MAC++: Going Further with Maximal Cliques for 3D Registration

Xiyu Zhang Yanning Zhang Jiaqi Yang*
School of Computer Science, Northwestern Polytechnical University, China
{2426988253}@mail.nwpu.edu.cn; {ynzhang, jqyang}@nwpu.edu.cn

Abstract

*Maximal cliques (MAC) represent a novel state-of-the-art approach for 3D registration from correspondences, however, it still suffers from extremely severe outliers. In this paper, we introduce a robust learning-free estimator called MAC++, exploring maximal cliques for 3D registration from the following two perspectives: 1) A novel hypothesis generation method utilizing putative seeds through voting to guide the construction of maximal clique pools, effectively preserving more potential correct hypotheses. 2) A progressive hypothesis evaluation method that continuously reduces the solution space in a "global-clusters-cluster-individual" manner rather than traditional one-shot techniques, greatly alleviating the issue of missing good hypotheses. Experiments conducted on U3M, 3DMatch/3DLoMatch, and KITTI-LC datasets show the new state-of-the-art performance of MAC++. MAC++ demonstrates the capability to handle extremely low inlier ratio data where MAC fails (*e.g.*, showing* $\mathbf{27.1\%/30.6\%}$ *registration recall improvements on 3DMatch/3DLoMatch with $<$ 1% inliers).*

1. Introduction

3D point cloud registration (PCR) is a foundational problem in 3D computer vision, critical for applications such as autonomous driving [41, 56], simultaneous localization and mapping (SLAM) [6], and 3D reconstruction [13, 37]. PCR aligns point clouds to estimate a six-degree-of-freedom (6-DoF) pose transformation, generally from a set of correspondences. However, registration from extremely noisy correspondences remains a challenging issue.

Hypothesis generation and evaluation (HGE) is a popular paradigm for PCR with correspondences, which are either traditional or deep-learned. For traditional methods, the widely applied RANSAC and its variants [4, 12, 16, 18, 39, 46] follow an iterative sampling and verification pipeline. RANSAC-based approaches are characterized by

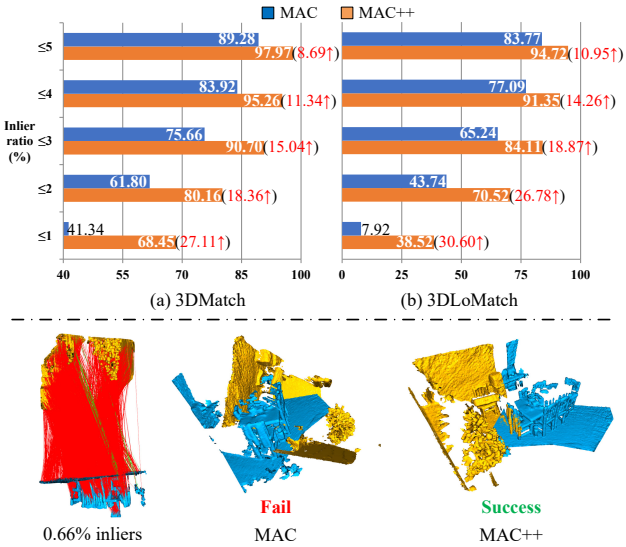

Figure 1. Comparison of MAC++ and MAC on data with low inlier ratios. Please refer to Tables 1 and 2 for more results.

their simplicity and efficiency but degrade with severe outliers. Some approaches [5, 10, 27, 28, 44] employ branch-and-bound (BnB) to achieve the global optimal solution, while they are time-consuming when dealing with considerably large data and a high proportion of outliers. For deep-learned methods, many approaches [3, 15, 19, 26, 33] focus on improving the quality of correspondences for hypothesis generation. However, deep-learned methods require extensive training data and often lack generalization ability to new scenarios.

In recent years, several approaches [40, 43, 52] perform inlier selection using the maximum clique search technique. These methods assume inliers are mutually compatible, forming a maximum clique within the graph constructed from correspondences. However, this assumption is questionable as inliers also suffer from matching errors, and existing compatibility metrics have measuring errors. Recently, Zhang *et al.* [57] relaxed the maximum clique constraint to the maximal cliques (MAC) and have achieved promising results. However, we find that MAC still has two

*Corresponding author. Code is available at https://github.com/zhangxy0517/MAC-PLUS-PLUS.

limitations: 1) Correct cliques are missed. The node-guided (NG) clique selection method, which relies on clique weight to identify suitable cliques, can overlook many correct ones due to its sensitivity to measurement errors in compatibility metrics. 2) Correct hypotheses are missed. MAC uses traditional one-shot hypothesis evaluation metrics that are less selective when faced with massive false hypotheses.

In this paper, we introduce a robust learning-free estimator called MAC++ for 3D registration. It can handle challenging data that MAC encounters difficulties, as illustrated in Fig. 1. First, more correct maximal cliques are retained. Given the high probability of correct cliques being present around reliable nodes, we utilize a clique voting mechanism to select potential seed nodes, ensuring that clique sampling is guided by these seeds. Furthermore, we establish a clique pool for each seed, thus relaxing the single maximal clique constraint to the maximal clique pool for each node. It can enhance the probability of sampling correct cliques. Second, hypotheses are more reliably evaluated. Previous one-shot evaluation approaches struggle to identify correct hypotheses in ultra-noisy hypothesis space caused by heavy outliers. By contrast, we propose a hierarchical approach and introduce a "global-clusters-cluster-individual" strategy to prune incorrect matches and hypotheses simultaneously.

To summarize, our main contributions are as follows:

- We introduce a novel hypothesis generation method, which utilizes a group of voted putative seeds to construct maximal clique pools and guide the clique sampling process. This approach further relaxes the constraints on nodes and effectively retains more correct hypotheses.
- We propose a progressive hypothesis evaluation module that reduces the solution space in a "global-clusters-cluster-individual" manner. Compared to one-shot evaluation approaches, it greatly alleviates the issue of missing correct solutions in the presence of heavy outliers.
- We introduce MAC++, a learning-free robust estimator for 3D registration. It excels in handling challenging scenarios such as extremely low inlier ratios, low overlap rates, and long distances, where MAC fails. It reaches new state-of-the-art performance on challenging datasets and also acts as a booster to deep-learned methods.

2. Related Work

2.1. Traditional PCR Methods

HGE-based Methods. The RANSAC algorithm [18] is commonly employed in the HGE-based registration process, which iteratively performs random sampling and evaluation. Since the discovery of its simplicity and effectiveness, numerous researchers have made efforts toward enhancing the correspondence sampling [4, 16, 39, 46, 48] and evaluation metrics [47] for further improvements. Heuristic sampling techniques have been introduced to improve both efficiency and quality. Yang *et al*. [46] proposed SAC-COT, which samples correspondence triplets under the guidance of graph properties. Accurate registration heavily relies on the presence of efficient and resilient evaluation metrics. As a solution to the current challenges associated with evaluation metrics that are both time-consuming and sensitive to noise, Yang *et al*. [47] investigated how inliers and outliers affect the computation and present a range of metrics aimed at substantially improving the registration performance. However, these RANSAC-based methods suffer from low time efficiency and limited accuracy in cases with high outlier rates. To address the ambiguity of commonly used spatial consistency [36], Chen *et al*. [11] presented a second-order compatibility measurement (SC^2) to distinguish inliers and obtain reliable sampling. Zhang *et al*. [57] introduced MAC, which employs maximal cliques in the SC^2 graph to mine local consensus and generate transformation hypotheses.

Others. In recent times, a number of approaches utilizing BnB have emerged since it can provide the globally best solution. These methods [5, 10, 27, 28, 44] recursively divide the parameter space into smaller branches and eliminate those that are incapable of containing the optimal solution through bound checks. Bustos and Chin [5] presented guaranteed outlier removal (GORE) to reject true outliers by reducing the original 6-DoF problem to a 3-DoF problem. Chen *et al*. [10] designed a two-stage search strategy to decompose the 6-DoF into (2+1)-DoF and (1+2)-DoF sub-problems. However, BnB techniques will be slow for large-scale inputs.

2.2. Deep-learned PCR Methods

Some methods [3, 15, 19, 26, 33] focus on efficient inlier prediction. Deep global registration (DGR) [15] and 3DRegNet [33] classify a given correspondence by training end-to-end neural networks and using operators such as sparse convolution and point-by-point MLP. PointDSC [3] explicitly explores spatial consistency for removing outlier correspondences and 3D point cloud registration. Lee [26] presented DHVR to identify the consensus among the correspondences from the 6D Hough space. Yao *et al*. [50] proposed Hunter to learn the high-order consistency among inliers in a global-to-local exploration scheme. More recently, several methods [8, 35, 53] estimate the transformation in an end-to-end way. CoFiNet [53] extracts correspondences from coarse to fine without keypoint detection. GeoTransformer [35] learns geometric features for robust superpoint matching and is robust in low-overlap cases and invariant to rigid transformation. RegFormer [30] performs end-to-end registration on large-scale point clouds without any further post-processing.

Despite their impressive capabilities, deep learning methods often require extensive training data and struggle

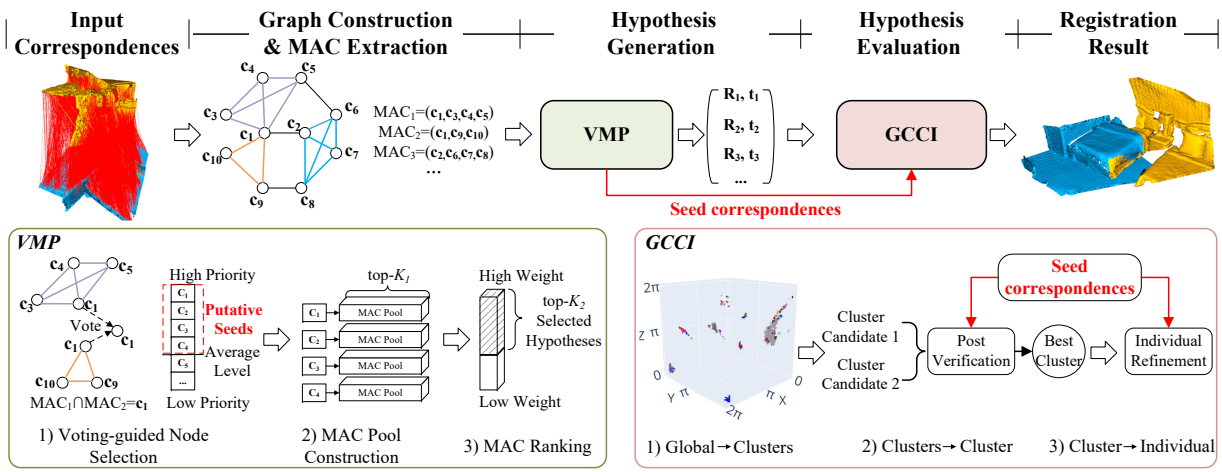

Figure 2. **Pipeline of MAC++**. **1.** Construct a graph for the input correspondences and search for maximal cliques within the graph space. **2.** Hypothesis generation with voted maximal clique pool (VMP). **3.** Progressive hypothesis evaluation in a "global-clusters-cluster-individual" (GCCI) manner. **4.** The best hypothesis is selected to perform 3D registration.

to generalize well to new scenarios.

3. Revisiting MAC

The key insight of MAC [57] is to loosen the previous maximum clique constraint, and mine more local consensus information in a graph to generate accurate hypotheses. 1) It represents the initial correspondence set as a compatibility graph with the SC^2 measurement. 2) It then detects maximal cliques within the graph space and utilizes a node-guided clique filtering method (NG) to associate each graph node with the most appropriate maximal clique. Because maximal cliques provide a more flexible constraint compared to the maximum cliques and enable the extraction of additional local information from the graph, this method can generate numerous accurate hypotheses from the graph. 3) Finally, the generated hypotheses are directly evaluated through RANSAC metrics to find the optimal solution.

The shortcomings of the MAC pipeline are twofold. 1) *Hypothesis generation*: The NG selection strategy in MAC considers the clique with the largest weight around each node as the best local consensus set. However, the clique that can generate the correct hypothesis is not necessarily the one with the highest weight among neighboring cliques. NG may lose many correct cliques, resulting in the inability to provide enough correct hypotheses for subsequent evaluation (as verified in Tables 10 and 11). 2) *Hypothesis evaluation*: When faced with a large number of outliers, two challenges emerge: firstly, the upstream generates significantly fewer correct hypotheses compared to incorrect ones. In such scenarios, the conventional one-shot evaluation method struggles to identify the correct solution (as verified in Table 12). Secondly, the existing RANSAC metrics have difficulty identifying the maximum inlier set and determining the correct hypothesis, as they rely on calculating the inlier count within the correspondence set. The existence of these two issues makes it difficult for MAC to handle data with extremely low inlier ratios and limits its performance in practical applications.

In the next section, we will provide a detailed explanation of MAC++ to address the shortcomings of MAC. Specifically, we propose a novel hypothesis generation method with voted maximal clique pool to produce more correct hypotheses and a progressive hypothesis evaluation method to find the correct solution from severe outliers.

4. MAC++

4.1. Problem Formulation

Given source cloud $\mathbf{P}^s$ and target cloud $\mathbf{P}^t$, we first extract local features for them using geometric or learned descriptors. Let $\{\mathbf{p}^s\}$ and $\{\mathbf{p}^t\}$ denote the points in the $\mathbf{P}^s$ and $\mathbf{P}^t$, respectively. An initial correspondence set $\mathbf{C}_{initial} = \{\mathbf{c}\}$ can be formed by matching local features, where $\mathbf{c} = (\mathbf{p}^s, \mathbf{p}^t)$. The goal is to estimate an accurate transformation from $\mathbf{C}_{initial}$. Since 3D correspondences are unordered in the Euclidean space, we suggest representing them as a graph. Specifically, we model the input correspondences as a compatibility graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where nodes $\mathbf{V}$ represent correspondences and edges $\mathbf{E}$ connect nodes that exhibit geometric compatibility. Given that the second-order graph (SOG) possesses characteristics that are more consistent and robust compared to the first-order graph (FOG) [11, 57], we perform a clique search in SOG as [57] and obtain the maximal clique set $\mathbf{MAC}_{initial}$. The pipeline of our method is shown in Fig. 2. Compared with MAC, our approach has more robust and efficient hypothe-

sis generation and evaluation modules.

4.2. Hypothesis Generation with Voted Maximal Clique Pool

As $\mathbf{MAC}_{initial}$ is typically large, it needs filtering to reduce its magnitude while preserving correct cliques for hypothesis generation. To address the issue of missing correct cliques in MAC, we propose a novel hypothesis generation method with voted maximal clique pool (VMP), which includes three steps: voting-guided node selection, maximal clique pool construction, and clique ranking.

Voting-Guided Node Selection. Given a clique $\mathbf{CL}_i = (\mathbf{V}_i, \mathbf{E}_i)$, the weight $w(\mathbf{CL}_i)$ is calculated as:

$$w(\mathbf{CL}_i) = \sum_{e_j \in \mathbf{E}_i} w(e_j), \tag{1}$$

where $w(e_j)$ represents the edge weight of e_j in the adjacent matrix of SOG. For each node $v_i \in \mathbf{V}$, the MACs that contain it are regarded as its neighbor cliques and the neighbor clique set is denoted as $NC(v_i) = \{\mathbf{CL}_j | v_i \in \mathbf{CL}_j\}$. Then, v_i is voted by $NC(v_i)$ as:

$$w(v_i) = \sum_{\mathbf{CL}_j \in NC(v_i)} w(\mathbf{CL}_j). \tag{2}$$

The voting score $w(v_i)$ is related to both the size of $NC(v_i)$ and the consistency of each $\mathbf{CL}_j$ in the set. A higher score indicates a higher likelihood of correct cliques being present in $NC(v_i)$. The nodes are sorted in descending order by their scores, and the top-ranked ones are selected as reliable seeds (*i.e.*, putative correspondences). Specifically, we select nodes with scores higher than the average as the final seeds. Defining some reliable seed nodes enhances the correctness of the following generated hypotheses and supports the later evaluation.

Maximal Clique Pool Construction. We loosen the NG-based maximal clique selection constraint in [57] by extending one to a pool. Following the seed ranking order, each seed selects the top-K_1 weighted cliques in $NC(v_i)$ to place in the pool $P(v_i)$. If the pool generated by a lower-priority seed contains the same cliques as the pool of a higher-priority seed, they will be removed from the lower-priority pool. Compared to the method of selecting a unique clique for each node, our sampling approach can effectively mine more correct cliques.

Clique Ranking. After the maximal clique pools are constructed, we consolidate the cliques from all the pools and arrange them in descending order by their weights, and then finally retain the top-K_2 cliques. Every maximal clique represents a coherent collection of correspondences. Each maximal clique can generate a hypothesis via SVD.

4.3. GCCI-based Progressive Hypothesis Evaluation

After obtaining the hypotheses, our objective is to select the most suitable one. When the data is heavily influenced by outliers, determining the most appropriate transformation straightforwardly becomes a challenging task. Consequently, we assess hypotheses progressively rather than in the previous one-shot manner. Our approach consists of three parts: first, clustering all generated hypotheses (from global to clusters); then, evaluating and selecting the best cluster from clusters (from clusters to a single cluster); and finally, refining individuals based on the selected cluster (from cluster to individual). This process is denoted as GCCI, exhibiting a "global-clusters-cluster-individual" progressive fashion.

Transformation Clustering. Based on the assumption that correct transformations are similar to each other while incorrect ones are distributed discretely under heavy outliers, we partition the solution space into several clusters (as shown in Fig. 2), aiming to hierarchically search for the correct solution. For each hypothesis $\mathbf{h}_i = (\mathbf{R}_i, \mathbf{t}_i)$, where the rotation matrix $\mathbf{R}_i \in SO(3)$ and translation vector $\mathbf{t}_i \in \mathbb{R}^3$, we convert $\mathbf{R}_i$ to an Euler angle vector $\mathbf{r}_i$. Each angle vector can be treated as a 3D point, and all the angular vectors of the transformations form a point cloud representing the resolution space. Then, we employ the segmentation algorithm *ConditionalEuclideanClustering* provided by the PCL [38] library to segment the point cloud (refer to the supplementary). Transformation clustering can systematically organize the diverse transformations in the solution space, and simultaneously filter out outlier transformations.

Evaluation with Outlier-Aware MAE. The initial correspondence set $\mathbf{C}_{initial}$ is typically formed by finding the nearest neighbors of points in the source cloud $\mathbf{P}^s$ among target points of $\mathbf{P}^t$ in the feature space. Commonly applied metrics [47] (*e.g.*, inlier-count, MAE, and MSE) evaluate the generated hypotheses as follows:

$$Score(\mathbf{h}_i) = \sum_{j=1}^{N} \phi(\left\| \mathbf{R}_i \mathbf{p}_j^s + \mathbf{t}_i - \mathbf{p}_j^t \right\|), \tag{3}$$

where $\mathbf{c}_j = (\mathbf{p}_j^s, \mathbf{p}_j^t) \in \mathbf{C}_{initial}$, $N = |\mathbf{C}_{initial}|$, and ϕ is a truncated function [47], meaning matches within a certain range of errors are considered. Due to the similarity of features among neighbor points, multiple source points can be shared by the same target point. In that case, most matches formed by these points are outliers. As shown in Fig. 3, for the same set of matches, when the number of inliers predicted by the wrong transformation exceeds that of the correct transformation, simply adding the score of predicted inliers can result in a higher evaluation score for incorrect transformations. This could cause the evaluation failure of the metrics mentioned above. The situation be-

comes more severe when the inlier ratio is low, as fewer correct hypotheses are generated, and the probability that estimated inliers are outliers is higher. Based on the above observations, we introduce an Outlier-Aware Mean Absolute Error (OA-MAE) metric.

The motivation behind this is to enhance the contribution of predicted inliers to the overall evaluation score and mitigate the negative impact of outliers. We represent source points corresponding with a common target point $\mathbf{p}_j^t$ in predicted inliers by a set $\mathbf{S}_{\mathbf{p}_j^t}$. Thus, the correspondence set forms relationships between sets and target points. The evaluation score is calculated as follows:

$$Score(\mathbf{h}_k) = \sum_{j=1}^{M} \frac{1}{\left|\mathbf{S}_{\mathbf{p}_j^t}\right|} \sum_{\mathbf{p}_i^s \in \mathbf{S}_{\mathbf{p}_j^t}} \phi_{mae}(\left\|\mathbf{R}_k\mathbf{p}_i^s + \mathbf{t}_k - \mathbf{p}_j^t\right\|), \tag{4}$$

where M represents the number of target points in all matches, and ϕ_{mae} is the MAE score function [47]. The difference between Eq. 3 and Eq. 4 is, that Eq. 3 simply sums up scores of all the matches to evaluate the hypothesis, where Eq. 4 first calculates the average score of all matches in each set, and then adds them up, which means that Eq. 4 suppresses the accumulation of scores for predicted inliers in the same set. We use OA-MAE for hypothesis evaluation.

Cluster Selection. After clustering the transformations, we obtain multiple clusters, each containing similar transformations. To find the cluster containing the potential optimal solution, we propose a two-stage selection method.

In the first stage, we select cluster candidates from two perspectives. The first perspective evaluates all cluster centers and chooses the one with the best center. Within each cluster, we select the highest-scoring transformation as the central transformation and evaluate these to determine the best overall using the OA-MAE metric. To mitigate the exclusion of correct solutions during clustering, MAC++ also examines overall transformations from Sect. 4.2 from the second perspective. The cluster containing the best transformation is selected as the other candidate.

In the second stage, we perform a post-verification between the central transformation of the best cluster and the best transformation of individuals. Because seed matches described in Sect. 4.2 are generally more reliable compared to initial matches, they can reveal the overlapping regions of point clouds to some extent. As illustrated in Fig. 4, we propose a match-to-patch method to generate semi-dense point clouds using the putative seeds. Then, the two central transformations of cluster candidates are verified using the match-to-patch constraint. Specifically, the one yielding the best Chamfer distance [22, 24, 49] metric (refer to the supplementary) on the obtained point cloud patches is retained. The cluster containing this transformation is served as the final selected cluster. This cluster selection process has considered both correspondence-level (OA-MAE evaluation) and point-cloud-level (point-to-patch data) constraints to more convincingly find the best cluster.

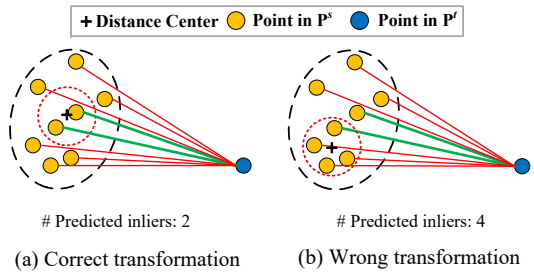

Figure 3. Multiple source points are matched to the same target point. The green and red lines denote ground truth inliers and outliers, respectively. The distance center '+' represents the position of the transformed target point in the source cloud. The points in the red circle satisfy the spatial distance constraint with the transformed target point, and the matches they formed are predicted inliers. In this example, the predicted inliers of the wrong transformation are more than those of the correct transformation.

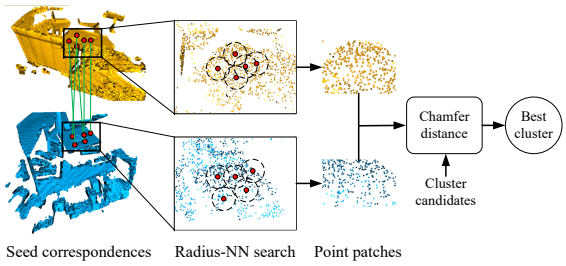

Figure 4. Illustration of post verification. We first perform a radius-NN search for every correspondence point to obtain a local keypoint patch. Then, a point patch is generated by aggregating all obtained local keypoint patches. Cluster candidates are finally verified by the semi-dense point patch.

Individual Refinement. Because the central transformation may not be optimal in the cluster, we apply individual refinement to find the best solution. We examine each transformation within the cluster one by one. Simultaneously, we intersect seed correspondences with cluster correspondences (which consist of all matches generating these transformations within the same cluster) to improve the correctness of matches used for OA-MAE evaluation. The one with the highest OA-MAE score is kept for registration.

5. Experiments

5.1. Experimental Setup

Datasets. We consider the object-scale dataset U3M [32], the scene-scale indoor datasets 3DMatch [55] & 3DLoMatch [23], and the scene-scale outdoor datasets KITTI-LC [34] for evaluation. U3M contains various nuisances presented in real-world scenarios. 3DLoMatch is the subset

Inlier ratio	$\leq 1\%$	$\leq 2\%$	$\leq 3\%$	$\leq 4\%$	$\leq 5\%$
PointDSC[3]	27.97	38.26	56.62	67.59	75.17
TEASER++[43]	23.23	46.21	66.30	76.16	83.12
SC^2-PCR[11]	36.41	57.24	72.64	80.28	86.69
MAC[57]	41.34	61.80	75.66	83.92	89.28
MAC++	**68.45**	**80.16**	**90.70**	**95.26**	**97.97**

Table 1. Controlled experiments on 3DMatch dataset.

Inlier ratio	$\leq 1\%$	$\leq 2\%$	$\leq 3\%$	$\leq 4\%$	$\leq 5\%$
PointDSC[3]	1.35	8.14	29.70	49.35	59.74
TEASER++[43]	6.51	31.72	56.49	69.96	78.05
SC^2-PCR[11]	3.31	28.58	53.96	68.56	75.91
MAC[57]	7.92	43.74	65.24	77.09	83.77
MAC++	**38.52**	**70.52**	**84.11**	**91.35**	**94.72**

Table 2. Controlled experiments on 3DLoMatch dataset.

of 3DMatch, in which the point cloud pairs exhibit an overlap rate spanning from 10% to 30%, presenting a significant level of difficulty. The KITTI-LC dataset contains more sequences compared to KITTI [20]. It provides a more comprehensive evaluation of registration methods as it takes different translation ranges into account for point cloud pairs.

Evaluation Criteria. For all experiments, we report registration recall (RR), which represents the percentage of successful pairs under specific error thresholds. For the object dataset U3M, we employ root mean squared error (RMSE). In the comparative experiments on scene datasets, the performance is evaluated using rotation error (RE) and translation error (TE). Registration is considered successful when RE $\leq 15°$, TE ≤ 30 cm on 3DMatch & 3DLoMatch datasets. For the KITTI-LC dataset, the sub-datasets "easy [0-10m]", "medium [10-20m]", and "hard [20-30m]", employ TE thresholds of 60cm, 120cm, and 180cm, respectively. The RE threshold is all set to 5°. In the performance-boosting experiments, registration is successful when RMSE < 20 cm, following the settings in [35].

Implementation Details. Our method is implemented in C++ based on the point cloud library (PCL) and igraph library. We keep the same setting in [57] to generate correspondences and construct the compatibility graph. By default, we set K_1 and K_2 mentioned in Sect. 4.2 to 10 and the size of the input correspondence set, respectively. All the experiments are conducted on a PC with an Intel 12700 CPU and 32GB RAM.

5.2. Controlled experiments on 3DMatch and 3DLoMatch

We generate point cloud pairs with varying inlier ratios based on 3DMatch/3DLoMatch (please refer to supplementary). PointDSC [3], TEASER++ [43], SC^2-PCR [11], and MAC [57] are used for comparison. The results shown in Tables 1 and 2 indicate that MAC++ is quite robust even for inputs with extremely low inlier ratios. MAC++ has **27.1%/30.6%** registration recall improvements on 3DMatch/3DLoMatch with lower than 1% inliers.

5.3. Experiments on Real Data

Results on U3M Dataset. We perform an extensive comparison in Table 3. Here, the following methods are tested, including MAC [57], SC^2-PCR [11], SAC-COT [46], OSAC [45], SAC-IA [39], RANSAC [18], FGR [58], GO-ICP [44], and PPF [17]. The RMSE threshold is varied from 1 pr to 5 pr with a step of 1 pr ('pr' denotes point cloud resolution [48]). The results indicate that MAC++ achieves the best performance at all thresholds.

	1 pr	2 pr	3 pr	4 pr	5 pr
PPF[17]	0	0.40	0.60	1.21	1.41
GO-ICP[44]	0.40	1.21	2.62	4.84	6.25
RANSAC[18]	3.23	10.28	15.93	17.94	20.77
OSAC[45]	10.48	20.16	25.00	26.61	28.23
SAC-IA[39]	2.42	10.08	16.13	20.77	22.98
FGR[58]	21.77	33.47	38.51	42.54	46.77
SAC-COT[46]	20.97	38.71	43.75	46.37	48.19
SC^2-PCR[11]	29.34	35.04	37.04	38.45	39.60
MAC[57]	33.62	48.15	54.70	57.55	59.26
MAC++	**36.54**	**50.71**	**55.81**	**58.92**	**60.06**

Table 3. Results on U3M dataset varying RMSE threshold.

	FPFH Setting			FCGF Setting		
	RR(%)	RE(°)	TE(cm)	RR(%)	RE(°)	TE(cm)
i) Traditional						
RANSAC-1M[18]	0.67	10.27	15.06	9.77	7.01	14.87
RANSAC-4M[18]	0.45	10.39	20.03	10.44	6.91	15.14
TEASER++[43]	35.15	4.38	10.96	46.76	4.12	12.89
SC^2-PCR[11]	38.57	4.03	10.31	58.73	3.80	10.44
MAC[57]	40.88	**3.66**	**9.45**	59.85	**3.50**	**9.75**
ii) Deep learned						
DGR[15]	19.88	5.07	13.53	43.80	4.17	10.82
PointDSC[3]	20.38	4.04	10.25	56.20	3.87	10.48
MAC++	**44.30**	4.12	11.02	**61.03**	3.93	11.17

Table 4. Registration results on 3DLoMatch dataset. (FPFH/FCGF: 1.76%/7.26% inliers)

Results on 3DLoMatch Datasets. Both geometric-only and deep-learned methods including RANSAC [18], TEASER++ [43], SC^2-PCR [11], MAC [57], DGR [15], and PointDSC [3] are considered for comparison.

On the 3DLoMatch dataset, MAC++ significantly improves registration performance compared to MAC and other competitors. As shown in Table 4, MAC++'s performance is 3.82% higher than MAC's when combined with FPFH, and 1.18% higher when combined with FCGF. *Note that MAC++ achieves such performance without needing additional matching* [11].

Boosting Deep-learned Methods with MAC++. Several deep-learned methods such as FCGF [14], SpinNet [1], PREDATOR [23], CoFiNet [53] and GeoTransformer [35] are integrated with MAC++ for evaluation. Each method is tested under different numbers of sampled correspondences. The results are reported in Table 6.

MAC++ working with GeoTransformer achieves com-

Subset	0-10m(3.32% inliers)						10-20m(0.86% inliers ⋆)						20-30m(0.32% inliers ⋆)					
	00	02	05	06	08	Total	00	02	05	06	08	Total	00	02	05	06	08	Total
RANSAC[18]	0.99	1.16	1.14	0	0	0.77	0	2.02	0	0.40	0	0.43	0.26	0.90	0.46	1.19	0.54	0.63
TEASER++[43]	91.42	59.88	81.82	89.66	74.79	79.32	54.55	22.73	50.75	62.55	23.46	45.79	12.60	2.69	5.94	28.57	0.54	11.11
SC^2-PCR[11]	98.68	83.72	97.16	**100**	94.87	94.64	63.64	39.90	64.82	70.12	34.57	57.08	12.60	6.28	7.76	28.97	1.08	12.22
MAC[57]	98.02	86.63	99.43	**100**	90.60	94.31	69.79	52.02	69.85	74.50	44.44	64.21	23.10	10.76	11.87	41.27	3.78	19.76
MAC++	**99.67**	**87.21**	**100**	**100**	**96.16**	**96.50**	**72.73**	**52.53**	**71.86**	**82.47**	**50**	**68.03**	**25.20**	**11.21**	**13.70**	**42.06**	**4.86**	**21.11**

Table 5. Registration results on KITTI-LC dataset. '⋆' denotes ultra-challenging cases.

# Samples	3DMatch RR(%)					3DLoMatch RR(%)				
	5000	2500	1000	500	250	5000	2500	1000	500	250
FCGF[14]	85.1	84.7	83.3	81.6	71.4	40.1	41.7	38.2	35.4	26.8
SpinNet[1]	88.6	86.6	85.5	83.5	70.2	59.8	54.9	48.3	39.8	26.8
PREDATOR[23]	89.0	89.9	90.6	88.5	86.6	59.8	61.2	62.4	60.8	58.1
CoFiNet[53]	89.3	88.9	88.4	87.4	87.0	67.5	66.2	64.2	63.1	61.0
GeoTransformer[35]	92.0	91.8	91.8	91.4	91.2	75.0	74.8	74.2	74.1	73.5
FCGF w. MAC	91.3	92.2	91.6	90.4	85.6	57.2	56.0	52.6	42.4	32.1
FCGF w. MAC++	**93.3**	**93.5**	**93.2**	**90.9**	**89.8**	**59.5**	**58.9**	**56.2**	**51.9**	**43.2**
	8.2↑	8.8↑	9.9↑	9.3↑	18.4↑	19.4↑	17.2↑	18.0↑	16.5↑	16.4↑
SpinNet w. MAC	**95.3**	95.1	93.3	91.4	81.2	**72.8**	69.9	59.2	54.8	32.1
SpinNet w. MAC++	**95.3**	**95.7**	**95.3**	**93.5**	**84.5**	**72.8**	**71.7**	**67.3**	**59.1**	**40.0**
	6.7↑	9.1↑	9.8↑	10.0↑	14.3↑	13.0↑	16.8↑	19.0↑	19.3↑	13.2↑
PREDATOR w. MAC	94.6	94.4	94.0	93.5	92.3	70.9	70.4	69.8	67.2	64.1
PREDATOR w. MAC++	**94.7**	**94.5**	**94.5**	**93.8**	**93.2**	**73.7**	**72.5**	**70.9**	**69.5**	**67.5**
	5.7↑	4.6↑	3.9↑	5.3↑	6.6↑	13.9↑	11.3↑	8.5↑	8.7↑	9.4↑
CoFiNet w. MAC	94.1	94.4	94.5	**93.8**	92.7	71.6	71.5	70.6	69.2	68.1
CoFiNet w. MAC++	**95.2**	**95.0**	**95.3**	93.7	**93.6**	**73.7**	**73.2**	**72.1**	**70.1**	**69.6**
	5.9↑	6.1↑	6.9↑	6.3↑	6.6↑	6.2↑	7.0↑	7.9↑	7.0↑	8.6↑
GeoTransformer w. MAC	**95.7**	**95.7**	**95.2**	**95.3**	94.6	78.9	78.7	**78.2**	**77.7**	76.6
GeoTransformer w. MAC++	**95.7**	**95.7**	95.1	**95.3**	**94.7**	**79.1**	**79.0**	78.1	**77.7**	**76.7**
	3.7↑	3.9↑	3.3↑	3.9↑	3.5↑	4.1↑	4.2↑	3.9↑	3.6↑	3.2↑

Table 6. Boosting results. Boost ratios of MAC++ are provided.

petitive registration recalls of **95.7% / 79.1%** on 3DMatch / 3DLoMatch. The GeoTransformer baseline produces high-quality matches, making it challenging to distinguish between MAC and MAC++. Remarkably, the performance-boosting brought by MAC++ is generally more significant than that of MAC, especially when the size of samples is low. The clear difference observed when compared to lower-quality matching data indicates the advantages of MAC++ in handling more challenging data.

Results on KITTI-LC Dataset. The results of RANSAC [18], TEASER++ [43], SC^2-PCR [11], and MAC [57] are reported in Table 5. We apply the nearest neighbor search to generate initial correspondences and keep the magnitude to 5000 by random sampling. FPFH descriptor is used for correspondence generation.

MAC++ outperforms others under all the settings. For registration problems, it is more meaningful to investigate more challenging data. MAC++ achieves 3.64% and 1.35% RR improvements over MAC on the ultra-challenging 10-20m and 20-30m data, respectively. As such, *MAC++ is more competitive on more challenging datasets*.

5.4. Analysis Experiments

Parameter Analysis. We conduct experiments to evaluate the impact of different parameter settings on MAC++'s performance. The most critical parameter is the sample size during hypothesis generation. As mentioned in Sect. 4.2, MAC++ finds K_1 cliques for each seed correspondence, placing them in the pool, and retains the top-K_2 highest-scoring cliques for subsequent evaluation. In Table 7, K_1 is set to 1, 5, 10, and 20. In Table 8, K_2 is set to 100, 200, 500, 1000, and 2000. Results for different descriptor settings on the 3DMatch and 3DLoMatch datasets are presented.

	K_1	RR(%)	RE(°)	TE(cm)
FPFH Setting	1	84.29 / 43.80	1.77 / 4.25	5.76 / 11.12
	5	83.86 / 44.53	1.74 / 4.18	5.74 / 11.06
	10	83.73 / 44.30	1.74 / 4.12	5.66 / 11.02
	20	83.67 / 44.19	1.75 / 4.17	5.70 / 11.04
FCGF Setting	1	93.22 / 61.03	1.94 / 3.94	6.32 / 11.18
	5	93.10 / 60.42	1.98 / 3.93	6.33 / 10.96
	10	93.22 / 61.03	1.97 / 3.93	6.34 / 11.17
	20	93.35 / 60.64	2.00 / 3.87	6.41 / 11.00

Table 7. Parameter analysis for K_1.

	K_2	RR(%)	RE(°)	TE(cm)
FPFH Setting	100	84.17 / 44.19	2.09 / 4.13	6.79 / 11.08
	200	83.92 / 44.53	1.74 / 4.14	5.65 / 11.14
	500	83.73 / 44.24	1.72 / 4.14	5.59 / 11.06
	1000	83.73 / 44.30	1.73 / 4.16	5.58 / 11.16
	2000	83.86 / 44.36	1.74 / 4.14	5.64 / 11.14
FCGF Setting	100	93.16 / 60.89	1.94 / 3.86	6.30 / 10.91
	200	93.04 / 61.03	1.94 / 3.89	6.28 / 11.09
	500	93.22 / 61.09	1.96 / 3.90	6.34 / 11.03
	1000	93.16 / 60.42	1.96 / 3.88	6.24 / 10.90
	2000	93.22 / 60.86	1.96 / 3.90	6.34 / 10.98

Table 8. Parameter analysis for K_2.

As shown in Table 7, MAC++ performs optimally on 3DMatch when K_1 is set to 1. This is because both descriptors can generate matches with a relatively high inlier ratio on 3DMatch, ensuring that each seed only needs to select one clique to guarantee sufficient correct hypotheses. On 3DLoMatch with FPFH, performance initially improves and then declines as K_1 increases, indicating that too few correct hypotheses are sampled when K_1 is small, and too many invalid hypotheses are sampled when K_1 is large. Table 8 shows that MAC++'s performance remains stable even with a small K_2, and fewer hypotheses reduce time consumption. Overall, MAC++ is robust to parameter changes.

Comparative Efficiency Results. Table 9 displays the time efficiency and memory consumption of various high-performing methods. Correspondences with different magnitude settings are generated with PREDATOR on the 3DMatch dataset. Each method's time efficiency was tested over ten rounds, with mean and standard deviation results reported. All tests were performed using only the CPU.

MAC++ introduces advanced hypothesis generation and evaluation mechanisms, slightly increasing computational

# Corr.	250	500	1000	2500	5000
PointDSC[3]	32.24±0.81 3531.46	78.38±0.89 3538.26	240.46±2.18 3582.57	1401.97±12.24 3634.22	5504.11±10.32 3736.10
TEASER++[43]	6.40±1.88 1631.92	6.68±0.66 1634.77	16.74±1.21 2029.22	104.24±0.53 2266.84	484.93±1.87 2484.83
SC^2-PCR[11]	19.34±0.63 448.01	63.23±0.55 453.18	215.98±1.24 508.40	1282.73±4.05 621.27	5210.17±8.30 690.22
MAC[57]	7.32±0.55 15.59	23.32±0.38 17.43	56.45±1.41 23.49	282.67±7.83 52.79	3259.38±12.66 150.86
MAC++	79.60±0.49 25.26	116.61±1.43 26.79	117.20±1.17 30.95	230.20±2.79 60.87	4620.39±23.23 187.12

Table 9. Efficiency results. Time (ms): upper value, memory (MB): lower value.

	# Hypo.	RANSAC	MAC	MAC++
FPFH Setting	100	0.76 / 0.05	50.67 / 12.22	**79.63 / 32.81**
	200	1.50 / 0.09	89.27 / 17.59	**154.19 / 56.20**
	500	3.68 / 0.21	162.41 / 23.32	**358.71 / 104.52**
	1000	7.39 / 0.41	217.32 / 26.02	**643.55 / 151.19**
	2000	14.90 / 0.81	254.13 / 29.31	**1088.26 / 190.57**
FCGF Setting	100	10.45 / 1.25	61.94 / 30.47	**91.86 / 57.02**
	200	20.76 / 2.52	119.20 / 55.57	**181.95 / 111.18**
	500	51.74 / 6.21	269.06 / 109.32	**437.05 / 261.07**
	1000	103.65 / 12.43	456.18 / 156.11	**811.74 / 475.10**
	2000	208.24 / 24.80	669.32 / 202.12	**1489.00 / 803.07**

Table 10. Comparison of the number of correct hypotheses on the 3DMatch/3DLoMatch datasets.

	n	MAC	MAC++
FPFH Setting	1	96.30 / 75.01	**97.10 / 77.88**
	5	90.39 / 52.27	**94.58 / 65.30**
	10	85.95 / 41.66	**92.79 / 59.01**
	20	78.68 / 30.71	**90.76 / 50.81**
	50	67.10 / 14.94	**87.37 / 40.48**
FCGF Setting	1	**98.46 / 91.42**	98.27 / 86.19
	5	97.10 / **83.32**	**97.72** / 83.27
	10	96.43 / 77.93	**97.54 / 81.08**
	20	94.70 / 70.47	**96.98 / 78.16**
	50	91.13 / 56.37	**95.69 / 73.55**

Table 11. Results of judging the correctness of hypotheses on 3DMatch/3DLoMatch.

overhead but remaining lightweight for input sizes below 2.5k. The hypothesis clustering part increases the runtime due to the default setting of having the number of filtered hypotheses (K_2) equal to the input correspondences. Our parameter analysis indicates that MAC++ can reduce the time consumption by decreasing the number of hypotheses (K_1, K_2) without compromising performance. Overall, MAC++ is both time and memory-efficient in most cases.

Quality of Hypotheses. We evaluate the quality of the generated hypotheses by comparing the hypotheses from RANSAC, MAC, and MAC++ with the ground-truth transformation, respectively. The results are shown in Table 10.

Compared to MAC, the number of correct hypotheses generated by MAC++ multiplies under all settings, demonstrating the effectiveness of the proposed hypothesis generation method with voted maximal clique pool.

Performance Upper Bound. Given an ideal hypothesis evaluation metric that permits a point cloud pair can be aligned whenever correct hypotheses are generated, we perform experiments in this configuration to assess the performance upper bound of MAC++. We vary the judging threshold for the number of generated correct hypotheses (a point cloud pair is considered alignable if at least n hypotheses are correct) and report the results in Table 11.

In general, MAC++ still has a large room for improvement. When n equals 1, the performance upper-bound under the FCGF setting is lower than MAC, because MAC++ enforces additional constraints during clique sampling, which happens to filter out some correct maximal cliques. However, MAC++ significantly mines more correct cliques in general as n increases. In addition, the gap becomes significant under the FPFH setting and the upper bound of MAC++ exceeds MAC consistently, indicating a significant increase in the number of correct hypotheses generated by most point cloud pairs.

	NC	VMP	One-shot	GCCI	RR(%)	RE(°)	TE(cm)	#CH
MAC	✓		✓		41.34	3.71	14.51	22.94
MAC++ w.o. VMP	✓			✓	67.65	2.94	10.81	22.94
MAC++ w.o. GCCI		✓	✓		42.47	3.68	14.56	47.90
MAC++		✓		✓	68.45	2.95	10.80	47.90

Table 12. Module effectiveness experiments. **NC**: Node-guided clique selection. **VMP**: Hypothesis generation with voted maximal clique pool. **One-shot**: Original one-shot hypothesis evaluation. **GCCI**: GCCI-based progressive hypothesis evaluation. #CH: the number of correct hypotheses.

Module Effectiveness Analysis. In this section, we validate the effectiveness of each module in MAC++ using the data corresponding to the "$\leq 1\%$" setting in Table 1. We establish MAC as our baseline, and the experimental results are shown in Table 12.

The following observations can be made: 1) When the module VMP is removed, MAC++ shows a notable 26.31% increase over the baseline MAC, indicating the effectiveness of GCCI in finding the correct solutions. 2) When removing the GCCI module, the registration recall increases by 1.13% compared to MAC. The VMP module leads to a correct hypotheses increase of 109%, indicating its effectiveness in selecting valid hypotheses. 3) MAC++ achieves the highest performance when both VMP and GCCI are implemented. These findings validate the effectiveness of each novel contribution in MAC++.

6. Conclusion

In this paper, we have presented MAC++, a learning-free robust estimator that delves into maximal cliques for 3D registration from noisy correspondences. Our method is competitive against existing competitors on several challenging benchmarks. We will explore the feasibility of MAC++ to 2D-3D registration issues in the future.

Limitation. In Table 9, the overall runtime increases. We hope to design a more suitable algorithm to improve the time efficiency of the hypothesis clustering part.

Acknowledgments. This work is supported in part by National Natural Science Foundation of China (No. 62372377).

References

[1] Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11753–11762, 2021. 6, 7

[2] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6359–6367, 2020. 1

[3] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 15859–15869. IEEE, 2021. 1, 2, 6, 8

[4] Daniel Barath and Jiří Matas. Graph-cut ransac. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6733–6741, 2018. 1, 2

[5] Alvaro Parra Bustos and Tat-Jun Chin. Guaranteed outlier removal for point cloud registration with correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2868–2882, 2017. 1, 2

[6] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016. 1

[7] Anh-Quan Cao, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Pcam: Product of cross-attention matrices for rigid registration of point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13229–13238, 2021. 1

[8] Guangyan Chen, Meiling Wang, Li Yuan, Yi Yang, and Yufeng Yue. Rethinking point cloud registration as masking and reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17717–17727, 2023. 2, 1

[9] Suyi Chen, Hao Xu, Ru Li, Guanghui Liu, Chi-Wing Fu, and Shuaicheng Liu. Sira-pcr: Sim-to-real adaptation for 3d point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14394–14405, 2023. 1

[10] Wen Chen, Haoang Li, Qiang Nie, and Yun-Hui Liu. Deterministic point cloud registration via novel transformation decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6348–6356, 2022. 1, 2

[11] Zhi Chen, Kun Sun, Fan Yang, Lin Guo, and Wenbing Tao. SC^2-PCR++: Rethinking the generation and selection for efficient and robust point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 3, 6, 7, 8, 1

[12] Yuxin Cheng, Zhiqiang Huang, Siwen Quan, Xinyue Cao, Shikun Zhang, and Jiaqi Yang. Sampling locally, hypothesis globally: accurate 3d point cloud registration with a ransac variant. *Visual Intelligence*, 1(1):20, 2023. 1

[13] Sungjoon Choi, Qian-Yi Zhou, and Vladlen Koltun. Robust reconstruction of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5556–5565, 2015. 1

[14] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8958–8966, 2019. 6, 7, 1

[15] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2514–2523. IEEE, 2020. 1, 2, 6

[16] Ondrej Chum and Jiri Matas. Matching with prosac-progressive sample consensus. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 220–226. IEEE, 2005. 1, 2

[17] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 998–1005. IEEE, 2010. 6

[18] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1, 2, 6, 7

[19] Kexue Fu, Shaolei Liu, Xiaoyuan Luo, and Manning Wang. Robust point cloud registration framework based on deep graph matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8893–8902, 2021. 1, 2

[20] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 6

[21] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5545–5554, 2019. 1

[22] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 216–224, 2018. 5

[23] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4267–4276, 2021. 5, 6, 7, 1

[24] Xiaoshui Huang, Guofeng Mei, and Jian Zhang. Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11366–11374, 2020. 5

[25] Haobo Jiang, Zheng Dang, Zhen Wei, Jin Xie, Jian Yang, and Mathieu Salzmann. Robust outlier rejection for 3d registration with variational bayes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1148–1157, 2023. 1

[26] Junha Lee, Seungwook Kim, Minsu Cho, and Jaesik Park. Deep hough voting for robust global registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15994–16003, 2021. 1, 2

[27] Jiayuan Li. A practical o (n2) outlier removal method for point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 2

[28] Jiayuan Li, Pengcheng Shi, Qingwu Hu, and Yongjun Zhang. Qgore: Quadratic-time guaranteed outlier removal for point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1, 2

[29] Yang Li and Tatsuya Harada. Lepard: Learning partial point cloud matching in rigid and deformable scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5554–5564, 2022. 1

[30] Jiuming Liu, Guangming Wang, Zhe Liu, Chaokang Jiang, Marc Pollefeys, and Hesheng Wang. Regformer: An efficient projection-aware transformer network for large-scale point cloud registration. *arXiv preprint arXiv:2303.12384*, 2023. 2

[31] Guofeng Mei, Hao Tang, Xiaoshui Huang, Weijie Wang, Juan Liu, Jian Zhang, Luc Van Gool, and Qiang Wu. Unsupervised deep probabilistic approach for partial point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13611–13620, 2023. 1

[32] Ajmal S Mian, Mohammed Bennamoun, and Robyn A Owens. A novel representation and feature matching algorithm for automatic pairwise registration of range images. *International Journal of Computer Vision*, 66(1):19–40, 2006. 5, 1

[33] G Dias Pais, Srikumar Ramalingam, Venu Madhav Govindu, Jacinto C Nascimento, Rama Chellappa, and Pedro Miraldo. 3dregnet: A deep neural network for 3d point registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7193–7203. IEEE, 2020. 1, 2

[34] Zhijian Qiao, Zehuan Yu, Binqian Jiang, Huan Yin, and Shaojie Shen. G3reg: Pyramid graph-based global registration using gaussian ellipsoid model. *arXiv preprint arXiv:2308.11573*, 2023. 5, 1

[35] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11143–11152, 2022. 2, 6, 7, 1

[36] Siwen Quan and Jiaqi Yang. Compatibility-guided sampling consensus for 3-d point cloud registration. *IEEE Transactions on Geoscience and Remote Sensing*, 58(10):7380–7392, 2020. 2

[37] Chunlin Ren, Qingshan Xu, Shikun Zhang, and Jiaqi Yang. Hierarchical prior mining for non-local multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3611–3620, 2023. 1

[38] Radu Bogdan Rusu and Steve Cousins. 3d is here: Point cloud library (pcl). In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1–4. IEEE, 2011. 4

[39] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3212–3217. IEEE, 2009. 1, 2, 6

[40] Jingnan Shi, Heng Yang, and Luca Carlone. Robin: a graph-theoretic approach to reject outliers in robust estimation using invariants. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 13820–13827. IEEE, 2021. 1

[41] Qiao Wu, Jiaqi Yang, Kun Sun, Chu'ai Zhang, Yanning Zhang, and Mathieu Salzmann. Mixcycle: Mixup assisted semi-supervised 3d single object tracking with cycle consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13956–13966, 2023. 1

[42] Fan Yang, Lin Guo, Zhi Chen, and Wenbing Tao. One-inlier is first: Towards efficient position encoding for point cloud registration. *Advances in Neural Information Processing Systems*, 35:6982–6995, 2022. 1

[43] Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2020. 1, 6, 7, 8

[44] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: A globally optimal solution to 3d icp point-set registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2241–2254, 2015. 1, 2, 6

[45] Jiaqi Yang, Zhiguo Cao, and Qian Zhang. A fast and robust local descriptor for 3d point cloud registration. *Information Sciences*, 346:163–179, 2016. 6

[46] Jiaqi Yang, Zhiqiang Huang, Siwen Quan, Zhaoshuai Qi, and Yanning Zhang. Sac-cot: Sample consensus by sampling compatibility triangles in graphs for 3-d point cloud registration. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2021. 1, 2, 6

[47] Jiaqi Yang, Zhiqiang Huang, Siwen Quan, Qian Zhang, Yanning Zhang, and Zhiguo Cao. Toward efficient and robust metrics for ransac hypotheses and 3d rigid registration. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(2):893–906, 2021. 2, 4, 5

[48] Jiaqi Yang, Xiyu Zhang, Shichao Fan, Chunlin Ren, and Yanning Zhang. Mutual voting for ranking 3d correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 6

[49] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018. 5

[50] Runzhao Yao, Shaoyi Du, Wenting Cui, Aixue Ye, Feng Wen, Hongbo Zhang, Zhiqiang Tian, and Yue Gao. Hunter: Exploring high-order consistency for point cloud registration with severe outliers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2

[51] Zi Jian Yew and Gim Hee Lee. Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6677–6686, 2022. 1

[52] Pengyu Yin, Shenghai Yuan, Haozhi Cao, Xingyu Ji, Shuyang Zhang, and Lihua Xie. Segregator: Global point cloud registration with semantic and geometric cues. *arXiv preprint arXiv:2301.07425*, 2023. 1

[53] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. *Advances in Neural Information Processing Systems*, 34, 2021. 2, 6, 7, 1

[54] Hao Yu, Zheng Qin, Ji Hou, Mahdi Saleh, Dongsheng Li, Benjamin Busam, and Slobodan Ilic. Rotation-invariant transformer for point cloud matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5384–5393, 2023. 1

[55] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1802–1811, 2017. 5, 1

[56] Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2174–2181. IEEE, 2015. 1

[57] Xiyu Zhang, Jiaqi Yang, Shikun Zhang, and Yanning Zhang. 3d registration with maximal cliques. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 17745–17754, 2023. 1, 2, 3, 4, 6, 7, 8

[58] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Proceedings of the European Conference on Computer Vision*, pages 766–782. Springer, 2016. 6