IRAv3: Hierarchical Incremental Rotation Averaging on the Fly

Xiang Gao, Hainan Cui, Menghan Li, Zexiao Xie, and Shuhan Shen, *Member, IEEE*

***Abstract*— We present IRAv3, which is built upon the state-of-the-art rotation averaging method, IRA++, to push this fundamental task in 3D computer vision one step further. The key observation of this letter lies in that during IRA++, the community detection-based Epipolar-geometry Graph (EG) clustering is preemptive and permanent, which is not relevant to the follow-up rotation averaging task and limits the upper bound of absolute rotation estimation accuracy. In this letter, however, the EG clustering is performed along with the cluster-wise absolute rotation estimation, *i.e.* instead of pre-determination, the affiliation of each vertex to which EG cluster is determined "on the fly", and the EG clustering finishes until all the vertices find the clusters they belong to, together with their absolute rotations estimated (in the local coordinate systems of the clusters they attached). By this way, a rotation averaging-targeted and -friendly EG clustering is obtained, which facilitates the rotation averaging task in turn. Experiments on both 1DSfM and KITTI odometry datasets demonstrate the effectiveness of our proposed IRAv3 on large-scale rotation averaging problems and its advantages over its previous works (IRA and IRA++) and other state of the arts.**

***Index Terms*— Global structure from motion, large-scale rotation averaging, on-the-fly epipolar-geometry graph clustering.**

I. Introduction

ROTATION averaging [1] is acknowledged to recover the absolute camera orientations provided the relative rotation measurements, which is of significance in many 3D computer vision tasks, such as Structure from Motion (SfM) [2], [3], [4], [5], Simultaneous Localization And Mapping (SLAM) [6], [7], [8], Visual Odometry (VO) [9], [10], [11], and Pose Graph Optimization (PGO) [12], [13], [14]. Though having been widely investigated in recent years [15], [16], [17], [18], [19], [20], [21], [22], existing rotation averaging methods still expose their limitations in accuracy, efficiency, and scalability when dealing with high-noise, spare-connectivity, or large-scale Epipolar-geometry Graph (EG) as input.

In order to perform large-scale rotation averaging with high accuracy and robustness, Incremental Rotation Averaging (IRA) [23] and its upgradation, IRA++ [24], were presented recently. In IRA, the absolute rotations are estimated in an incremental manner, during which the relative rotation measurement outliers are filtered simultaneously. Focusing on the IRA's limitations of low scalability in large-scale situations and insufficient focus on EG structure, IRA++ was proposed. During IRA++, the input EG is clustered into several sub-graphs, and IRA is performed both on each low-level inner-sub-EG and the high-level inter-sub-EG. Then, the absolute rotations are globally aligned and optimized to achieve advantages in both accuracy and efficiency compared with IRA.

Similar to other community-based rotation averaging [25], [26], SfM [27], [28], and Bundle Adjustment (BA) [29], [30] methods, IRA++ also follows the (first) divide and (then) conquer strategy, *i.e.* the input EG is first divided into several clusters, and then inner- and inter-community rotation averaging are performed, followed by absolute rotation global alignment and optimization [24]. However, it should be noted that: 1) once the EG clustering is done, this community-based structure remains unchanged during the entire follow-up process of IRA++; 2) the graph clustering methods [31], [32] employed in IRA++ and other methods [25], [26], [27], [28], [29], [30] only consider the connectivity of the EG, which means the generated community-based structure is not highly relevant to the primary tasks (rotation averaging, SfM, or BA). Even though sometimes the EG edges are weighted by the inlier feature match number [25], [26], [28], [30] of fundamental/essential matrix estimation [33], [34], the imprecise correlation between the inlier feature match number and the relative rotation measurement accuracy [23] still leads to imperfect EG clustering and thus limits the upper bound of absolute rotation estimation accuracy.

Based on the above observation, a novel rotation averaging pipeline, which is built upon IRA++ and termed as IRAv3, is presented in this letter. Its most significant characteristic lies in that the EG clustering is performed along with the rotation estimation task, which makes its clustering process possible to consider the relative rotation measurement error on each EG edge directly and obtain a rotation averaging-targeted and -friendly EG clustering. Specifically, several camera triplets

Manuscript received 1 August 2022; revised 30 September 2022; accepted 22 October 2022. Date of publication 26 October 2022; date of current version 5 April 2023. This work was supported in part by the National Science Foundation of China under Grant 62003319, Grant 62076026, and Grant 42076192; and in part by the Shandong Provincial Natural Science Foundation under Grant ZR2020QF075. This article was recommended by Associate Editor S. Du. *(Corresponding author: Shuhan Shen.)*

Xiang Gao, Hainan Cui, and Shuhan Shen are with the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China, also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the CASIA-SenseTime Research Group, Beijing 100190, China (e-mail: xiang.gao@ia.ac.cn; hncui@nlpr.ia.ac.cn; shshen@nlpr.ia.ac.cn).

Menghan Li and Zexiao Xie are with the College of Engineering, Ocean University of China, Qingdao 266100, China (e-mail: limenghan@stu.ouc.edu.cn; xiezexiao@ouc.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2022.3217151.

Digital Object Identifier 10.1109/TCSVT.2022.3217151

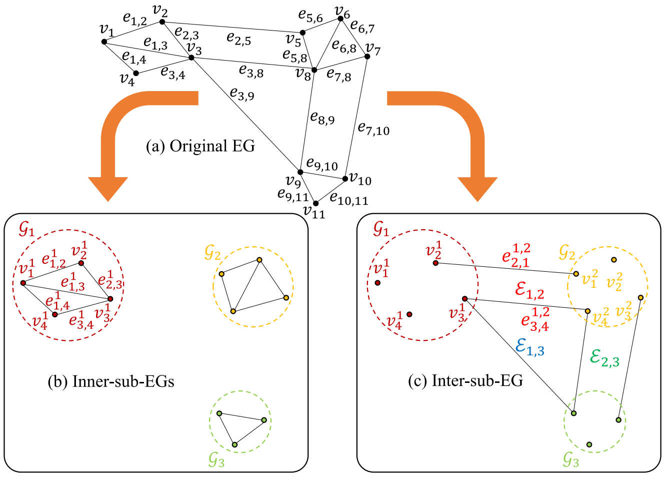

Fig. 1. Toy example on different levels of Epipolar-geometry Graph (EG) structures. (a) The original EG $\mathcal{G}$: $\mathcal{G} = \left\{\mathcal{V} = \{v_i\}, \mathcal{E} = \{e_{i,j}\}\right\}$. (b) The Inner-sub-EGs $\{\mathcal{G}_p\}$: $\mathcal{G}_p = \left\{\mathcal{V}_p = \{v_i^p\}, \mathcal{E}_p = \{e_{i,j}^p\}\right\}$. And (c) The Inter-sub-EG $\mathcal{G}_c$: $\mathcal{G}_c = \left\{\mathcal{V}_c = \{\mathcal{G}_p\}, \mathcal{E}_c = \{\mathcal{E}_{p,q}\} = \{\{e_{i,j}^{p,q}\}\}\right\}$. See the main text for more details.

are selected in the input EG and served as the cluster seeds at first. Then, during the incremental absolute rotation computation process, not only which vertex is added, but also which cluster it is added to is dynamically determined. By this way, the cluster affiliation and the cluster-level absolute rotation estimation of each vertex are simultaneously achieved. After that, Voting-based Single Rotation Averaging (VSRA) proposed in Gao et al. [26], which solves the single rotation averaging problem by random selection, supporter acquisition, and rotation optimization, is performed for inter-cluster rotation estimation, IRA is carried out for inter-sub-EG rotation averaging, and rotation global alignment and optimization are sequentially conducted to produce the final rotation averaging result, which is similar to that of IRA++.

II. Brief Review of IRA++

As our proposed IRAv3 in this letter is an upgradation of IRA++ [24], it is briefly reviewed before introducing IRAv3. For a better understanding of both IRA++ and IRAv3, several concepts are also described beforehand. For the rotation averaging problem, EG is a graph structure where its vertices are the absolute rotations to be estimated and its edges are the known relative rotation measurements. After EG clustering, each cluster also contains several vertices and edges. The EG cluster could be regarded as an EG as well and is termed as inner-sub-EG. In addition, if each EG cluster is treated as a cluster-level vertex, and EG edge set between each EG cluster pair are treated as an cluster-level edge, a high-level graph structure is constructed and is termed as inter-sub-EG. Fig. 1 gives a toy example to illustrate different levels of EG structures. The IRA++ mainly contains five steps: 1) Community detection-based EG clustering [30] is carried out on the input EG to obtain several inner-sub-EGs. 2) IRA-based rotation averaging [23] is performed on each inner-sub-EG to estimate the cameras' absolute rotations in it w.r.t. its local coordinate system. 3) VSRA-based rotation estimation [26] is conducted to estimate the relative rotation between the local coordinate systems of each inner-sub-EG pair. 4) IRA-based rotation averaging is performed again on the inter-sub-EG to estimate the absolute rotation of each inner-sub-EG's local coordinate system. 5) And finally, rotation global alignment and optimization is conducted to first globally align all the absolute rotations in the input EG to a uniform coordinate system, then globally optimize them to produce the final rotation averaging result.

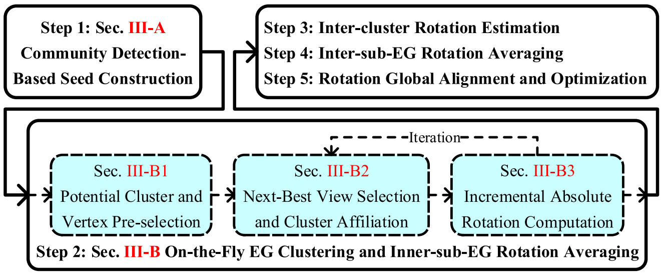

Fig. 2. Pipeline of our proposed IRAv3. The input of IRAv3 is the relative rotation measurements and the feature match number $\left\{\boldsymbol{R}_{ij}, n_{ij} \middle| e_{ij} \in \mathcal{E}\right\}$, and its output is the estimated absolute rotations $\left\{\boldsymbol{R}_i^* \middle| v_i \in \mathcal{V}\right\}$. It mainly contains five key steps, including 1) community detection-based seed construction, 2) on-the-fly Epipolar-geometry Graph (EG) clustering and inner-sub-EG rotation averaging (**core step of IRAv3**), 3) Voting-based Single Rotation Averaging (VSRA)-based inter-cluster rotation estimation, 4) IRA-based inter-sub-EG rotation averaging, and 5) rotation global alignment and optimization. The last three steps of IRAv3 are similar to those of IRA++ [24].

III. Proposed Method

The same as IRA and IRA++, the input of our proposed IRAv3 is the relative rotation measurements and the feature match number of each matched image pair, which is denoted as $\left\{\boldsymbol{R}_{ij}, n_{ij} \middle| e_{ij} \in \mathcal{E}\right\}$, where $\mathcal{E}$ is the edge set of the input EG. Its output, denoted as $\left\{\boldsymbol{R}_i^* \middle| v_i \in \mathcal{V}\right\}$, where $\mathcal{V}$ is the vertex set of the input EG, is the estimated absolute rotations of all the cameras in the input EG. Fig. 2 illustrates the pipeline of our proposed IRAv3, which also has five key steps: 1) Community detection-based seed construction is performed to construct several cluster seeds for the follow-up EG clustering. 2) On-the-fly EG clustering and inner-sub-EG rotation averaging are conducted to iteratively assign vertex to certain EG cluster and estimate its cluster-level absolute rotation, which is the core step of IRAv3 and contains three sub-steps, potential cluster and vertex pre-selection, Next-Best View (NBV) selection and cluster affiliation, and incremental absolute rotation computation. And steps 3) to 5) are similar to those of IRA++, and readers may refer to Gao et al. [24] for more details. In the following, steps 1) and 2) are described in detail.

A. Community Detection-Based Seed Construction

In order to perform on-the-fly hierarchical incremental rotation averaging, several vertex pairs or triplets should be selected to serve as cluster seeds for the follow-up EG clustering and cluster-level rotation averaging. Though it has been shown in IRA [23] that compared with vertex pair, vertex triplet is preferred for seed construction, two other tough problems still arise here: which and how many camera

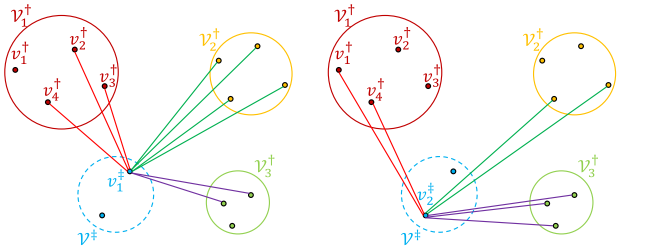

Fig. 3. A toy example on potential cluster and vertex pre-selection. Suppose the feature match number of all the edges in this figure is identical, for the first vertex $v_1^{\ddagger}$ in $\mathcal{V}^{\ddagger}$, it has 3, 4, and 2 edges connected with $\mathcal{V}_1^{\dagger}$, $\mathcal{V}_2^{\dagger}$, and $\mathcal{V}_3^{\dagger}$, respectively; while for the second vertex $v_2^{\ddagger}$ in $\mathcal{V}^{\ddagger}$, it has 2, 2, and 3 edges connected with $\mathcal{V}_1^{\dagger}$, $\mathcal{V}_2^{\dagger}$, and $\mathcal{V}_3^{\dagger}$, respectively. As a result, according to Eq. 1, $v_1^{\ddagger} \leftrightarrow \mathcal{V}_2^{\dagger}$ has the largest pre-selection reward among all the vertex-cluster pairs and is selected with the highest priority for the following procedure. See the main text for more details.

TABLE I

META-DATA OF THE 1DSFM AND KITTI ODOMETRY DATASETS FOR EVALUATION, WHERE $|\mathcal{V}|$, $|\mathcal{E}|$, $|\mathcal{V}^*|$, AND n_c INDICATE THE NUMBER OF VERTICES AND EDGES IN THE INPUT EG, THE NUMBER OF VERTICES WITH GROUND-TRUTH ABSOLUTE ROTATIONS, AND THE NUMBER OF CLUSTERS OBTAINED BY THE CONSTRAINED GRAPH CLUSTERING (CGC) IN SEC. III-A, RESPECTIVELY; $\tilde{r}_{ij}$ AND $\bar{r}_{ij}$ ARE THE MEDIAN AND MEAN ERRORS IN DEGREES OF THE RELATIVE ROTATION MEASUREMENTS, RESPECTIVELY; AND $\tilde{n}_{ij}$ AND $\bar{n}_{ij}$ ARE THE MEDIAN AND MEAN VALUES OF THE FEATURE MATCH NUMBER, RESPECTIVELY

Data	$\|\mathcal{V}\|(\|\mathcal{V}^*\|)$	$\|\mathcal{E}\|$	n_c	$\tilde{r}_{ij}$	$\bar{r}_{ij}$	$\tilde{n}_{ij}$	$\bar{n}_{ij}$
ALM	627(577)	97206	7	2.78°	9.09°	105	192
ELS	247(227)	20297	5	2.89°	12.50°	106	160
GDM	742(677)	48144	11	12.30°	33.33°	73	144
MDR	394(341)	23784	6	9.34°	29.30°	61	128
MND	474(450)	52424	7	1.67°	7.51°	180	310
NYC	376(332)	20680	5	4.22°	14.14°	80	167
PDP	354(338)	24710	6	1.81°	8.38°	87	128
PIC	2508(2152)	319257	29	4.93°	19.09°	56	97
ROF	1134(1084)	70187	12	2.97°	13.83°	65	188
TOL	508(472)	23863	7	2.60°	11.58°	81	220
TFG	5433(5058)	680012	58	3.01°	8.62°	71	109
USQ	930(789)	25561	13	3.61°	9.02°	87	150
VNC	918(836)	103550	13	2.59°	11.26°	229	408
YKM	458(437)	27729	7	2.68°	11.16°	112	245
00	9082(9082)	503591	90	0.33°	3.60°	244	296
01	2202(2202)	108280	23	0.35°	3.14°	151	210
02	9322(9322)	475500	93	0.39°	10.70°	197	278
03	1602(1602)	23014	16	0.14°	0.45°	594	709
04	542(542)	20000	6	0.41°	6.09°	59	179
05	5522(5522)	313804	55	0.24°	3.29°	263	314
06	2202(2202)	108805	22	0.18°	2.39°	245	295
07	2202(2202)	121605	22	0.30°	2.63°	289	336
08	8142(8142)	357817	82	0.22°	0.73°	282	340
09	3182(3182)	158892	31	0.36°	6.65°	173	250
10	2402(2402)	128547	24	0.28°	3.23°	201	281

triplets should be selected for IRAv3. In this letter, we resort to community detection method to solve them. Specifically, similar to IRA++, the input EG, denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, is first clustered into several smaller sub-EGs using the Constrained Graph Clustering (CGC) [30], which are denoted as $\left\{\mathcal{G}_p = \{\mathcal{V}_p, \mathcal{E}_p\} \middle| p = 1, 2, \ldots, n_c\right\}$, and n_c is the number of clusters. The same to IRA++ [24], the max cluster size is set to 100 in this letter. Then, for each sub-EG, the Local Optimization-based Initial Triplet Selection (LOITS) proposed in IRA is performed to construct the cluster seeds of IRAv3. The selected vertex triplets and the corresponding absolute rotations after local optimization are denoted as $\left\{v_{r^*}^p, v_{s^*}^p, v_{t^*}^p \in \mathcal{V}_p\right\}$ and $\left\{(\boldsymbol{R}_{r^*}^p)^*, (\boldsymbol{R}_{s^*}^p)^*, (\boldsymbol{R}_{t^*}^p)^*\right\}$, where $v_{r^*}^p$ and $(\boldsymbol{R}_{r^*}^p)^*$ indicate the r^*-th vertex and its optimized absolute rotation of the p-th sub-EG, $\mathcal{G}_p$, respectively. By this way, the number of cluster seeds is adaptively determined, and the constructed cluster seeds are with high reliability as the effectiveness of LOITS and CGC has been demonstrated in IRA and IRA++, respectively. It should be noted that different from IRA++, which uses the community-based structure obtained from CGC for the following inner- and inter-sub EG rotation averaging, community detection is employed in IRAv3 for seed construction only.

B. On-the-Fly EG Clustering and Inner-Sub-EG RA

Based on the constructed cluster seeds, the unregistered vertices are iteratively affiliated to their most closely associated clusters together with their cluster-level absolute rotations estimated on the fly. To achieve this, 1) the association of each currently unregistered vertex to each cluster is first computed for potential cluster and vertex pre-selection; 2) then, which unregistered vertex is added next and which cluster it is added to is dynamically determined based on its weighted supporting set used in IRA [23]; 3) subsequently, the cluster-level absolute rotations are locally or globally optimized following the procedure of incremental absolute rotation computation proposed in IRA. Steps 2) and 3) are iteratively conducted until all vertices are cluster-affiliated and rotation-estimated.

1) Potential Cluster and Vertex Pre-Selection: In order to accelerate NBV selection and cluster affiliation, several potentially added vertices and their potentially affiliated clusters are pre-selected. Suppose the vertex sets of the currently absolute rotations estimated or not are denoted as $\left\{\mathcal{V}_p^{\dagger} \middle| p = 1, 2, \ldots, n_c\right\}$ and $\mathcal{V}^{\ddagger}$, which means $\left\{\mathcal{V}_p^{\dagger}\right\} \cup \mathcal{V}^{\ddagger} = \mathcal{V}$. For each vertex $v_m^{\ddagger}$ in $\mathcal{V}^{\ddagger}$, one could obtain the edge set between it and $\mathcal{V}_p^{\dagger}$, which is denoted as $\mathcal{E}_m^p$. Then, the pre-selection reward (the larger, the better) of $v_m^{\ddagger}$ corresponding to the vertex set $\mathcal{V}_p^{\dagger}$ could be computed by:

$$R_p\left(v_m^{\ddagger}, \mathcal{V}_p^{\dagger}\right) = \frac{1}{\left|\mathcal{V}_p^{\dagger}\right|} \sum_{\substack{v_n^{\dagger} \in \mathcal{V}_p^{\dagger} \\ e_{m,n}^p \in \mathcal{E}_m^p}} n_{m,n}^p, \tag{1}$$

where $\left|\mathcal{V}_p^{\dagger}\right|$ is used to encourage a more balanced EG clustering in cluster size and $n_{m,n}^p$ is the feature match number of the edge $e_{m,n}^p$. Note that the above pre-selection reward is computed between each vertex in $\mathcal{V}^{\ddagger}$ and each vertex set in $\left\{\mathcal{V}_p^{\dagger}\right\}$, which results in a reward matrix with size $\left|\mathcal{V}^{\ddagger}\right| \times n_c$. Then, n_a vertex-cluster pairs, which are denoted as $\left\{v_{m_i}^{\ddagger}, \mathcal{V}_{p_i}^{\dagger} \middle| i = 1, 2, \ldots, n_a\right\}$, with largest pre-selection rewards are pre-selected for the following NBV selection and cluster affiliation, and n_a is set to 10 in this letter. Fig. 3 gives a toy example on potential cluster and vertex pre-selection.

2) Next-Best View Selection and Cluster Affiliation: We denote the edge sets between the vertex-cluster pairs obtained

TABLE II

COMPARATIVE RESULTS IN ROTATION AVERAGING ACCURACY ON THE 1DSFM DATASET: FIRST SECOND THIRD FORTH

Data	IRLS-GM [15]	IRLS-$\ell_{\frac{1}{2}}$ [16]	MPLS [18]	OMSTs [35]	HRRA [26]	HARA [21]	NeuRoRA [17]	MSPRA [19]	RAGO [22]	IRA [23]	IRA++ [24]	IRAv3
ALM	2.12°	2.14°	1.16°	1.26°	1.03°	1.15°	1.16°	1.07°	0.88°	0.83°	0.80°	0.73°
ELS	1.08°	1.15°	0.88°	0.75°	0.59°	0.62°	0.64°	0.83°	0.46°	0.51°	0.46°	0.44°
GDM	35.83°	28.20°	9.87°	45.15°	4.04°	63.74°	2.94°	3.69°	2.68°	5.32°	2.88°	1.99°
MDR	4.52°	3.08°	1.26°	1.12°	2.54°	1.50°	1.13°	1.09°	1.03°	0.85°	0.83°	0.75°
MND	0.77°	0.71°	0.51°	0.68°	0.62°	0.51°	0.60°	0.50°	0.46°	0.51°	0.50°	0.44°
NYC	1.43°	1.40°	1.24°	1.30°	1.24°	1.37°	1.18°	1.12°	0.71°	1.00°	0.95°	0.82°
PDP	2.16°	2.62°	1.93°	1.73°	0.92°	0.92°	0.79°	0.76°	0.63°	0.90°	0.75°	0.72°
PIC	4.14°	3.12°	1.81°	1.41°	4.87°	3.22°	1.91°	1.80°	0.58°	1.67°	1.70°	1.50°
ROF	1.62°	1.70°	1.37°	1.85°	2.48°	2.42°	1.31°	1.19°	1.10°	1.51°	1.24°	1.09°
TOL	2.59°	2.45°	2.20°	2.10°	2.05°	2.36°	1.46°	1.25°	1.20°	2.45°	1.33°	1.44°
TFG	1.94°	2.03°	—	2.63°	4.88°	2.06°	2.25°	—	1.53°	3.30°	1.74°	1.49°
USQ	4.93°	4.97°	3.48°	3.83°	3.77°	4.78°	2.01°	1.85°	1.92°	4.40°	3.70°	3.27°
VNC	4.87°	4.64°	2.83°	3.30°	1.84°	1.49°	1.50°	1.10°	0.89°	1.02°	0.94°	0.86°
YKM	1.70°	1.62°	1.45°	1.55°	1.57°	1.65°	0.99°	0.91°	0.92°	1.57°	1.38°	1.36°
Rank	10.64	10.29	7.46	8.21	8.00	8.50	5.93	4.23	2.00	5.93	3.36	1.93

above as $\left\{\mathcal{E}_{m_i}^{p_i}\middle|i=1,2,\ldots,n_a\right\}$. For each vertex-cluster pair $v_{m_i}^{\ddagger}\leftrightarrow\mathcal{V}_{p_i}^{\dagger}$, the absolute rotation of $v_{m_i}^{\ddagger}$, denoted as $\boldsymbol{R}_{m_i}^{p_i\ddagger}$, in the local coordinate system of $\left\{\boldsymbol{R}_n^{p_i\dagger}\middle|v_n^{p_i\dagger}\in\mathcal{V}_{p_i}^{\dagger}\right\}$ could be estimated with each edge $e_{m_i,n}^{p_i}$ in $\mathcal{E}_{m_i}^{p_i}$ by:

$$\left\{\boldsymbol{R}_{m_i}^{n,p_i\ddagger}=\boldsymbol{R}_{m_i,n}^{p_i}\boldsymbol{R}_n^{p_i\dagger}\middle|v_n^{p_i\dagger}\in\mathcal{V}_{p_i}^{\dagger},e_{m_i,n}^{p_i}\in\mathcal{E}_{m_i}^{p_i}\right\}. \tag{2}$$

Ideally, all the elements in $\left\{\boldsymbol{R}_{m_i}^{n,p_i\ddagger}\right\}$ should be identical as they denote the same rotation. However, this would not happen practically as there are inevitable measurement and estimation errors in $\left\{\boldsymbol{R}_{m_i,n}^{p_i}\right\}$ and $\left\{\boldsymbol{R}_n^{p_i\dagger}\right\}$, respectively. To perform NBV selection and cluster affiliation, the selection reward for each pre-computed cluster-level absolute rotation in $\left\{\boldsymbol{R}_{m_i}^{n,p_i\ddagger}\right\}$, which is indicated by its supporting set in weighted manner, should be computed first:

$$R_s\left(v_{m_i}^{\ddagger},v_n^{p_i\dagger}\right)=\frac{1}{\left|\mathcal{V}_{p_i}^{\dagger}\right|}\sum_{\substack{v_{n'}^{p_i\dagger}\in\mathcal{V}_{p_i}^{\dagger}\\ e_{m_i,n'}^{p_i}\in\mathcal{E}_{m_i}^{p_i}}} n_{m_i,n'}^{p_i}\cos\left(d_\theta\left(\boldsymbol{R}_1,\boldsymbol{R}_2\right)\right), \tag{3}$$

where $d_\theta(\boldsymbol{R}_1,\boldsymbol{R}_2)$ is the angular distance [23] between $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$, $\boldsymbol{R}_1=\boldsymbol{R}_{m_i,n'}^{p_i}$, and $\boldsymbol{R}_2=\boldsymbol{R}_{m_i}^{n,p_i\ddagger}\left(\boldsymbol{R}_{n'}^{p_i\dagger}\right)^{\mathrm{T}}$. Then, the representative absolute rotation $\boldsymbol{R}_{m_i}^{n^*,p_i\ddagger}$ in $\left\{\boldsymbol{R}_{m_i}^{n,p_i\ddagger}\right\}$ is selected by:

$$n^*=\arg\max\left\{R_s\left(v_{m_i}^{\ddagger},v_n^{p_i\dagger}\right)\middle|v_n^{p_i\dagger}\in\mathcal{V}_{p_i}^{\dagger},e_{m_i,n}^{p_i}\in\mathcal{E}_{m_i}^{p_i}\right\}. \tag{4}$$

And finally, the vertex-cluster pair for NBV selection and cluster affiliation is determined by:

$$i^*=\arg\max\left\{R_s\left(v_{m_i}^{\ddagger},v_{n^*}^{p_i\dagger}\right)\middle|i=1,2,\ldots,n_a\right\}, \tag{5}$$

which means by NBV selection and cluster affiliation, the vertex of $v_{m_{i^*}}^{\ddagger}$ is selected and added to the cluster of $\mathcal{V}_{p_{i^*}}^{\dagger}$, with its cluster-level absolute rotation initialized as $\boldsymbol{R}_{m_{i^*}}^{n^*,p_{i^*}\ddagger}$.

3) Incremental Absolute Rotation Computation: After NBV selection and cluster affiliation, the cluster-level absolute rotations are further optimized using the incremental absolute rotation computation procedure proposed in IRA [23]. Specifically, given the selected NBV $v_{m_{i^*}}^{\ddagger}$, and its affiliated cluster $\mathcal{V}_{p_{i^*}}^{\dagger}$ and initialized rotation $\boldsymbol{R}_{m_{i^*}}^{n^*,p_{i^*}\ddagger}$, local (optimizing the NBV's rotation $\boldsymbol{R}_{m_{i^*}}^{n^*,p_{i^*}\ddagger}$ only) or global (optimizing the NBV's rotation and all the currently estimated rotations of its affiliated cluster $\left\{\boldsymbol{R}_{m_{i^*}}^{n^*,p_{i^*}\ddagger}\right\}\bigcup\left\{\boldsymbol{R}_n^{p_{i^*}\dagger}\middle|v_n^{p_{i^*}\dagger}\in\mathcal{V}_{p_{i^*}}^{\dagger}\right\}$ together) optimization is performed with the inlier edge set in weighted manner. Details on the inlier acquisition and optimization weighting methods could be found in IRA [23]. Note that the local/global optimization switching condition here is individually considered for each EG cluster, *i.e.* Weighted Global Optimization (WGO) is conducted on a EG cluster *iff* the size of that cluster has increased by a certain ratio r since its last WGO, otherwise Weighted Local Optimization (WLO) is performed, and r is set to 40% in this letter. In addition, similar to IRA, Re-Rotation Averaging (RRA) is performed after each cluster-level WGO.

TABLE III

COMPARATIVE RESULTS IN ROTATION AVERAGING ACCURACY ON THE KITTI ODOMETRY DATASET: FIRST SECOND THIRD FORTH

Data	IRLS-GM [15]	IRLS-$\ell_{\frac{1}{2}}$ [16]	HARA [21]	IRA [23]	IRA++ [24]	IRAv3
00	0.85°	0.66°	0.63°	0.71°	0.71°	0.69°
01	1.03°	1.12°	1.15°	0.96°	0.98°	0.82°
02	52.54°	24.49°	0.63°	1.00°	0.98°	0.84°
03	0.53°	0.31°	0.30°	0.42°	0.42°	0.39°
04	0.34°	0.24°	0.28°	0.33°	0.33°	0.22°
05	2.41°	0.37°	0.34°	0.46°	0.50°	0.38°
06	0.61°	0.57°	0.37°	0.44°	0.45°	0.36°
07	0.67°	0.46°	0.46°	0.59°	0.58°	0.33°
08	1.00°	0.63°	0.68°	0.68°	0.67°	0.60°
09	1.64°	0.51°	0.52°	0.97°	0.97°	0.46°
10	0.84°	0.44°	0.40°	0.61°	0.61°	0.38°
Rank	5.82	2.91	2.36	3.82	3.82	1.64

IV. EVALUATION

In this section, our proposed IRAv3 is evaluated and compared with some other state-of-the-art rotation averaging methods. The evaluation is performed on a PC with I7-9750H CPU and 32GB RAM and on 2 commonly-used datasets, 1DSfM [36] and KITTI odometry [37], which contain 14 and 11 test data, respectively. The metadata of these two datasets is listed in Table I, and the absolute rotation ground truths of them are provided by the SfM results of Bundler [38] (1DSfM)

TABLE IV
PARAMETER SETTING EVALUATION RESULTS ON THE NUMBER OF PRE-SELECTED VERTEX-CLUSTER PAIRS n_a AND THE INCREASING RATIO r

Data	n_a: estimation accuracy (left) and elapsed time (right)					r: estimation accuracy (left) and elapsed time (right)				
	2	5	10	15	20	10%	20%	40%	80%	160%
ALM	0.75° 11s	0.74° 12s	0.73° 12s	0.76° 13s	0.69° 13s	0.73° 17s	0.73° 14s	0.73° 12s	0.72° 13s	0.75° 13s
ELS	0.42° 3s	0.43° 3s	0.44° 4s	0.44° 4s	0.44° 4s	0.44° 4s	0.44° 3s	0.44° 4s	0.45° 3s	0.44° 3s
GDM	2.07° 8s	2.02° 8s	1.99° 10s	1.96° 10s	1.82° 11s	1.99° 11s	1.99° 9s	1.99° 10s	1.97° 9s	2.03° 8s
MDR	0.94° 4s	0.87° 4s	0.75° 5s	0.76° 4s	0.80° 5s	0.74° 5s	0.74° 5s	0.75° 5s	0.84° 5s	0.86° 4s
MND	0.45° 7s	0.45° 7s	0.44° 7s	0.45° 7s	0.45° 7s	0.44° 8s	0.44° 7s	0.44° 7s	0.45° 7s	0.44° 7s
NYC	0.86° 4s	0.79° 4s	0.82° 5s	0.80° 5s	0.83° 5s	0.83° 6s	0.83° 5s	0.82° 5s	0.82° 4s	0.80° 4s
PDP	0.83° 5s	0.73° 5s	0.72° 5s	0.72° 5s	0.69° 5s	0.74° 7s	0.74° 5s	0.72° 5s	0.71° 5s	0.70° 5s
PIC	1.54° 39s	1.48° 42s	1.50° 44s	1.54° 44s	1.52° 46s	1.47° 47s	1.47° 43s	1.50° 44s	1.53° 40s	1.56° 39s
ROF	1.83° 11s	1.26° 11s	1.09° 12s	1.05° 12s	1.24° 12s	0.97° 14s	1.02° 12s	1.09° 12s	1.09° 11s	1.09° 11s
TOL	2.09° 5s	1.87° 5s	1.44° 5s	1.41° 5s	1.40° 5s	1.49° 6s	1.49° 6s	1.44° 5s	1.47° 5s	1.53° 5s
TFG	1.58° 110s	1.58° 113s	1.49° 112s	1.48° 112s	1.58° 113s	1.39° 127s	1.43° 118s	1.49° 112s	1.63° 118s	1.59° 108s
USQ	3.30° 6s	3.42° 6s	3.27° 8s	3.75° 6s	3.20° 6s	3.27° 7s	3.27° 7s	3.27° 8s	4.06° 6s	3.39° 6s
VNC	0.84° 12s	0.79° 12s	0.86° 14s	0.83° 13s	0.83° 14s	0.85° 16s	0.85° 14s	0.86° 14s	0.84° 12s	0.84° 11s
YKM	1.36° 6s	1.37° 6s	1.36° 6s	1.31° 6s	1.29° 6s	1.35° 7s	1.35° 7s	1.36° 6s	1.32° 6s	1.32° 6s
00	0.70° 214s	0.71° 215s	0.69° 225s	0.69° 250s	0.68° 252s	0.68° 282s	0.69° 252s	0.69° 225s	0.68° 217s	0.67° 208s
01	0.80° 19s	0.87° 20s	0.82° 20s	0.73° 20s	0.74° 21s	0.76° 28s	0.73° 23s	0.82° 20s	0.87° 18s	0.95° 17s
02	0.86° 223s	0.81° 233s	0.84° 238s	0.81° 263s	0.79° 266s	0.78° 286s	0.85° 269s	0.84° 238s	0.89° 228s	0.90° 220s
03	0.39° 6s	0.39° 6s	0.39° 6s	0.39° 6s	0.39° 6s	0.39° 9s	0.39° 9s	0.39° 6s	0.39° 6s	0.39° 5s
04	0.21° 5s	0.21° 5s	0.22° 5s	0.22° 5s	0.21° 5s	0.22° 6s	0.21° 6s	0.22° 5s	0.22° 5s	0.23° 5s
05	0.45° 83s	0.43° 84s	0.38° 85s	0.33° 91s	0.36° 94s	0.40° 113s	0.39° 97s	0.38° 85s	0.38° 83s	0.38° 81s
06	0.33° 20s	0.35° 19s	0.36° 21s	0.34° 20s	0.35° 21s	0.44° 29s	0.44° 23s	0.36° 21s	0.43° 18s	0.46° 17s
07	0.31° 20s	0.32° 21s	0.33° 21s	0.29° 22s	0.29° 22s	0.29° 30s	0.32° 25s	0.33° 21s	0.34° 20s	0.29° 18s
08	0.58° 160s	0.63° 162s	0.60° 166s	0.62° 180s	0.55° 185s	0.54° 198s	0.56° 179s	0.60° 166s	0.66° 161s	0.65° 156s
09	0.51° 33s	0.51° 33s	0.46° 34s	0.44° 35s	0.43° 36s	0.43° 46s	0.40° 39s	0.46° 34s	0.51° 32s	0.51° 29s
10	0.36° 22s	0.39° 22s	0.38° 22s	0.39° 23s	0.30° 24s	0.37° 32s	0.39° 26s	0.38° 22s	0.40° 21s	0.37° 19s

and the output of the GPS/IMU localization unit (KITTI odometry), respectively. The accuracy evaluation method and measure in this letter are the same as those in IRA [23] and IRA++ [24]. Specifically, a best relative rotation between the absolute rotation estimations and ground truths is estimated and used to align these two rotation sets. And then the median alignment error in degree is used as the accuracy evaluation measure. Based on the evaluation environment, dataset, method, and measure introduced above, comparative experiments in rotation averaging accuracy and parameter setting evaluation of our proposed IRAv3 are performed.

A. Comparative Experiments

Comparative experiments are performed on the 1DSfM dataset between our proposed IRAv3 and other 11 representative rotation averaging methods, including 3 robust loss-based methods, IRLS-GM [15], IRLS-$\ell_{\frac{1}{2}}$ [16], and MPLS [18], 3 outlier filtering-based methods, ÖMSTs [35], HRRA [26], and HARA [21], 3 deep learning-based methods, NeuRoRA [17], MSPRA [19], and RAGO [22], and 2 previous works of IRAv3, IRA [23] and IRA++ [24]. In addition, comparative experiments are also performed on the KITTI odometry dataset, where the first 3 and last 2 of the above 11 comparative methods are compared with IRAv3. The comparative experimental results on the 1DSfM and KITTI odometry datasets are shown in Table II and Table III, respectively. The top-four results on each test data are highlighted, and in order to perform a comprehensive comparison, the ranks of the comparative methods on each test data are averaged and shown in the last rows of these 2 tables, and the top-four methods are highlighted as well. We can see from the tables that on both 1DSfM and KITTI odometry datasets, our proposed IRAv3 not only achieves upgradation with respect to its previous works, IRA [23] and IRA++ [24], but also performs better than the very recent state-of-the-art methods, HARA [21] and RAGO [22], which demonstrates the effectiveness in accuracy, robustness, and scalability of the IRAv3 proposed in this letter.

B. Parameter Setting Evaluations

Parameter setting evaluations are conducted on two important parameters of our proposed IRAv3, the number of pre-selected vertex-cluster pairs n_a defined in Sec. III-B.1 and the increasing ratio for WL/GO switching r defined in Sec. III-B.3. During the evaluation, we change the value of one parameter and fix the other to perform rotation averaging on each test data, and the results on both estimation accuracy and efficiency are shown in Table IV. The left and right half part of the table are the results for n_a and r, respectively. Note that when performing the parameter setting evaluation on n_a, the value of r is set to 40%, and when performing the evaluation on r, the value of n_a is set to 10. We can see from the table that the estimation accuracy of IRAv3 is not quite sensitive to either n_a or r. And the estimation accuracy tends to get better, while the estimation efficiency tends to decrease as n_a increases or r decreases. As a result, to balance the estimation accuracy and efficiency, the values of n_a and r are set to 10 and 40% in this letter, respectively. It should also be noted from the table that for either n_a or r, there is no globally optimal value for all test data to achieve best rotation averaging accuracy, but a global value setting for a specific parameter on all test data is recommended to improve the robustness of the method.

V. CONCLUSION

By observing that the recently presented community-based rotation averaging method IRA++ suffers from EG clustering preemption and permanence, it is further upgraded to IRAv3 in this letter. Different to that in IRA++, the EG clustering in IRAv3 is performed along with the rotation estimation task, which makes its clustering process possible to consider the relative rotation measurement error on each EG edge directly and achieve a rotation averaging-targeted and -friendly EG clustering. Our proposed IRAv3 is evaluated on both 1DSfM and KITTI odometry datasets to demonstrate its effectiveness on rotation averaging problems and its advantages over the state of the arts. In the future, we intend to introduce the on-the-fly clustering scheme proposed in this letter to some other high related topics, such as SfM [27], [28], and BA [29], [30].

References

[1] R. Hartley, J. Trumpf, Y. Dai, and H. Li, "Rotation averaging," *Int. J. Comput. Vis.*, vol. 103, no. 3, pp. 267–305, Jul. 2013.

[2] H. Cui, S. Shen, X. Gao, and Z. Hu, "Batched incremental structure-from-motion," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 205–214.

[3] X. Gao, S. Shen, L. Zhu, T. Shi, Z. Wang, and Z. Hu, "Complete scene reconstruction by merging images and laser scans," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3688–3701, Oct. 2020.

[4] Q. Dong, X. Gao, H. Cui, and Z. Hu, "Robust camera translation estimation via rank enforcement," *IEEE Trans. Cybern.*, vol. 52, no. 2, pp. 862–872, Feb. 2022.

[5] X. Gao, L. Zhu, B. Fan, H. Liu, and S. Shen, "Incremental translation averaging," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jun. 16, 2022, doi: 10.1109/TCSVT.2022.3183631.

[6] A. P. Bustos, T.-J. Chin, A. Eriksson, and I. Reid, "Visual SLAM: Why bundle adjust?" in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 2385–2391.

[7] F. Tang, Y. Wu, X. Hou, and H. Ling, "3D mapping and 6D pose computation for real time augmented reality on cylindrical objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 9, pp. 2887–2899, Sep. 2020.

[8] X. Shao, L. Zhang, T. Zhang, Y. Shen, and Y. Zhou, "MOFISSLAM: A multi-object semantic SLAM system with front-view, inertial and surround-view sensors for indoor parking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4788–4803, 2022.

[9] I. Shim, T.-H. Oh, J.-Y. Lee, J. Choi, D.-G. Choi, and I. S. Kweon, "Gradient-based camera exposure control for outdoor mobile platforms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 6, pp. 1569–1583, Jun. 2019.

[10] C. K. Chng, A. Parra, T. J. Chin, and Y. Latif, "Monocular rotational odometry with incremental rotation averaging and loop closure," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2020, pp. 1–8.

[11] D. Liu, A. Parra, and T.-J. Chin, "Spatiotemporal registration for event-based visual odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4937–4946.

[12] G. Moreira, M. Marques, and J. P. Costeira, "Fast pose graph optimization via Krylov–Schur and Cholesky factorization," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1897–1905.

[13] Z. J. Yew and G. H. Lee, "Learning iterative robust transformation synchronization," in *Proc. Int. Conf. 3D Vis. (3DV)*, Dec. 2021, pp. 1206–1215.

[14] J. Yang, Z. Huang, S. Quan, Q. Zhang, Y. Zhang, and Z. Cao, "Toward efficient and robust metrics for RANSAC hypotheses and 3D rigid registration," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 2, pp. 893–906, Feb. 2022.

[15] A. Chatterjee and V. M. Govindu, "Efficient and robust large-scale rotation averaging," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 521–528.

[16] A. Chatterjee and V. Govindu, "Robust relative rotation averaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 958–972, Apr. 2017.

[17] P. Purkait, T. J. Chin, and I. Reid, "NeuRoRA: Neural robust rotation averaging," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Aug. 2020, pp. 137–154.

[18] Y. Shi and G. Lerman, "Message passing least squares framework and its application to rotation synchronization," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 8796–8806.

[19] L. Yang, H. Li, J. A. Rahim, Z. Cui, and P. Tan, "End-to-end rotation averaging with multi-source propagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11774–11783.

[20] A. Eriksson, C. Olsson, F. Kahl, and T.-J. Chin, "Rotation averaging with the chordal distance: Global minimizers and strong duality," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 256–268, Jan. 2021.

[21] S. H. Lee and J. Civera, "HARA: A hierarchical approach for robust rotation averaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15777–15786.

[22] H. Li, Z. Cui, S. Liu, and P. Tan, "RAGO: Recurrent graph optimizer for multiple rotation averaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15787–15796.

[23] X. Gao, L. Zhu, Z. Xie, H. Liu, and S. Shen, "Incremental rotation averaging," *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1202–1216, Apr. 2021.

[24] X. Gao, L. Zhu, H. Cui, Z. Xie, and S. Shen, "IRA++: Distributed incremental rotation averaging," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4885–4892, Jul. 2022.

[25] H. Cui, X. Gao, S. Shen, and Z. Hu, "HSfM: Hybrid structure-from-motion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2393–2402.

[26] X. Gao, J. Luo, K. Li, and Z. Xie, "Hierarchical RANSAC-based rotation averaging," *IEEE Signal Process. Lett.*, vol. 27, pp. 1874–1878, 2020.

[27] H. Cui, S. Shen, X. Gao, and Z. Hu, "CSFM: Community-based structure from motion," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4517–4521.

[28] S. Zhu et al., "Very large-scale global SfM by distributed motion averaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4568–4577.

[29] R. Zhang et al., "Distributed very large scale bundle adjustment by global camera consensus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 291–303, Feb. 2020.

[30] L. Zhou et al., "Stochastic bundle adjustment for efficient and scalable 3D reconstruction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 364–379.

[31] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 70, no. 6, Dec. 2004, Art. no. 066111.

[32] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 11, pp. 1944–1957, Nov. 2007.

[33] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 580–593, Jun. 1997.

[34] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–770, Jun. 2004.

[35] H. Cui, S. Shen, W. Gao, H. Liu, and Z. Wang, "Efficient and robust large-scale structure-from-motion via track selection and camera prioritization," *ISPRS J. Photogramm. Remote Sens.*, vol. 156, pp. 202–214, Oct. 2019.

[36] K. Wilson and N. Snavely, "Robust global translations with 1DSfM," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2014, pp. 61–75.

[37] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[38] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from internet photo collections," *Int. J. Comput. Vis.*, vol. 80, no. 2, pp. 189–210, 2007.