algorithm takes a set of measurements $\mathcal{Y}$ and outputs a subset $\mathcal{Y}^\star \subset \mathcal{Y}$ from which many outliers have been pruned. If the problem admits an n-measurement invariant, ROBIN first performs compatibility tests on all subsets of n measurements and builds the corresponding compatibility graph (lines 3-11). Then it uses maximum clique or k-core solvers to compute and return the subset of measurements containing the inliers (lines 12-18). We have implemented ROBIN in C++, using the parallel core decomposition algorithm from [33] and parallel maximum clique solver from [50]. We remark that ROBIN is not guaranteed to reject all outliers. Indeed, as mentioned in the introduction, ROBIN is designed to be a preprocessing step to enhance the robustness of existing robust estimators.

VI. Experiments

We test the performance of ROBIN on four geometric perception problems: single rotation averaging (Section VI-A), point cloud registration and point-with-normal registration (Section VI-B and Section VI-C), and 2D-3D camera pose estimation (Section VI-D). We show that (i) ROBIN boosts the robustness of GNC to more than 95% outliers and dominates the state of the art, and (ii) ROBIN runs in tens of milliseconds on challenging large-scale datasets. In addition, we present an experiment in which the maximum k-core becomes overly conservative and fails to reject outliers (Section VI-C).

We denote as ROBIN* the version of ROBIN that computes the maximum clique (fastMode=false in Algorithm 1), while we denote as ROBIN the version that computes the maximum k-core. Moreover, we denote as ROBIN*+GNC (resp. ROBIN+GNC) the estimation approach where we first remove outliers using ROBIN* (resp. ROBIN), and then use GNC [57] to compute an estimate. All experiments are run on a Linux computer with an Intel i9-9920X CPU at 3.5 GHz.

A. Single Rotation Averaging

Setup. In each Monte Carlo run, we first generate a random ground-truth rotation $\boldsymbol{R}^\circ \in \mathrm{SO}(3)$. Then we create $N = 1000$ measurements $\{\boldsymbol{R}_i\}_{i=1}^N$ with increasing outlier rates from 0% to 99%. An inlier measurement $\boldsymbol{R}_i$ is generated as $\boldsymbol{R}_i = \boldsymbol{R}^\circ \mathrm{Exp}(\theta_i \boldsymbol{u}_i)$, where $\boldsymbol{u}_i$ is a random 3D unit vector and $\theta_i \sim \mathcal{N}(0, \sigma^2)$ is a random rotation angle ($\sigma = 5°$), while an outlier measurement is a random 3D rotation. We compare five algorithms in 100 Monte Carlo runs: (i) ROBIN*+GNC; (ii) ROBIN+GNC; (iii) GNC (without ROBIN); (iv-v) Lee-Chordal and Lee-Geodesic [36], two recent state-of-the-art methods (we use the authors' implementations).

Results. Fig. 2 boxplots the rotation errors of the five algorithms under selected outlier rates (full results can be found in [52]). Both ROBIN*+GNC and ROBIN+GNC are robust against 98% outliers, while GNC, Lee-Chordal, and Lee-Geodesic start failing at 90% outliers. As expected, ROBIN*+GNC slightly outperforms ROBIN+GNC at 99% outliers, as shown by the fewer failures ("+") of ROBIN*+GNC.

B. Registration With Correspondences

Simulated Benchmarks. We use the Bunny model from the Stanford 3D scanning repository. We downsample the model to 1000 points and resize within a $[0, 1]^3$ cube to obtain the source point cloud. We then apply a random transformation $(\boldsymbol{R}, \boldsymbol{t})$ with $\boldsymbol{R} \in \mathrm{SO}(3)$ and $\|t\| \leq 1$ according to (5). We add bounded noise $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$ and ensure $\|\boldsymbol{\epsilon}_i\| \leq \beta$. We set $\sigma = 0.01$ and $\beta = 5.54\sigma$ such that $\mathbb{P}\left(\|\boldsymbol{\epsilon}_i\|^2 > \beta^2\right) \leq 10^{-6}$. To generate outliers, we replace some $\boldsymbol{b}_i$'s with vectors uniformly sampled in a sphere with radius 5. We benchmark 9 methods: (i) FGR: Fast Global Registration [65], which is a GNC approach tailored to registration; (ii-iii) ROBIN*+FGR, ROBIN+FGR: FGR with our outlier pruning; (iv-v) ROBIN*+HORN, ROBIN+HORN: Horn's method [30] with our outlier pruning; (vi) TEASER++ [62], which is equivalent to using ROBIN* with a decoupled GNC solver; (vii) TEASER(ROBIN)++: same as TEASER++, except ROBIN (with k-core) is used to reject outliers; (viii-ix) RANSAC1min and RANSAC10K: two RANSAC variants (with 99% confidence) that terminate after a maximum of one minute or 10,000 iterations, respectively.

Results. Fig. 3 shows translation errors for increasing outlier rates (rotation errors are qualitatively similar and reported in [52]). None of the ROBIN-based methods fail at outlier rates below 95%, so we omit those results for brevity. Methods using ROBIN* and ROBIN dominate FGR and RANSAC. ROBIN+HORN, ROBIN+FGR, TEASER(ROBIN)++ fail at 99%. Notably, ROBIN* is so robust that even Horn's method (which is not robust to outliers) succeeds at 98% outliers. All six ROBIN-based methods have runtime of tens of milliseconds (average: 4ms), with TEASER(ROBIN)++ being the fastest, due to its optimized C++ implementation of GNC.

3DMatch Dataset. We evaluate ROBIN on the 8 test scenes of the 3DMatch dataset [64], which contains real-world indoor RGB-D scans. Using FPFH descriptors, we generate feature correspondences and feed them to TEASER++ and TEASER(ROBIN)++. We also compute the normals using Open3D [66] and feed the point-with-normal correspondences to ROBIN*+GNC and ROBIN+GNC (the description of the corresponding GNC solver can be found in [52]).

Results. Table I shows the percentage of successful registrations for each scene. A registration is considered successful if the estimated transformation has (i) rotation error $\leq 15°$, and (ii) translation error $\leq$ 30cm (similar to [62]). In some scenarios, using normals might boost performance,

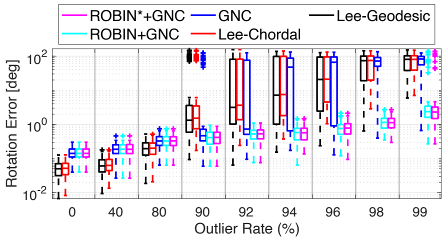

Fig. 2. Single rotation averaging.

Scenes	Kitchen (%)	Home 1 (%)	Home 2 (%)	Hotel 1 (%)	Hotel 2 (%)	Hotel 3 (%)	Study Room (%)	MIT Lab (%)	Average Runtime [ms]
TEASER++	**93.9**	**93.6**	71.6	96.0	87.5	**92.6**	**79.8**	**76.6**	33.8
TEASER(ROBIN)++	83.0	84.0	68.3	90.3	81.7	87.0	71.2	64.9	14.9
ROBIN*+GNC (points with normals)	91.9	89.1	**74.5**	**96.5**	**89.4**	**92.6**	77.7	71.4	41.2
ROBIN+GNC (points with normals)	91.9	89.1	**74.5**	**96.5**	**89.4**	**92.6**	77.7	71.4	39.5

TABLE I

PERCENTAGE OF SUCCESSFUL REGISTRATIONS AND RUNTIME ON THE 3DMATCH DATASET [64].

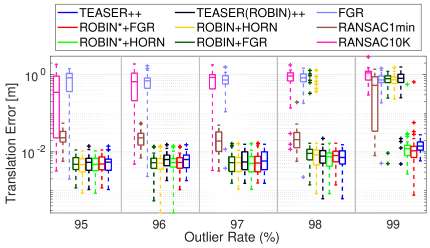

Fig. 3. Registration with correspondences.

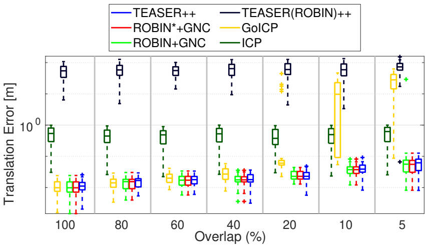

Fig. 4. Registration without correspondences

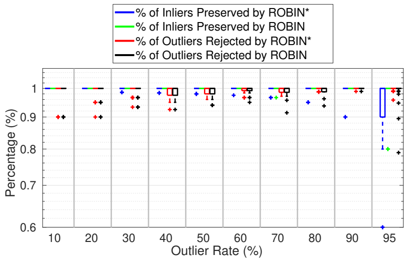

Fig. 5. 2D-3D pose estimation

depending on the quality of the normals. More interestingly, using normals makes ROBIN*+GNC and ROBIN+GNC attain similar performance: intuitively, better invariants lead to sparser compatibility graphs, making the maximum k-core a better approximation of the maximum clique. In terms of runtime, ROBIN can be up to 2 times faster than ROBIN*.

C. Registration Without Correspondences

Setup. We test the extreme case where we assume all-to-all correspondences between the two point clouds [62]. We use the TeddyBear model from HomebrewedDB [34]. Contrary to Bunny, TeddyBear comes with precomputed normals so we also test the point-with-normal variant of ROBIN. We first downsample the TeddyBear to 100 points to obtain the source point cloud. Then, we apply a random rotation and translation to obtain the transformed point cloud. To simulate a partial overlap, we randomly discard a percentage of the transformed point cloud. For example, to simulate a 80% overlap, we randomly discard 20% of the transformed points. We compare the performance of 6 algorithms in 40 Monte Carlo runs: (i-ii) ROBIN*+GNC and ROBIN+GNC, which use points with normals; (iii-iv) TEASER++, TEASER(ROBIN)++, which disregard the normals, (v) GoICP [63], and (vi) ICP. We feed ROBIN*+GNC, ROBIN+GNC, TEASER++, and TEASER(ROBIN)++ all possible correspondences: for each point in a source point cloud, we add all transformed points as potential correspondences. For ICP, we use the identity as initial guess.

Results. Fig. 4 shows translation errors for different overlap ratios (rotation errors are similar and reported in [52]). ICP fails to converge in most cases, due to the inaccurate initial guess. GoICP is robust until 20% overlap. ROBIN*+GNC, ROBIN+GNC, TEASER++ ensure best performance for all overlap ratios. ROBIN*+GNC and ROBIN+GNC have slightly lower errors (since they use extra information from the normals). As expected the variants using the maximum clique are more robust and still recover highly-accurate estimates at 5% overlap. Unlike ROBIN+GNC, TEASER(ROBIN)++ fails at all overlap ratios because the maximum k-core is only able to reject a small percentage of outliers, inducing failures in the robust estimator. In terms of runtime, the k-core variants are faster: the average runtime for ROBIN ranges from 0.1ms to 46ms, and is 2 to 3.4 times faster than ROBIN*. In our current implementation the construction of the compatibility graph is single-threaded and dominates the total runtime; this provides further speed-up opportunities.

D. 2D-3D Pose Estimation

Setup. We evaluate ROBIN and ROBIN* in a simulated 2D-3D pose estimation problem. We use the cross ratio as a 4-measurement invariant. We assume a pinhole camera projection model, with a image size of 640 × 480. We general 100 collinear 3D points and ensure they are in the field of view of the camera (more details in [52]). We then project the 3D points back to the image plane to have 2D-3D correspondences. Bounded random noise $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$, with $\sigma = 0.1$, is added to the 2D projections, and we ensure $\|\boldsymbol{\epsilon}_i\| \leq \beta = 0.25$. Outliers are introduced by replacing some of the 2D points with randomly sampled points in the image. We conduct 40 Monte Carlo runs and record: (i) the percentage of rejected outliers, and (ii) the percentage of inliers in the set returned by ROBIN and ROBIN*.

Results. Fig. 5 shows that both ROBIN and ROBIN* reject nearly 100% outliers at all outlier levels. The percentages of inliers preserved by ROBIN and ROBIN* are close to 100% until 90%, after which it drops to around 90%; this is the case where outlier correspondences are starting to form large cliques. We remark that ROBIN and ROBIN* are able to reject outliers even in a setup (such as the one we used in our simulations) where the pose cannot be uniquely computed due to the collinearity of the points.

VII. CONCLUSION

We proposed ROBIN, a fast approach to prune outliers in generic estimation problems. ROBIN uses the notion of n-invariant to quickly check if subsets of measurements are mutually compatible. Then, it prunes outliers by computing the maximum clique or maximum k-core of the graph induced by the invariants. We evaluate ROBIN in several problems and show that ROBIN boosts robustness of existing estimators –making them robust to more than 95% outliers– and runs in milliseconds on large-scale problems.

References

[1] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard. Robust map optimization using dynamic covariance scaling. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2013.

[2] P. Antonante, V. Tzoumas, H. Yang, and L. Carlone. Outlier-robust estimation: Hardness, minimally-tuned algorithms, and applications. *arXiv preprint arXiv: 2007.15109*, 2020. (pdf).

[3] K. Arun, T. Huang, and S. Blostein. Least-squares fitting of two 3-D point sets. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(5):698–700, sept. 1987.

[4] E. Ask, O. Enqvist, and F. Kahl. Optimal geometric fitting under the truncated l2-norm. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1722–1729, 2013.

[5] T. Bailey, E. M. Nebot, J. K. Rosenblatt, and H. F. Durrant-Whyte. Data association for mobile robot navigation: a graph theoretic approach. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, volume 3, pages 2512–2517, 2000.

[6] D. Barath and J. Matas. Graph-cut ransac. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6733–6741, 2018.

[7] D. Barath and J. Matas. Graph-cut ransac. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6733–6741, 2018.

[8] T. Barfoot. *State Estimation for Robotics*. Cambridge University Press, 2017.

[9] J. T. Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4331–4339, 2019.

[10] M. J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *Intl. J. of Computer Vision*, 19(1):57–91, 1996.

[11] A. Blake and A. Zisserman. *Visual reconstruction*. MIT Press, 1987.

[12] M. Bosse, G. Agamennoni, and I. Gilitschenski. Robust estimation and applications in robotics. *Foundations and Trends in Robotics*, 4(4):225–269, 2016.

[13] M. Bray, P. Kohli, and P. Torr. PoseCut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *European Conf. on Computer Vision (ECCV)*, pages 642–655, 2006.

[14] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans. Robotics*, 32(6):1309–1332, 2016. arxiv preprint: 1606.05830, (pdf).

[15] Z. Cai, T.-J. Chin, and V. Koltun. Consensus maximization tree search revisited. In *Intl. Conf. on Computer Vision (ICCV)*, pages 1637–1645, 2019.

[16] L. Carlone, A. Censi, and F. Dellaert. Selecting good measurements via ℓ_1 relaxation: a convex approach for robust estimation over graphs. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014. (pdf).

[17] T.-J. Chin, Z. Cai, and F. Neumann. Robust fitting in computer vision: Easy or hard? In *European Conf. on Computer Vision (ECCV)*, 2018.

[18] T.-J. Chin, P. Purkait, A. Eriksson, and D. Suter. Efficient globally optimal consensus maximisation with tree search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2413–2421, 2015.

[19] T. J. Chin and D. Suter. The maximum consensus problem: recent algorithmic advances. *Synthesis Lectures on Computer Vision*, 7(2):1–194, 2017.

[20] J. Choi and G. Medioni. Starsac: Stable random sample consensus for parameter estimation. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 675–682. IEEE, 2009.

[21] O. Chum and J. Matas. Matching with PROSAC - progressive sample consensus. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[22] N. S. Dasari, R. Desh, and M. Zubair. Park: An efficient algorithm for k-core decomposition on multicore processors. In *2014 IEEE International Conference on Big Data (Big Data)*, pages 9–16.

[23] F. Dellaert and M. Kaess. Factor graphs for robot perception. *Foundations and Trends in Robotics*, 6(1-2):1–139, 2017.

[24] O. Enqvist, E. Ask, F. Kahl, and K. Åström. Robust fitting for multiple view geometry. In *European Conf. on Computer Vision (ECCV)*, pages 738–751. Springer, 2012.

[25] O. Enqvist, K. Josephson, and F. Kahl. Optimal correspondences from pairwise constraints. In *Intl. Conf. on Computer Vision (ICCV)*, pages 1295–1302, 2009.

[26] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. ACM*, 24:381–395, 1981.

[27] P. Gros and L. Quan. Projective invariants for vision. 1992.

[28] R. Hartley, J. Trumpf, Y. Dai, and H. Li. Rotation averaging. *IJCV*, 103(3):267–305, 2013.

[29] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[30] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Amer.*, 4(4):629–642, Apr 1987.

[31] P. Huber. *Robust Statistics*. John Wiley & Sons, New York, NY, 1981.

[32] G. Izatt, H. Dai, and R. Tedrake. Globally optimal object pose estimation in point clouds with mixed-integer programming. In *Proc. of the Intl. Symp. of Robotics Research (ISRR)*, 2017.

[33] H. Kabir and K. Madduri. Parallel k-core decomposition on multicore platforms. In *2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 1482–1491. IEEE, 2017.

[34] R. Kaskman, S. Zakharov, I. Shugurov, and S. Ilic. Homebreweddb: Rgb-d dataset for 6d pose estimation of 3d objects. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[35] P. Lajoie, S. Hu, G. Beltrame, and L. Carlone. Modeling perceptual aliasing in SLAM via discrete-continuous graphical models. *IEEE Robotics and Automation Letters (RA-L)*, 2019. extended ArXiv version: (pdf), Supplemental Material: (pdf).

[36] S. H. Lee and J. Civera. Robust single rotation averaging. *arXiv preprint arXiv:2004.00732*, 2020.

[37] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1482–1489. IEEE, 2005.

[38] H. Li. Consensus set maximization with guaranteed global optimality for robust geometry estimation. In *Intl. Conf. on Computer Vision (ICCV)*, pages 1074–1080, 2009.

[39] J. G. Mangelson, D. Dominic, R. M. Eustice, and R. Vasudevan. Pairwise consistent measurement set maximization for robust multi-robot map merging. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 2916–2923, 2018.

[40] J. Mundy and A. Zisserman. *Geometric invariance in computer vision*. MIT Press, Cambridge, MA, USA, 1992.

[41] J. Neira and J. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Trans. Robot. Automat.*, 17(6):890–897, December 2001.

[42] Á. Parra Bustos and T. J. Chin. Guaranteed outlier removal for rotation search. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2165–2173, 2015.

[43] Á. Parra Bustos and T. J. Chin. Guaranteed outlier removal for point cloud registration with correspondences. *IEEE Trans. Pattern Anal. Machine Intell.*, 40(12):2868–2882, 2018.

[44] A. Parra Bustos, T.-J. Chin, F. Neumann, T. Friedrich, and M. Katzmann. A practical maximum clique algorithm for matching with pairwise constraints. *arXiv preprint arXiv:1902.01534*, 2019.

[45] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis. 6-dof object pose from semantic keypoints. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[46] S. Perera and N. Barnes. Maximal cliques based rigid body motion segmentation with a rgb-d camera. In *Asian Conference on Computer Vision*, pages 120–133. Springer, 2012.

[47] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. Usac: a universal framework for random sample consensus. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):2022–2038, 2012.

[48] R. Raguram, J.-M. Frahm, and M. Pollefeys. A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus. In *European Conference on Computer Vision*, pages 500–513. Springer, 2008.

[49] A. Rosinol, A. Gupta, M. Abate, J. Shi, and L. Carlone. 3D dynamic scene graphs: Actionable spatial perception with places, objects, and humans. In *Robotics: Science and Systems (RSS)*, 2020. (pdf), (video).

[50] R. A. Rossi, D. F. Gleich, and A. H. Gebremedhin. Parallel maximum clique algorithms with applications to network analysis. *SIAM Journal on Scientific Computing*, 37(5):C589–C616, 2015.

[51] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016.

[52] J. Shi, H. Yang, and L. Carlone. ROBIN: a graph-theoretic approach to reject outliers in robust estimation using invariants. *arXiv preprint arXiv: 2011.03659*, 2020. (pdf).

[53] K. Tanaka and E. Kondo. Incremental ransac for online relocation in large dynamic environments. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 68–75. IEEE, 2006.

[54] K. M. Tavish and T. D. Barfoot. At all costs: A comparison of robust

cost functions for camera correspondence outliers. In *Computer and Robot Vision (CRV), 2015 12th Conference on*, pages 62–69. IEEE, 2015.

[55] P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Comput. Vis. Image Underst.*, 78(1):138–156, 2000.

[56] J. L. Walteros and A. Buchanan. Why is maximum clique often easy in practice? *Operations Research*, 2020.

[57] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robotics and Automation Letters (RA-L)*, 5(2):1127–1134, 2020. arXiv preprint arXiv:1909.08605 (with supplemental material), (pdf) .

[58] H. Yang and L. Carlone. A polynomial-time solution for robust registration with extreme outlier rates. In *Robotics: Science and Systems (RSS)*, 2019. (pdf), (video), (media), (media), (media).

[59] H. Yang and L. Carlone. A quaternion-based certifiably optimal solution to the Wahba problem with outliers. In *Intl. Conf. on Computer Vision (ICCV)*, 2019. (Oral Presentation, accept rate: 4%), Arxiv version: 1905.12536, (pdf).

[60] H. Yang and L. Carlone. In perfect shape: Certifiably optimal 3D shape reconstruction from 2D landmarks. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. Arxiv version: 1911.11924, (pdf).

[61] H. Yang and L. Carlone. One ring to rule them all: Certifiably robust geometric perception with outliers. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. (pdf).

[62] H. Yang, J. Shi, and L. Carlone. TEASER: Fast and Certifiable Point Cloud Registration. *IEEE Trans. Robotics*, 2020. extended arXiv version 2001.07715 (pdf).

[63] J. Yang, H. Li, D. Campbell, and Y. Jia. Go-ICP: A globally optimal solution to 3D ICP point-set registration. *IEEE Trans. Pattern Anal. Machine Intell.*, 38(11):2241–2254, Nov. 2016.

[64] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser. 3dmatch: Learning the matching of local 3d geometry in range scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 4, 2017.

[65] Q. Zhou, J. Park, and V. Koltun. Fast global registration. In *European Conf. on Computer Vision (ECCV)*, pages 766–782. Springer, 2016.

[66] Q.-Y. Zhou, J. Park, and V. Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.