MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Parallel Rigidity Matters for Bundle Adjustment

Lalit, Manam; Govindu, Venu

TR2026-053 May 16, 2026

Abstract

Bundle adjustment is a long-standing problem in computer vision that solves for camera parameters and 3D point coordinates from 2D image observations. While there has been much work on various aspects, like adaptation to different camera models and sensors, and considerations for solving the optimization problem, in this paper, we deal with a fundamental and distinct question of the uniqueness of its solution. In particular, we examine the unique solvability of the 3D reconstruction problem using parallel rigidity theory. We design an algorithm to ensure that the topology of the bipartite graph formed by the camera-3D point relations in bundle adjustment does not result in independent scaling of the edges in its subgraphs. To tackle the generally large-sized bipartite graph, we leverage camera- camera relationships in 3D reconstruction problems for efficiency. We demonstrate the benefits of our analysis for a global structure-from-motion pipeline. Applying our pro- posed algorithm results in significantly cleaner reconstructions by removing misplaced cameras and 3D points

IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2026

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Parallel Rigidity Matters for Bundle Adjustment

Lalit Manam
Mitsubishi Electric Research Labs (MERL)
Cambridge, MA 02139, USA
manam@merl.com

Venu Madhav Govindu
Indian Institute of Science
Bengaluru, KA 560012, INDIA
venug@iisc.ac.in

Abstract

Bundle adjustment is a long-standing problem in computer vision that solves for camera parameters and 3D point coordinates from 2D image observations. While there has been much work on various aspects, like adaptation to different camera models and sensors, and considerations for solving the optimization problem, in this paper, we deal with a fundamental and distinct question of the uniqueness of its solution. In particular, we examine the unique solvability of the 3D reconstruction problem using parallel rigidity theory. We design an algorithm to ensure that the topology of the bipartite graph formed by the camera-3D point relations in bundle adjustment does not result in independent scaling of the edges in its subgraphs. To tackle the generally large-sized bipartite graph, we leverage camera-camera relationships in 3D reconstruction problems for efficiency. We demonstrate the benefits of our analysis for a global structure-from-motion pipeline. Applying our proposed algorithm results in significantly cleaner reconstructions by removing misplaced cameras and 3D points.

1. Introduction

Bundle adjustment (BA) [74] is fundamental to 3D reconstruction in computer vision, with wide usage in structure-from-motion (SfM) [58, 64, 66, 70–72, 83] and simultaneous localization and mapping (SLAM) [20, 21, 27, 39, 60]. BA optimizes for camera parameters (intrinsics and extrinsics) and 3D point coordinates to be consistent with 2D observations or image features [74]. The camera extrinsics (a rotation and a translation per camera) and 3D point coordinates are defined with respect to a global coordinate system. The camera-3D point relationships result in a **bipartite graph** (e.g., see Fig. 1a), which is also known as a visibility graph [19, 54, 79, 85]. The existing literature has focused on different aspects of BA like adaptation for different camera models [32, 41, 44, 55, 59, 67], integration of sensors like GPS or IMU [15, 43], photometric constraints [4, 67], optimization techniques [1, 2, 13, 45, 46], uncertainty considerations [26, 56, 81], robustness to outlier observations [86, 87], distributed computing frameworks [25, 61] among others. Many efficient implementations of BA are publicly available [3, 20, 40, 47, 84]. In this paper, we use parallel rigidity theory to examine a fundamental and distinct question in bundle adjustment that has remained unaddressed in the literature: under what conditions is a bundle adjustment solution unique?

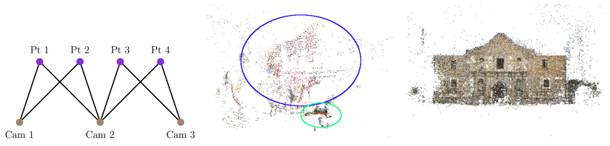
Parallel rigidity: The issue of *parallel rigidity* [5] (or the solvability problem [5, 22, 24, 76, 77]) occurs in a graph embedded in $\mathbb{R}^d$ with $d \geq 2$, which aims to recover vectors on the nodes ($\mathbb{R}^d$) given relative directions (unit norm vectors) on the edges ($\mathbb{S}^{d-1}$) as the input. The mismatch in the input ($\mathbb{S}^{d-1}$) and output ($\mathbb{R}^d$) spaces necessitates estimating edge scales. In a **parallel rigid (PR) graph**, none of its subgraphs can have edge scales independent of any other subgraph, thus ensuring a unique solution (up to the gauge freedom of an origin and a global scale factor). The parallel rigidity of a graph is dependent on its topology as well as on the input directions on the edges.

Parallel rigidity in BA: The bipartite BA graph, denoted as $\mathcal{G}_{BA}$, consists of cameras and 3D points as nodes and 2D observations[1] on the image as edges. Each 2D observation corresponds to a direction measured by the camera in its local coordinate system[2] because a pinhole camera is a projective mapping. Hence, the consideration of parallel rigidity emerges in BA when camera translations and 3D point coordinates (both corresponding to nodes in $\mathcal{G}_{BA}$ and in $\mathbb{R}^3$) are simultaneously optimized, with 2D observations (edges in $\mathcal{G}_{BA}$ and effectively in $\mathbb{S}^2$) as the input. We highlight that parallel rigidity in BA deals with the unique solvability of camera translations and 3D point coordinates (up to $Sim(3)$ gauge freedom in BA), and not of other camera parameters. Some thumb rules for 3D

---

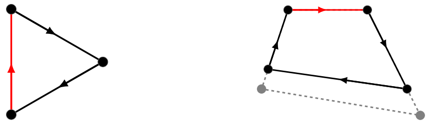
[1] 2D observations correspond to the projection of 3D points on images.

[2] For example, given a camera with linear intrinsics $\mathbf{K} \in \mathbb{R}^{3\times3}$, a direction ($\mathbb{S}^2$) in its local coordinate system can be obtained from a 2D observation $\mathbf{o} \in \mathbb{R}^2$ by $\mathbf{K}^{-1}\tilde{\mathbf{o}}/||\mathbf{K}^{-1}\tilde{\mathbf{o}}||$, where $\tilde{\mathbf{o}} \in \mathbb{R}^3$ is the homogeneous representation of $\mathbf{o}$.



(a) (b) (c)

Figure 1. Impact of parallel rigidity issue in BA. (a) Example of a bipartite BA graph with cameras (Cam) and 3D points (Pt) as nodes, and 2D observations as edges. (b) A 3D reconstruction with misplaced 3D points of a facade shown in the blue ellipse, which should be present inside the green ellipse. (c) Zoomed view of the green ellipse that contains the facade. Misplaced 3D points are scaled independent of other 3D points, which can be identified using parallel rigidity principles.



(a) (b)

Figure 2. Toy examples of graphs in $\mathbb{R}^2$ with input directions (arrows) depicted as edges that form loops. The output we seek is the set of node values satisfying the input directions. When we fix the red edge's scale, (a) has only one solution for node values (black), while (b) has infinite solutions (another solution in gray by scaling the black edges to gray ones independent of the red edge). The independent scaling in (b) occurs with any set of four directions forming loops; thus, this issue is due to its topology. Hence, (a) is a generically parallel rigid graph in $\mathbb{R}^2$ while (b) is not.

scene coverage by cameras are presented in [74, Sec. 11] to have a bipartite BA graph with "sufficient information for reliable estimation", which may indirectly ensure parallel rigidity. However, to the best of our knowledge, the bipartite BA graph has not been studied in terms of parallel rigidity and its impact on 3D reconstructions. An illustration of the impact of parallel rigidity (or lack thereof) in BA on 3D reconstructions is shown in Fig. 1.

The theory of parallel rigidity is developed when all input directions are represented in a global coordinate system. 2D observations provide directions in the local coordinate system of individual cameras and can be represented in a global coordinate system only when camera rotations and intrinsics are known[3], which is seldom true in practice for BA. Hence, we want the bipartite BA graph to be a **generically parallel rigid (GPR) graph** [5, Sec. 2.2 & 3.2], which examines independent scaling of subgraphs based only on the graph topology and dimension of the space. Fig. 2 provides toy examples in $\mathbb{R}^2$.

Our contributions: In this paper, we analyze generic parallel rigidity of the bipartite BA graph, which, to the best of our knowledge, has not been dealt with before in the literature. Although there are randomized tests to infer whether a graph is GPR [5, 63], our practical interest lies in extracting its maximal GPR subgraphs. In BA, this ensures that when any such subgraph is solved, a unique solution of camera translations and 3D point coordinates is obtained. Specifically, we aim to analyze the topology of the bipartite BA graph through its 4-length loops since a 4-length loop has the smallest possible loop length in any bipartite graph and is also GPR in $\mathbb{R}^3$ [5, Prop. 9]. The nodes in the bipartite BA graph consist of cameras and 3D points, whose count is huge for large-scale problems. Instead of directly analyzing its generic parallel rigidity, we check it through a **viewgraph**, which consists of camera-camera relationships in terms of matched 2D observations. We design an algorithm to check for 4-length loops in the bipartite BA graph implicitly via the viewgraph and efficiently merge them if they together form a GPR graph. We demonstrate the value of our analysis using GLOMAP [64], a recent SfM pipeline based on global approaches [16, 17, 29–31, 58, 62, 80, 91]. Applying our algorithm results in significantly cleaner reconstructions by removing misplaced cameras and 3D points occurring due to the independent scaling of subgraphs. Moreover, we remove observations in the bipartite BA graph that do not have a match in the viewgraph, significantly increasing reconstructed 3D points.

2. Literature Review

Bundle adjustment: A large body of literature on BA focuses on optimizing the non-linear least squares cost, which is generally the reprojection error. Levenberg-Marquardt algorithm [42, 53] and conjugate gradient method [33] with preconditioning [35] are popular optimization routines for solving BA [2, 12–14, 45, 46]. The bipartite graph topology of the input allows the usage of Schur's complement trick [68], which is employed by almost every optimization routine for BA. For most BA problems, connectivity in the bipartite graph is sparse, which is efficiently utilized by sparse solvers [18, 65, 82]. These properties of the bipartite BA graph persist irrespective of the camera model used [32, 41, 44, 55, 59, 67], usage of additional sensors [4, 67], modifying the optimization cost with photometric constraints [4, 67] or robustness to outliers [86, 87]. In this paper, we also exploit the topology of the bipartite BA graph to extract its GPR subgraphs.

Parallel rigidity: Parallel rigidity has been of interest to many communities, including computer vision [6, 63] for translation averaging [7, 29, 48, 49, 51, 62], robotics [38],

---

[3] Continuing with the previous footnote example, if $\mathbf{R} \in \mathbb{SO}(3)$ is the rotation of the camera, a direction in the global coordinate system can be obtained as $\mathbf{R}^{-1}\mathbf{K}^{-1}\tilde{\mathbf{o}}/||\mathbf{K}^{-1}\tilde{\mathbf{o}}||$.

computer-aided design [69] and decision control [22, 24, 75, 88]. There are two approaches to define parallel rigidity: node-based [22, 24, 38, 69] via vector values on the nodes, and edge-based [6, 37, 75] through cycles in a graph. Theoretical findings on GPR graphs have been developed in [11, 23, 36, 37, 57, 76–78, 88, 89], which include conditions on graph topology. Results on GPR bipartite graphs, in terms of number of nodes and edges, are in [11, 76, 77]. A different notion of solvability, having similarities with parallel rigidity theory [8], is relevant to recover camera matrices from fundamental matrices [8–10, 73]. Readers are referred to [5] for an excellent survey on parallel rigidity.

3. Connecting Parallel Rigidity and BA

Parallel rigidity in $\mathbb{R}^3$: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph, where each node $i \in \mathcal{V}$ represents a vector $\mathbf{x}_i \in \mathbb{R}^3$ to be estimated and each edge $(i, j) \in \mathcal{E}$ represents a relative direction $\mathbf{u}_{ij} \in \mathbb{S}^2$ from node i to j. Then, the consistency relation on each edge (i, j) is given as

$$
\begin{aligned}
& (\mathbf{x}_j - \mathbf{x}_i) \parallel \mathbf{u}_{ij} && (1)\\
\iff & \mathbf{u}_{ij} \times (\mathbf{x}_j - \mathbf{x}_i) = \mathbf{0} && (2)\\
\iff & (\mathbf{x}_j - \mathbf{x}_i) = \alpha_{ij}\mathbf{u}_{ij} && (3)
\end{aligned}
$$

where $\times$ refers to the cross product and $\alpha_{ij} \in \mathbb{R}$. We note that Eqns. 1, 2 and 3 are equivalent. Collecting Eqn. 3 for all edges gives us an expression similar to [5, Eqn. 4], as

$$(\mathbf{B} \otimes \mathbf{I}_3)\,\mathbf{x} = \left(\mathbf{I}_{|\mathcal{E}|} \odot \mathbf{U}\right) \boldsymbol{\alpha}, \tag{4}$$

with $\otimes$ and $\odot$ denoting Kronecker and Khatri-Rao[4] products, respectively, $\mathbf{B} \in \mathbb{R}^{|\mathcal{E}| \times |\mathcal{V}|}$ being the incidence matrix of $\mathcal{G}$, $\mathbf{U} \in \mathbb{R}^{3 \times |\mathcal{E}|}$ with $\mathbf{u}_{ij}$ as its columns, $\mathbf{I}_d$ is the $d \times d$ identity matrix, and $\mathbf{x}$ and $\boldsymbol{\alpha}$ are vectors with stacked $\mathbf{x}_i$ and α_{ij}, respectively. For a connected graph, knowing any one of $\mathbf{x}$ or $\boldsymbol{\alpha}$ gives the other one uniquely, but in our case, both are unknowns. Parallel rigidity determines whether a unique solution exists for Eqn. 4. Equivalently, it tells us whether some α_{ij}'s are not dependent on other α_{ij}'s, thus checking for independent scaling of subgraphs.

Eqn. 2 provides a useful way to characterize parallel rigidity without explicitly dealing with scales. Let $[\mathbf{a}]_\times$ denote the skew-symmetric matrix associated with the cross product with $\mathbf{a} \in \mathbb{R}^3$. Let $\mathbf{S_U} \in \mathbb{R}^{3|\mathcal{E}| \times 3|\mathcal{E}|}$ be a block diagonal matrix containing all $[\mathbf{u}_{ij}]_\times$. Collecting Eqn. 2 for all edges gives us the following expression as in [5, Eqn. 19]

$$\mathbf{S_U}\,(\mathbf{B} \otimes \mathbf{I}_3)\,\mathbf{x} = \mathbf{0}. \tag{5}$$

The graph $\mathcal{G}$ is **parallel rigid (PR)** in $\mathbb{R}^3$ [5, Sec. 2.1] if and only if $rank(\mathbf{S_U}\,(\mathbf{B} \otimes \mathbf{I}_3)) = 3|\mathcal{V}| - 4$. We have infinitely many solutions when $rank(\mathbf{S_U}\,(\mathbf{B} \otimes \mathbf{I}_3)) < 3|\mathcal{V}| - 4$. We note that when all directions satisfy Eqn. 2, $rank(\mathbf{S_U}\,(\mathbf{B} \otimes \mathbf{I}_3)) \ngtr 3|\mathcal{V}| - 4$.

Generic parallel rigidity in $\mathbb{R}^3$: In Eqn. 5, the inference about parallel rigidity depends on the input directions $\mathbf{u}_{ij}$ (via $\mathbf{S_U}$). For cases when only the graph topology is available along with the dimension of the space, generic parallel rigidity is useful. In simplified terms, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is **generically parallel rigid (GPR)** in $\mathbb{R}^3$ [5, Sec. 2.2], if there exists a set of $\mathbf{x}_i \in \mathbb{R}^3$ such that $\mathbf{u}_{ij} \in \mathbb{S}^2$ obtained from $\mathbf{x}_i$ (by satisfying Eqn. 1) leads to $rank(\mathbf{S_U}\,(\mathbf{B} \otimes \mathbf{I}_3)) = 3|\mathcal{V}| - 4$. We note that GPR is an existence check on $\mathbf{x}_i$, and thus on the directions $\mathbf{u}_{ij}$. Only the graph topology (via $\mathbf{B}$) and dimension of the space (via $\mathbf{I}_3$) are required to analyze a graph being GPR. Although GPR graphs are defined with all input directions satisfying the consistency relation (Eqn. 1), a non-GPR graph has a direct impact on the estimation when dealing with noisy directions [5, Sec. 4].

Bundle adjustment: Let the **bipartite BA graph** be represented as $\mathcal{G}_{BA} = (\{\mathcal{V}_C \cup \mathcal{V}_P\}, \mathcal{E}_{BA})$, which consists of two sets of nodes $\mathcal{V}_C$ and $\mathcal{V}_P$, and an edge set $\mathcal{E}_{BA}$. Each node $i \in \mathcal{V}_C$ represents a camera with its parameters, denoted as $\mathbf{K}_i \in \mathbb{R}^{3\times 3}, \mathbf{R}_i \in \mathbb{SO}(3), \mathbf{T}_i \in \mathbb{R}^3$ for intrinsics, rotation and translation, respectively. Each node $j \in \mathcal{V}_P$ represents a 3D point with its coordinates, denoted as $\mathbf{P}_j \in \mathbb{R}^3$. Every edge $(i, j) \in \mathcal{E}_{BA}$ represents a 2D observation $\mathbf{o}_{ij} \in \mathbb{R}^2$ of j^{th} 3D point $(j \in \mathcal{V}_P)$ from i^{th} camera $(i \in \mathcal{V}_C)$ and, henceforth, referred to as a camera-3D point or **cam-pt edge**. The standard BA optimizes the reprojection error, which is given as

$$\min_{\substack{\{\mathbf{K}_i, \mathbf{R}_i, \mathbf{T}_i\},\\ \{\mathbf{P}_j\}}} \sum_{(i,j)\in\mathcal{E}_{BA}} \rho\left(||\mathbf{o}_{ij} - \Pi(\mathbf{K}_i, \mathbf{R}_i, \mathbf{T}_i, \mathbf{P}_j)||\right), \tag{6}$$

where $\Pi(\cdot,\cdot,\cdot,\cdot)$ projects a 3D point onto a camera and $\rho(\cdot)$ denotes a robust loss function.

GPR in bundle adjustment: With the pinhole camera model for projection, the consistency relation on every edge in $\mathcal{G}_{BA}$ is given as

$$
\begin{aligned}
& \mathbf{K}_i\mathbf{R}_i\,(\mathbf{P}_j - \mathbf{T}_i) = \beta_{ij}\tilde{\mathbf{o}}_{ij}\\
\iff & \mathbf{P}_j - \mathbf{T}_i = \beta_{ij}\mathbf{R}_i^T\mathbf{K}_i^{-1}\tilde{\mathbf{o}}_{ij}\\
\iff & (\mathbf{P}_j - \mathbf{T}_i) \parallel \left(\mathbf{R}_i^T\mathbf{K}_i^{-1}\tilde{\mathbf{o}}_{ij}/||\mathbf{R}_i^T\mathbf{K}_i^{-1}\tilde{\mathbf{o}}_{ij}||\right), \qquad (7)
\end{aligned}
$$

where $\tilde{\mathbf{o}}_{ij} \in \mathbb{R}^3$ is the homogeneous form of $\mathbf{o}_{ij}$ and $\beta_{ij} \in \mathbb{R}$ is the scaling factor (dependent on $\mathbf{K}_i$, $\mathbf{R}_i$, $\mathbf{T}_i$ and $\mathbf{P}_j$) required for the projection operation. It can be seen that Eqn. 7 is similar to Eqn. 1, with its right hand side in $\mathbb{S}^2$ and the unknowns $\mathbf{P}_j, \mathbf{T}_i \in \mathbb{R}^3$. This reveals that the issue of

---

[4] In Eqn. 4, Khatri-Rao product is the column-wise Kronecker product.

parallel rigidity exists in BA whenever camera translations ($\mathbf{T}_i$) and 3D point coordinates ($\mathbf{P}_j$) are optimized with 2D observations ($\mathbf{o}_{ij}$) as the input. The right side of Eqn. 7 is unknown since camera intrinsics and rotation are also optimized. Thus, we consider the topology of $\mathcal{G}_{BA}$ and dimension of the space ($\mathbb{R}^3$) to examine whether $\mathcal{G}_{BA}$ is GPR.

We remark that the parallel rigidity issue is independent of the optimization cost and remains valid with directional inputs relating to output vectors similar to Eqn. 1. For instance, GLOMAP [64] simultaneously optimizes for $\mathbf{T}_i$ and $\mathbf{P}_j$ [64, Sec. 3.2] using directional inputs but with an optimization cost different from the reprojection error [91, Eqn. 2]. The issue exists in such scenarios as well, but has not been dealt with in [64].

4. Proposed Method

Our goal is to obtain a GPR bipartite BA graph ($\mathcal{G}_{BA}$) to ensure unique solvability by avoiding independent scaling of its subgraphs. Our method checks for loops in $\mathcal{G}_{BA}$ for the analysis. From [5, Prop. 9], we know that loops of length ≤ 4 are GPR graphs in $\mathbb{R}^3$. In a bipartite graph, the smallest possible loop length is 4. In $\mathcal{G}_{BA}$, a 4-length loop consists of two cameras and two 3D points. We are interested in the interactions between 4-length loops to extract GPR subgraphs of $\mathcal{G}_{BA}$. We use the fact that two GPR subgraphs with at least two nodes in common together form a GPR graph [37, Sec. 2]. This holds for any $\mathbb{R}^d$ with $d \geq 2$. Using this result, the following **interactions** between two GPR subgraphs of $\mathcal{G}_{BA}$ consist of two common nodes[5], and hence, together form a GPR graph:

[**Type-1**] A camera and a 3D point in common, i.e. a cam-pt edge.
[**Type-2**] Two common cameras.
[**Type-3**] Two common 3D points.

Since 4-length loops are GPR in $\mathbb{R}^3$, our aim is to find them in $\mathcal{G}_{BA}$ and merge them based on their interactions. Figs. 3a, 3c, 3e provide illustrations of interactions between two 4-length loops. In a typical 3D reconstruction pipeline, the bipartite BA graph is stored as *tracks*. Given a 3D point $h \in \mathcal{V}_P$, a **track** $\mathcal{T}_h = (\{\mathcal{V}_{CTh} \cup \{h\}\}, \mathcal{E}_{Th})$ is a subgraph of $\mathcal{G}_{BA}$ such that $\mathcal{V}_{CTh} = \{i \in \mathcal{V}_C | (i,h) \in \mathcal{E}_{BA}\}$ and $\mathcal{E}_{Th} = \{(i,h) \in \mathcal{E}_{BA}\}$. In other words, a track corresponding to a 3D point is the subgraph of $\mathcal{G}_{BA}$ obtained by only considering cam-pt edges connected to that 3D point. E.g., Fig. 3b shows colour-coded tracks, illustrating the specific subgraph of $\mathcal{G}_{BA}$ belonging to each track. We note that obtaining maximal GPR subgraphs is NP-hard [37], and our method may miss higher-order interactions, due to which the GPR subgraphs may not be maximal. However, our method works well in practice, see Sec. 5.

Analysis through tracks: Here, we discuss how to extract GPR subgraphs of $\mathcal{G}_{BA}$ naively through tracks. Let $\mathcal{T}_k$ and $\mathcal{T}_l$ be two tracks with their common cameras $\mathcal{S}_{kl} = \mathcal{V}_{CTk} \cap \mathcal{V}_{CTl}$. If $|\mathcal{S}_{kl}| \geq 2$, then any two cameras in $\mathcal{S}_{kl}$ form a 4-length loop. E.g., in Fig. 3d, tracks belonging to points P_1 and P_2, having two common cameras C_1 and C_2, form a 4-length loop $C_1 - P_1 - C_2 - P_2 - C_1$. From these 4-length loop interactions, we can maintain GPR subgraphs obtained from $\mathcal{T}_k$ and $\mathcal{T}_l$. Now, for every new track, we can obtain new 4-length loops by checking all previously considered tracks to merge or expand existing GPR subgraphs or create new ones. This requires $O(|\mathcal{V}_P|^2)$ operations for set intersections. For large-scale problems, the number of tracks (or 3D points, $|\mathcal{V}_P|$) is huge. Thus, we need an efficient method to achieve our goal.

Analysis via the viewgraph: To obtain GPR subgraphs of the bipartite BA graph $\mathcal{G}_{BA}$, we look at the camera-camera relationships in a viewgraph through which the bipartite BA graph or the tracks are constructed. Let the **viewgraph**[6] be denoted as $\mathcal{G}_{VG} = (\mathcal{V}_C, \mathcal{E}_{VG})$. $\mathcal{V}_C$ is the same set of cameras as in $\mathcal{G}_{BA}$. Every edge $(u,v) \in \mathcal{E}_{VG}$ (with $\{u,v\} \in \mathcal{V}_C$) consists of many matched 2D observations between cameras u and v, and is referred to as a camera-camera or **cam-cam edge** in this paper. Each of those matched observations ideally belongs to a 3D point and, thus, contributes to two cam-pt edges in a track. E.g., in Fig. 3d, one matched observation in the cam-cam edge $C_1 - C_2$ gives two cam-pt edges, $C_1 - P_1$ and $C_2 - P_1$. In a 3D reconstruction pipeline, a track is constructed by concatenating the matched observations across adjacent cam-cam edges (edges having common nodes) through common observations (or cam-pt edges). This motivates us to infer the topology of the bipartite BA graph $\mathcal{G}_{BA}$ through the viewgraph $\mathcal{G}_{VG}$. For each cam-cam edge, we maintain a list of tracks contributed by its matched observations. We describe our proposed method to extract generically parallel rigid subgraphs of $\mathcal{G}_{BA}$ through $\mathcal{G}_{VG}$ in the following steps.

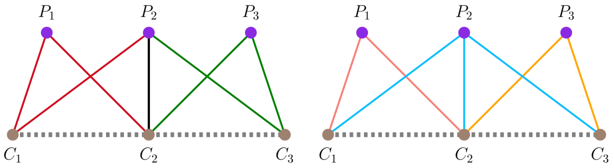
Step 1 - Identify 4-length loops of $\mathcal{G}_{BA}$ via $\mathcal{G}_{VG}$: In this step, we implicitly check for 4-length loops in the bipartite BA graph $\mathcal{G}_{BA}$ through the viewgraph $\mathcal{G}_{VG}$. We check for the conditions that form 4-length loops in $\mathcal{G}_{BA}$: every track should have at least two observations coming from two cameras and every cam-cam edge in $\mathcal{G}_{VG}$ should have at least two matched observations[7] (i.e. contributing to two tracks). E.g., in Fig. 3c, the cam-cam edge $C_1 - C_2$

---

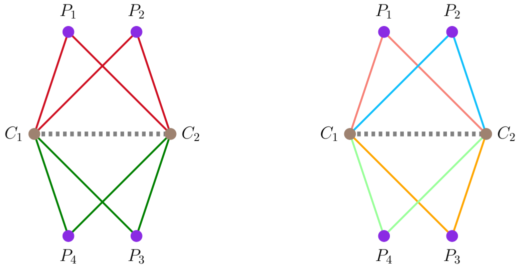
[5] An interaction with > 2 common nodes will contain at least one of Type-1,2,3 in $\mathcal{G}_{BA}$. Thus, it suffices to consider these three interactions.

[6] We provide visual descriptions of the notations related to $\mathcal{G}_{BA}$ and $\mathcal{G}_{VG}$ in Sec. A of the supplementary material.

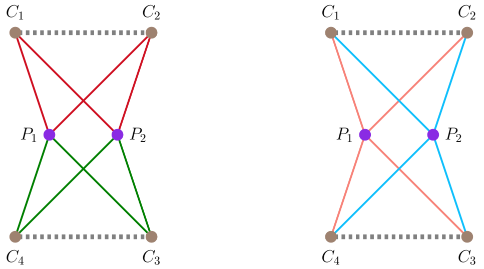
[7] We emphasize that our condition of two matched observations on a cam-cam edge in a viewgraph is applicable for generic parallel rigidity and is not the same as used for epipolar geometry estimation.



(a) Type-1 interaction via loops. (b) Type-1 interaction via tracks.



(c) Type-2 interaction via loops. (d) Type-2 interaction via tracks.



(e) Type-3 interaction via loops. (f) Type-3 interaction via tracks.

Figure 3. Different interactions between two 4-length loops in $\mathcal{G}_{BA}$ by cam-pt edges (solid lines). C_i and P_j denotes a camera and a 3D point, respectively. The interactions can either be viewed through (colour-coded) loops as in (a), (c) and (e) or can be viewed through (colour-coded) tracks as in (b), (d) and (f). For the Type-1 interaction in (a), the common cam-pt edge is shown in black. Each thick dashed gray line shows a cam-cam edge in $\mathcal{G}_{VG}$, that consists of many matched 2D observations.

contributes to at least two tracks (P_1 and P_2) and form a 4-length loop (marked in red). Similarly, in Fig. 3d, for the track corresponding to P_2 (marked in blue), there are at least two observations ($C_1 - P_2$ and $C_2 - P_2$) for it to be a part of a 4-length loop. We recursively discard the cam-cam edges in $\mathcal{G}_{VG}$ and the tracks until the conditions are met on the remaining ones.

Step 2 - Extract subgraphs of $\mathcal{G}_{BA}$ via $\mathcal{G}_{VG}$: In this step, we find cam-cam edges in $\mathcal{G}_{VG}$ that form GPR subgraphs of $\mathcal{G}_{BA}$. In Step 1, we ensured that each cam-cam edge in $\mathcal{G}_{VG}$ forms at least one 4-length loop in $\mathcal{G}_{BA}$. Each of the 4-length loops coming from a single cam-cam edge is connected by the same two cameras belonging to that cam-cam edge (e.g. red and green loops in Fig. 3c come from $C_1 - C_2$ cam-cam edge). So, all of these loops have Type-2 interactions between them. Hence, all matched observations in a cam-cam edge in $\mathcal{G}_{VG}$ together form a GPR subgraph of $\mathcal{G}_{BA}$. This gives us a way to deal with GPR subgraphs of $\mathcal{G}_{BA}$ implicitly through cam-cam edges in $\mathcal{G}_{VG}$. This drastically reduces the combinations to check since, in most practical scenarios, $|\mathcal{E}_{VG}| \ll |\mathcal{V}_P| \ll |\mathcal{E}_{BA}|$, leading to an efficient and tractable method.

Now, we want to infer which cam-cam edges in $\mathcal{G}_{VG}$ together form GPR subgraphs of $\mathcal{G}_{BA}$. Since tracks are constructed by concatenating the matched observations in the adjacent cam-cam edges, we check for at least one common track between two adjacent cam-cam edges (e.g., $C_1 - C_2$ and $C_2 - C_3$ in Fig. 3b have a common track P_2 in blue). Then, for every common track (blue in Fig. 3b), there is a Type-1 interaction between two 4-length loops in $\mathcal{G}_{BA}$ (red and green in Fig. 3a) with each loop belonging to a cam-cam edge. Hence, the two adjacent cam-cam edges with a common track together form a GPR subgraph of $\mathcal{G}_{BA}$. To efficiently keep a record of these subgraphs via cam-cam edges, we utilize the union-find data structure [28] on the cam-cam edges, whose details are provided in the supplementary material.

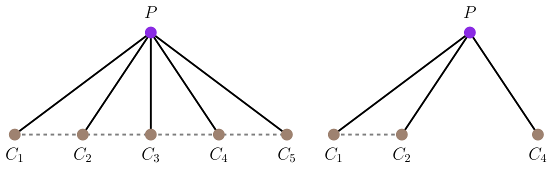
Step 3 - Merge appropriate subgraphs of $\mathcal{G}_{BA}$: In Step 2, 4-length loops of $\mathcal{G}_{BA}$ are merged according to Type-2 and then Type-1 interactions. In this step, we consider Type-3 interactions, which occur through two common 3D points. We merge any two subgraphs obtained from Step 2 if they have at least two common tracks. We note that the number of subgraphs obtained after Step 2 is small (less than 20 for the datasets used, including a dataset with $\sim$ 9000 cameras), and hence, checking for all pairs of subgraphs for common tracks is feasible.

The result of Step 3 gives us GPR subgraphs of $\mathcal{G}_{BA}$, using the three types of interactions between 4-length loops and the subgraphs formed after Step 2. We summarize our method, named **GPRBA**, in Alg. 1.

Practical considerations: In many 3D reconstruction pipelines, only a fraction of all the tracks are used in BA to reduce the problem size. Moreover, tracks are modified at different stages in the pipelines. For instance, observations are removed in the tracks when they have high residual errors in the optimization cost, for robustness to outliers, or when the triangulation angle for a 3D point is not sufficiently large. Such operations can impact the generic parallel rigidity of $\mathcal{G}_{BA}$. Hence, we apply our method whenever tracks are constructed, a subset is selected, or modified. Every invocation of our algorithm results in multiple subgraphs of $\mathcal{G}_{BA}$, whose count increases with multiple invocations. Hence, we only retain the largest subgraph after each invocation for pipeline efficiency.

Algorithm 1: GPRBA - Extract generically parallel rigid subgraphs of the bipartite graph in bundle adjustment

```
Input: Tracks $\mathcal{T}_j = (\mathcal{V}_{CTj}, \{j\}, \mathcal{E}_{Tj})$
       corresponding to 3D points, constructed
       from viewgraph $\mathcal{G}_{VG} = (\mathcal{V}_C, \mathcal{E}_{VG})$.
Output: Generically parallel rigid subgraphs of the
        bipartite BA graph.
// Step 1 - Identify 4-length loops
while any track ($\mathcal{T}_j$) or cam-cam edges ($\mathcal{E}_{VG}$) have
  $< 2$ observations do
    Remove tracks with $< 2$ observations.
    Remove cam-cam edges with $< 2$ observations.
end
// Step 2 - Extract subgraphs
for each cam-cam edge $(u, v) \in \mathcal{E}_{VG}$ do
    if adjacent edges $(r, z) \in \mathcal{E}_{VG}$ have at least one
      common track with $(u, v)$ then
        Create a new set $\{(u, v), (r, z)\}$.
        if $(u, v)$ or $(r, z)$ belongs to existing set(s)
          then
            Merge the existing set(s) to the new set.
        end
    end
end
For each set of cam-cam edges, extract tracks
  through matched observations to obtain subgraphs.
// Step 3 - Merge appropriate
   subgraphs
Merge two subgraphs if there are at least two
  common tracks between them.
```



(a) Original track. (b) Modified track.

Figure 4. Illustration of the occurrence of a hanging observation in a track. Black solid lines denote observations (or equivalently, cam-pt edges), while dashed lines denote cam-cam edges. When observations $P - C_3$ and $P - C_5$ are removed while modifying the track, it results in a hanging observation $P - C_4$.

Recall that every cam-cam edge provides two observations (or cam-pt edges) to a track. Whenever tracks are modified by removing some observations, a scenario can occur when an observation in a track is not supported by any cam-cam edge, which is not desirable. We call such an observation a **hanging observation** (see Fig. 4). Our method utilizes the cam-cam edges, which do not account for these observations. Hence, we remove such observations and denote this operation as **Hanging Observation Filter (HOF)**.

5. Experiments

In this section, we provide experimental results on IMC 2022 [34] and 1DSfM [80] datasets. Incremental pipelines, both in SfM and SLAM, add cam-pt edges in tracks based on the 3D points available in the intermediate reconstructions. At a given increment stage, these pipelines only consider a cam-cam edge in the viewgraph if there are sufficient matched observations in that cam-cam edge which have reconstructed 3D points. Thus, in incremental pipelines, the bipartite BA graph $\mathcal{G}_{BA}$ has a high chance of being GPR. Due to this reason, for our experiments, we use a global SfM pipeline, GLOMAP [64], to demonstrate the impact of our analysis. Our method is implemented in C++. All experiments are performed on a PC with an Intel Xeon CPU E5-2630 processor with 96 GB RAM.

GLOMAP [64] selects a subset of all constructed tracks and uses them for estimating camera parameters (including translations) and 3D point coordinates. It also filters the tracks at various stages in the pipeline to account for outliers. We apply our HOF and GPRBA after the subset selection and whenever tracks are filtered. We also present results by applying our HOF exclusively to study its impact on the reconstructions. GLOMAP also performs an incremental retriangulation (similar to [66]) with the estimated camera parameters to increase the number of reconstructed 3D points. We report results obtained at the output of BA both without and with retriangulation.

In Tabs. 1 and 2, we provide reconstruction statistics on IMC 2022 and 1DSfM datasets, respectively. GLOMAP filters the tracks and retains only those observations that have low reprojection errors. We observe that applying our HOF, whenever tracks are modified in GLOMAP [64], results in significantly more 3D points reconstructed with almost the same number of reconstructed cameras, especially without retriangulation. This reveals that removing hanging observations leads to more observations with low reprojection errors, suggesting their presence affects estimation in BA.

For IMC 2022 datasets (Tab. 1), incorporating our GPRBA along with HOF results in a slight reduction in the number of cameras and 3D points reconstructed compared to only using HOF. This is because our method removes cameras and 3D points that have the independent scaling issue. We also report camera translation errors as our analysis impacts them. We compute the errors after robustly aligning the solutions with the ground truth provided using the method in [80]. The median errors give

Dataset	# Cameras			# 3D points (in 10^3)						Camera translation errors (in m) w/ retriangulation				
				w/o retriangulation			w/ retriangulation			Median Errs.		Removed	# Cams. with errs. >5 m	
	GLO.	HOF	HOF+GPRBA	GLO.	HOF	HOF+GPRBA	GLO.	HOF	HOF+GPRBA	GLO.	HOF+GPRBA	cam. errs.	GLO.	HOF+GPRBA
	[64]	[Ours]	[Ours]	[64]	[Ours]	[Ours]	[64]	[Ours]	[Ours]	[64]	[Ours]	GLO. [64]	[64]	[Ours]
Brandenburg Gate (BDG)	349	349	346	0.5	16	15	31	29	29	**0.21**	0.26	1081.28	13	**6**
British Museum (BRM)	176	176	176	0.3	15	14	25	27	27	0.16	**0.14**	–	**0**	**0**
Buckingham Palace (BKP)	446	446	443	1.2	29	28	76	84	84	0.69	**0.17**	329.07	13	**9**
Colosseum Exterior (COE)	499	499	498	1.6	56	55	95	98	98	0.60	**0.40**	8.86	26	**24**
Lincoln Memorial Statue (LMS)	214	214	214	0.2	9	9	15	17	17	**0.04**	0.05	–	**2**	3
Notre Dame Front Facade (NDF)	911	911	900	2.8	93	90	154	157	158	**0.04**	**0.04**	77.16	**13**	14
Pantheon Exterior (PNE)	320	319	320	0.8	38	38	58	61	61	0.08	**0.05**	0.86	**2**	3
Piazza San Marco (PSM)	68	68	68	0.61	12	11	23	24	24	0.33	**0.22**	–	2	**0**
Sacre Coeur (SCR)	281	281	281	0.8	26	25	42	48	48	0.23	**0.16**	–	3	**1**
Sagrada Familia (SAF)	90	90	90	0.6	32	32	41	41	41	**0.07**	**0.07**	–	**0**	**0**
St Pauls Cathedral (SPC)	142	142	141	0.4	17	16	33	33	33	0.30	**0.27**	184.01	3	**2**
Temple Nara Japan (TNJ)	217	217	217	0.2	19	18	22	29	28	0.74	**0.28**	–	**8**	**8**
Trevi Fountain (TRF)	706	706	699	1.6	111	110	165	166	165	**0.05**	0.08	68.37	11	**7**

Table 1. Reconstruction statistics on IMC 2022 [34] before and after applying our HOF and GPRBA methods. Removed cam. errs. are the mean errors in [64] of the cameras removed after applying our HOF+GPRBA. '–' indicates no camera was removed. Applying our HOF increases reconstructed 3D points significantly, especially w/o retriangulation in GLOMAP [64]. Incorporating our HOF+GPRBA improves camera translations by removing cameras with high errors (Removed cam. errs), while reducing # cams. with errs. >5 m.

Dataset	# Cameras			# 3D points (in 10^3)					
				w/o retriangulation			w/ retriangulation		
	GLO.	HOF	HOF+GPRBA	GLO.	HOF	HOF+GPRBA	GLO.	HOF	HOF+GPRBA
	[64]	[Ours]	[Ours]	[64]	[Ours]	[Ours]	[64]	[Ours]	[Ours]
Alamo (ALM)	1117	1111	724	4.4	100	73	132	150	119
Ellis Island (ELS)	1302	1302	321	6.8	81	431	112	157	77
Gendarmenmarkt (GMM)	1122	1122	997	5.0	131	133	162	174	165
Madrid Metropolis (MDR)	718	718	443	3.3	55	44	65	73	60
Montreal Notre Dame (MND)	613	613	518	2.5	57	55	86	96	87
Notre Dame (ND)	1428	1428	1396	17.4	255	243	260	299	299
NYC Library (NYC)	1171	1171	466	14.1	118	51	141	166	82
Piazza del Popolo (PDP)	1263	1263	964	3.8	73	67	104	120	106
Piccadilly (PIC)	3476	3476	2818	8.2	214	199	266	309	271
Roman Forum (ROF)	1776	1776	1390	8.8	218	166	268	302	270
Tower of London (TOL)	926	926	675	6.1	121	114	136	153	144
Trafalgar (TFG)	8868	8863	6714	25.5	398	356	463	496	450
Union Square (USQ)	1434	1433	942	3.5	77	69	67	76	71
Vienna Cathedral (VNC)	1777	1777	1092	11.1	199	136	278	335	232
Yorkminster (YKM)	2014	2014	585	10.8	265	83	318	410	131

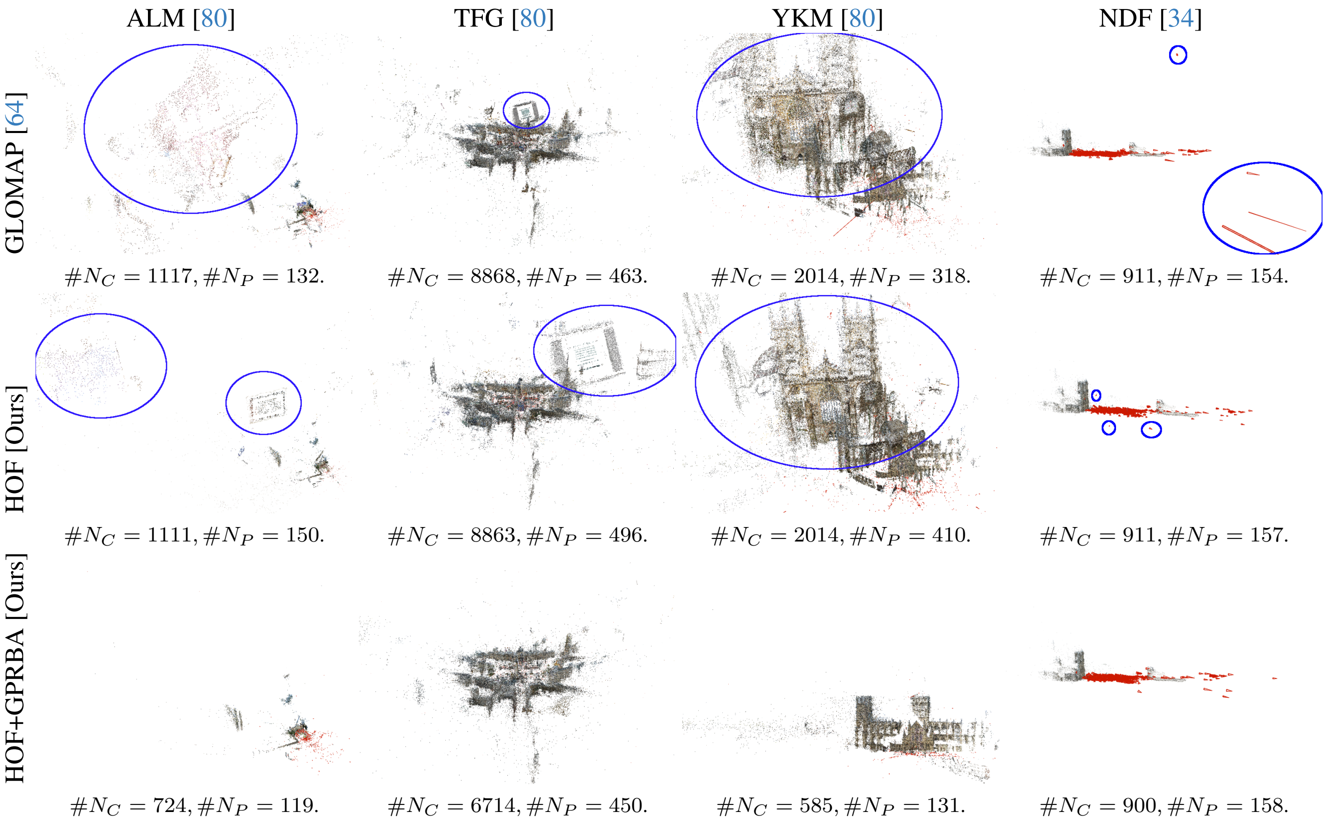
Table 2. Reconstruction statistics on 1DSfM datasets [80]. Applying our HOF increases reconstructed 3D points significantly, especially w/o retriangulation in GLOMAP [64]. Incorporating our HOF + GPRBA in [64] clearly indicates that a large number of cameras and 3D points do not form a GPR graph in BA, which cannot be merged since they can be scaled independent of the reconstruction obtained.

1DSfM	Time (mins)		IMC 2022	Time (mins)	
[80]	t_{GPR}	t_R	[34]	t_{GPR}	t_R
ALM	0.44	47.07	BDG	0.12	10.40
ELS	0.12	15.88	BRM	0.13	4.90
GMM	0.39	103.56	BKP	0.16	28.25
MDR	0.09	26.68	COE	0.78	38.55
MND	0.12	14.96	LMS	0.09	5.25
ND	1.80	328.83	NDF	1.47	112.38
NYC	0.11	14.21	PNE	0.33	14.48
PDP	0.15	31.17	PSM	0.01	2.05
PIC	1.49	177.07	SCR	0.13	15.25
ROF	0.37	58.29	SAF	0.08	8.66
TOL	0.25	20.92	SPC	0.06	5.06
TFG	4.07	651.54	TNJ	0.12	9.50
USQ	0.16	21.66	TRF	0.72	73.05
VNC	0.65	67.56			
YKM	0.27	26.22			

Table 3. Computation time. t_{GPR}: Time for five invocations of our HOF + GPRBA. t_R: Reconstruction time by GLOMAP [64] for one run. Our method takes $< 2\%$ of the reconstruction time.

an idea of the camera errors that are not misplaced, and our method improves the estimates of these cameras. To check for the errors of misplaced cameras, we report the mean error in GLOMAP [64] of the cameras removed by our method in the column "Removed cam. errs." of Tab. 1. It can be seen that the errors of the cameras removed are high, signifying the importance of our analysis. Moreover, our method reduces the number of cameras with errors >5 m. We note that high camera errors can be caused by multiple reasons, including the independent scaling issue (parallel rigidity) and the presence of outliers in the matched 2D observations. Our approach only removes the cameras with the independent scaling issue, and thus, some of the remaining cameras have high errors. For 1DSfM datasets (Tab. 2), our HOF+GPRBA removes a large number of cameras and 3D points because they do not form a GPR graph in BA. We emphasize that they cannot be merged with the reconstruction already obtained as they can be scaled independent of the obtained reconstruction resulting in misplaced cameras and 3D points. We could not analyze translation errors for 1DSfM datasets because the ground truth is not available for all cameras, and thus, the errors do not reflect the gains by removing the misplaced cameras.

In Fig. 5, we show reconstructions of various datasets with retriangulation in GLOMAP [64]. Our method extracts GPR subgraphs in BA, which removes misplaced parts of the reconstructions (marked in blue), leading to significantly cleaner reconstructions than GLOMAP alone. We reiterate that our method does not deal with outliers, and hence some reconstruction artifacts can be present. Moreover, only removing hanging observations (by HOF) does not lead to the removal of misplaced reconstruction parts, showing the significance of GPRBA. We also present reconstructions without retriangulation in the supplementary material, where we observe a similar trend as seen for



$\#N_C = 1117, \#N_P = 132.$ $\#N_C = 8868, \#N_P = 463.$ $\#N_C = 2014, \#N_P = 318.$ $\#N_C = 911, \#N_P = 154.$

$\#N_C = 1111, \#N_P = 150.$ $\#N_C = 8863, \#N_P = 496.$ $\#N_C = 2014, \#N_P = 410.$ $\#N_C = 911, \#N_P = 157.$

$\#N_C = 724, \#N_P = 119.$ $\#N_C = 6714, \#N_P = 450.$ $\#N_C = 585, \#N_P = 131.$ $\#N_C = 900, \#N_P = 158.$

Figure 5. Reconstruction results with retriangulation in GLOMAP [64]. $\#N_C$ and $\#N_P$: number of reconstructed cameras and 3D points (in 10^3). Applying only our HOF does not remove misplaced cameras and 3D points (blue ellipses). Our approach HOF+GPRBA, that extracts GPR subgraphs in BA, results in significantly cleaner reconstructions by removing misplaced cameras and 3D points.

the reconstructions with retriangulation. Additionally, we see that by incorporating HOF + GPRBA in GLOMAP, we can obtain reasonable reconstructions even without retriangulation, which is not the case when using GLOMAP alone.

GLOMAP filters the tracks five times in a single run. HOF+GPRBA is applied whenever the tracks are modified and complements the pipeline. This is because if many observations are removed while filtering tracks, $\mathcal{G}_{BA}$ can have many strongly connected clusters with few cam-pt edges across them, which may indicate the presence of incorrect (false) edges. If these clusters do not have enough cam-pt edges across them, they get separated by extracting a GPR subgraph using our approach. This case occurs in ELS, NYC, VNC and YKM, where incorporating our approach results in significantly cleaner reconstructions by separating unrelated parts of the scene in $\mathcal{G}_{BA}$. The reconstructions obtained with HOF+GPRBA are similar as shown in the dataset paper [80] and recent papers reporting COLMAP outputs [50, 52, 90]. This shows the usefulness of GPR analysis on $\mathcal{G}_{BA}$ to separate unrelated scene parts.

In Tab. 3, we provide the computation time for our approach. We see that five invocations of HOF+GPRBA in a single run of GLOMAP takes $< 2\%$ of the reconstruction time for all datasets, which further improves to $< 1\%$ for datasets with > 1000 cameras. This shows the efficiency and scalability of our approach.

6. Conclusion

This paper examines the issue of unique solvability in BA through parallel rigidity, which arises due to jointly solving camera translations and 3D points. We analyze the topology of the bipartite BA graph to avoid independent scaling of its edges. Our method inspects 4-length loops in the bipartite BA graph to extract its subgraphs that are generically parallel rigid. For efficient extraction, we utilize the camera-camera relationships in the viewgraph from which the bipartite BA graph is constructed. Incorporating our approach into GLOMAP [64] results in significantly cleaner reconstructions by removing misplaced cameras and 3D points. Moreover, we remove observations in the bipartite BA graph that are not matched in the viewgraph, which leads to significantly more reconstructed 3D points.

References

[1] Sameer Agarwal, Noah Snavely, Steven Seitz, and Richard Szeliski. Bundle adjustment in the large. In *European Conference on Computer Vision*, pages 29–42. Springer, 2010. 1

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54 (10):105–112, 2011. 1, 2

[3] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. `http://ceres-solver.org`, 2012. 1

[4] Hatem Alismail, Brett Browning, and Simon Lucey. Photometric bundle adjustment for vision-based SLAM. In *Asian Conference on Computer Vision*, pages 324–341. Springer, 2017. 1, 2

[5] Federica Arrigoni and Andrea Fusiello. Bearing-based network localizability: A unifying view. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2049–2069, 2018. 1, 2, 3, 4

[6] Federica Arrigoni, Andrea Fusiello, and Beatrice Rossi. On computing the translations norm in the epipolar graph. In *International Conference on 3D Vision*, pages 300–308. IEEE, 2015. 2, 3

[7] Federica Arrigoni, Andrea Fusiello, and Beatrice Rossi. Camera motion from group synchronization. In *International Conference on 3D Vision*, pages 546–555. IEEE, 2016. 2

[8] Federica Arrigoni, Andrea Fusiello, Elisa Ricci, and Tomas Pajdla. Viewing graph solvability via cycle consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5540–5549, 2021. 3

[9] Federica Arrigoni, Tomas Pajdla, and Andrea Fusiello. Viewing graph solvability in practice. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8147–8155, 2023.

[10] Federica Arrigoni, Andrea Fusiello, and Tomas Pajdla. A direct approach to viewing graph solvability. In *European Conference on Computer Vision*, pages 319–334. Springer, 2024. 3

[11] Alex R Berg and Tibor Jordán. Algorithms for graph rigidity and scene analysis. In *Algorithms-ESA 2003: 11th Annual European Symposium*, pages 78–89. Springer, 2003. 3

[12] Niclas Börlin and Pierre Grussenmeyer. Bundle adjustment with and without damping. *The Photogrammetric Record*, 28(144):396–415, 2013. 2

[13] Martin Byröd and Karl Åström. Bundle adjustment using conjugate gradients with multiscale preconditioning. In *British Machine Vision Conference*, 2009. 1

[14] Martin Byröd and Kalle Åström. Conjugate gradient bundle adjustment. In *European Conference on Computer Vision*, pages 114–127. Springer, 2010. 2

[15] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM. *IEEE Transactions on Robotics*, 37(6): 1874–1890, 2021. 1

[16] Avishek Chatterjee and Venu Madhav Govindu. Efficient and robust large-scale rotation averaging. In *International Conference on Computer Vision*, pages 521–528. IEEE, 2013. 2

[17] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):958–972, 2017. 2

[18] Yanqing Chen, Timothy A Davis, William W Hager, and Sivasankaran Rajamanickam. Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software (TOMS)*, 35(3):1–14, 2008. 2

[19] Siddharth Choudhary and PJ Narayanan. Visibility probability structure from SfM datasets and applications. In *European Conference on Computer Vision*, pages 130–143. Springer, 2012. 1

[20] Frank Dellaert. Factor graphs and GTSAM: A hands-on introduction. 2012. 1

[21] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision*, pages 834–849. Springer, 2014. 1

[22] Tolga Eren, Walter Whiteley, A Stephen Morse, Peter N Belhumeur, and Brian D O Anderson. Sensor and network topologies of formations with direction, bearing, and angle information between agents. In *IEEE International Conference on Decision and Control*, pages 3064–3069. IEEE, 2003. 1, 3

[23] Tolga Eren, Walter Whiteley, and Peter N Belhumeur. A theoretical analysis of the conditions for unambiguous node localization in sensor networks. Technical report, Department of Computer Science, Columbia University, Tech. Rep, 2004. 3

[24] Tolga Eren, Walter Whiteley, and Peter N Belhumeur. Using angle of arrival (bearing) information in network localization. In *IEEE Conference on Decision and Control*, pages 4676–4681. IEEE, 2006. 1, 3

[25] Anders Eriksson, John Bastian, Tat-Jun Chin, and Mats Isaksson. A consensus-based framework for distributed bundle adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1754–1762, 2016. 1

[26] Alexandre Eudes and Maxime Lhuillier. Error propagations for local bundle adjustment. In *2009 IEEE conference on computer vision and pattern recognition*, pages 2411–2418. IEEE, 2009. 1

[27] Maxime Ferrera, Alexandre Eudes, Julien Moras, Martial Sanfourche, and Guy Le Besnerais. OV^2 SLAM: A fully online and versatile visual SLAM for real-time applications. *IEEE Robotics and Automation Letters*, 6(2):1399–1406, 2021. 1

[28] Zvi Galil and Giuseppe F Italiano. Data structures and algorithms for disjoint set union problems. *ACM Computing Surveys (CSUR)*, 23(3):319–344, 1991. 5

[29] Thomas Goldstein, Paul Hand, Choongbum Lee, Vladislav Voroninski, and Stefano Soatto. Shapefit and shapekick for robust, scalable structure-from-motion. In *European Conference on Computer Vision*, pages 289–304. Springer, 2016. 2

[30] Venu Madhav Govindu. Combining two-view constraints for motion estimation. In *Conference on Computer Vision and Pattern Recognition*, pages 218–225. IEEE, 2001.

[31] Venu Madhav Govindu. Lie-algebraic averaging for globally consistent motion estimation. In *Conference on Computer Vision and Pattern Recognition*. IEEE, 2004. 2

[32] Johan Hedborg, Per-Erik Forssén, Michael Felsberg, and Erik Ringaby. Rolling shutter bundle adjustment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1434–1441. IEEE, 2012. 1, 2

[33] Magnus R Hestenes, Eduard Stiefel, et al. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409–436, 1952. 2

[34] Addison Howard, Eduard Trulls, etru1927, Kwang Moo Yi, old ufo, Sohier Dane, and Yuhe Jin. Image matching challenge 2022. https://kaggle.com/competitions/image-matching-challenge-2022, 2022. Kaggle. 6, 7, 8

[35] Yong-Dian Jian, Doru C Balcan, and Frank Dellaert. Generalized subgraph preconditioners for large-scale bundle adjustment. In *Outdoor and Large-Scale Real-World Scene Analysis: 15th International Workshop on Theoretical Foundations of Computer Vision, Revised Selected Papers*, pages 131–150. Springer, 2012. 2

[36] Arman Karimian and Roberto Tron. Theory and methods for bearing rigidity recovery. In *IEEE Conference on Decision and Control*, pages 2228–2235. IEEE, 2017. 3

[37] Bastian Katz, Marco Gaertler, and Dorothea Wagner. Maximum rigid components as means for direction-based localization in sensor networks. *Lecture Notes in Computer Science*, 4362:330, 2007. 3, 4

[38] Ryan Kennedy, Kostas Daniilidis, Oleg Naroditsky, and Camillo J Taylor. Identifying maximal rigid components in bearing-based localization. In *International Conference on Intelligent Robots and Systems*, pages 194–201. IEEE, 2012. 2, 3

[39] Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 225–234. IEEE, 2007. 1

[40] Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g^2o: A general framework for graph optimization. In *2011 IEEE international conference on robotics and automation*, pages 3607–3613. IEEE, 2011. 1

[41] Christian Kurz, Thorsten Thormählen, and Hans-Peter Seidel. Bundle adjustment for stereoscopic 3D. In *Computer Vision/Computer Graphics Collaboration Techniques*, pages 1–12. Springer, 2011. 1, 2

[42] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944. 2

[43] Maxime Lhuillier. Incremental fusion of structure-from-motion and gps using constrained bundle adjustments. *IEEE transactions on pattern analysis and machine intelligence*, 34(12):2489–2495, 2012. 1

[44] Zheng Liu and Fu Zhang. BALM: Bundle adjustment for lidar mapping. *IEEE Robotics and Automation Letters*, 6(2): 3184–3191, 2021. 1, 2

[45] Manolis Lourakis and Antonis Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the Levenberg-Marquardt algorithm. Technical Report 340, Institute of Computer Science-FORTH, Heraklion, Crete, Greece, 2004. 1, 2

[46] Manolis Lourakis and Antonis Argyros. Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment? In *International Conference on Computer Vision*, pages 1526–1531. IEEE, 2005. 1, 2

[47] Manolis Lourakis and Antonis Argyros. SBA: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1):1–30, 2009. 1

[48] Lalit Manam and Venu Madhav Govindu. Correspondence reweighted translation averaging. In *European Conference on Computer Vision*, pages 56–72. Springer, 2022. 2

[49] Lalit Manam and Venu Madhav Govindu. Sensitivity in translation averaging. In *Advances in Neural Information Processing Systems*, 2023. 2

[50] Lalit Manam and Venu Madhav Govindu. Leveraging camera triplets for efficient and accurate structure-from-motion. In *Conference on Computer Vision and Pattern Recognition*, pages 4959–4968. IEEE, 2024. 8

[51] Lalit Manam and Venu Madhav Govindu. Fusing directions and displacements in translation averaging. In *International Conference on 3D Vision*, pages 75–84. IEEE, 2024. 2

[52] Lalit Manam and Venu Madhav Govindu. Unifying view-graph sparsification and disambiguation of repeated structures in structure-from-motion. *International Journal of Computer Vision*, 134(4):171, 2026. 8

[53] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. 2

[54] Ellips Masehian and MR Amin-Naseri. A voronoi diagram-visibility graph-potential field compound algorithm for robot path planning. *Journal of Robotic Systems*, 21(6):275–300, 2004. 1

[55] Kaushik Mitra and Rama Chellappa. A scalable projective bundle adjustment algorithm using the L-∞ norm. In *Indian Conference on Computer Vision, Graphics & Image Processing*, pages 79–86. IEEE, 2008. 1, 2

[56] Daniel D Morris, Kenichi Kanatani, and Takeo Kanade. Uncertainty modeling for optimal structure-from-motion. In *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings*, pages 200–217. Springer, 2000. 1

[57] C Moukarzel. An efficient algorithm for testing the generic rigidity of graphs in the plane. *Journal of Physics A: Mathematical and General*, 29(24):8079–8098, 1996. 3

[58] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. OpenMVG: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 60–74. Springer, 2016. 1, 2

[59] Raul Mur-Artal and Juan D Tardós. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. 1, 2

[60] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. 1

[61] Karthikeyan Natesan Ramamurthy, Chung-Ching Lin, Aleksandr Aravkin, Sharath Pankanti, and Raphael Viguier. Distributed bundle adjustment. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2146–2154, 2017. 1

[62] Onur Ozyesil and Amit Singer. Robust camera location estimation by convex programming. In *Conference on Computer Vision and Pattern Recognition*, pages 2674–2683. IEEE, 2015. 2

[63] Onur Ozyesil, Amit Singer, and Ronen Basri. Stable camera motion estimation using convex programming. *SIAM Journal on Imaging Sciences*, 8(2):1220–1262, 2015. 2

[64] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision*, pages 58–77. Springer, 2024. 1, 2, 4, 6, 7, 8

[65] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003. 2

[66] Johannes Lutz Schonberger and Jan-Michael Frahm. structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition*, pages 4104–4113. IEEE, 2016. 1, 6

[67] Thomas Schops, Torsten Sattler, and Marc Pollefeys. BAD SLAM: Bundle adjusted direct RGB-D SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 134–144, 2019. 1, 2

[68] J Schur. Über potenzreihen, die im innern des einheitskreises beschränkt sind. 1917. 2

[69] Brigitte Servatius and Walter Whiteley. Constraining plane configurations in computer-aided design: Combinatorics of directions and lengths. *SIAM Journal on Discrete Mathematics*, 12(1):136–153, 1999. 3

[70] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3D. In *SIGGRAPH 2006 Papers*, pages 835–846. ACM, 2006. 1

[71] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80:189–210, 2008.

[72] Christopher Sweeney, Tobias Hollerer, and Matthew Turk. Theia: A fast and scalable structure-from-motion library. In *International Conference on Multimedia*, pages 693–696. ACM, 2015. 1

[73] Matthew Trager, Brian Osserman, and Jean Ponce. On the solvability of viewing graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 321–335, 2018. 3

[74] Bill Triggs, Philip F McLauchlan, Richard Hartley, and Andrew W Fitzgibbon. Bundle adjustment: A modern synthesis. In *International Workshop on Vision Algorithms*, pages 298–372. Springer, 1999. 1, 2

[75] Roberto Tron, Luca Carlone, Frank Dellaert, and Kostas Daniilidis. Rigid components identification and rigidity control in bearing-only localization using the graph cycle basis. In *American Control Conference*, pages 3911–3918. IEEE, 2015. 3

[76] Walter Whiteley. Parallel redrawing of configurations in 3-space. *preprint*, 1986. 1, 3

[77] Walter Whiteley. A matroid on hypergraphs, with applications in scene analysis and geometry. *Discrete & Computational Geometry*, 4(1):75–95, 1989. 1, 3

[78] Walter Whiteley. Some matroids from discrete applied geometry. *Contemporary Mathematics*, 197:171–312, 1996. 3

[79] Kyle Wilson and Noah Snavely. Network principles for SfM: Disambiguating repeated structures with local context. In *International Conference on Computer Vision*, pages 513–520. IEEE, 2013. 1

[80] Kyle Wilson and Noah Snavely. Robust global translations with 1DSfM. In *European Conference on Computer Vision*, pages 61–75. Springer, 2014. 2, 6, 7, 8

[81] Kyle Wilson and Scott Wehrwein. Visualizing spectral bundle adjustment uncertainty. In *International Conference on 3D Vision*, pages 663–671. IEEE, 2020. 1

[82] SJ Wright and John Norman Holt. An inexact Levenberg-Marquardt method for large sparse nonlinear least squres. *The ANZIAM Journal*, 26(4):387–403, 1985. 2

[83] Changchang Wu. VisualSfM: A visual structure-from-motion system. `http://www.cs.washington.edu/homes/ccwu/vsfm`, 2011. 1

[84] Changchang Wu, Sameer Agarwal, Brian Curless, and Steven M Seitz. Multicore bundle adjustment. In *Conference on Computer Vision and Pattern Recognition*, pages 3057–3064. IEEE, 2011. 1

[85] Qingan Yan, Long Yang, Ling Zhang, and Chunxia Xiao. Distinguishing the indistinguishable: Exploring structural ambiguities via geodesic context. In *Conference on Computer Vision and Pattern Recognition*, pages 3836–3844. IEEE, 2017. 1

[86] Christopher Zach. Robust bundle adjustment revisited. In *European Conference on Computer Vision*, pages 772–787. Springer, 2014. 1, 2

[87] Christopher Zach and Guillaume Bourmaud. Pareto meets Huber: Efficiently avoiding poor minima in robust estimation. In *International Conference on Computer Vision*, pages 10243–10251. IEEE, 2019. 1, 2

[88] Shiyu Zhao and Daniel Zelazo. Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions. *Automatica*, 69:334–341, 2016. 3

[89] Shiyu Zhao, Zhiyong Sun, Daniel Zelazo, Minh-Hoang Trinh, and Hyo-Sung Ahn. Laman graphs are generically bearing rigid in arbitrary dimensions. In *56th Annual Conference on Decision and Control (CDC)*, pages 3356–3361. IEEE, 2017. 3

[90] Pengwei Zhou, Hongche Yin, Guozheng Xu, Xiaosong Wei, Annan Zhou, Jian Yao, Li Li, and Huang Jing. Pgvs: A probabilistic graph-theoretic framework for view-graph selection in structure-from-motion. *ISPRS Journal of Photogrammetry and Remote Sensing*, 231:641–663, 2026. 8

[91] Bingbing Zhuang, Loong-Fah Cheong, and Gim Hee Lee. Baseline desensitizing in translation averaging. In *Conference on Computer Vision and Pattern Recognition*, pages 4539–4547. IEEE, 2018. 2, 4

Parallel Rigidity Matters for Bundle Adjustment
Supplementary Material

Lalit Manam
Mitsubishi Electric Research Labs (MERL)
Cambridge, MA 02139, USA
manam@merl.com

Venu Madhav Govindu
Indian Institute of Science
Bengaluru, KA 560012, INDIA
venug@iisc.ac.in

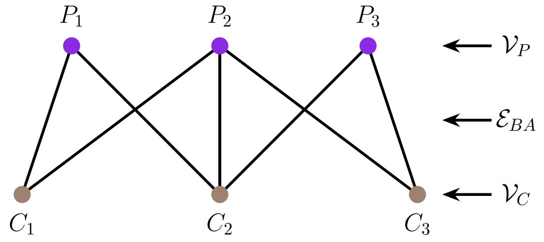
A. Graph Notations

In this section, we provide the graph notations used in Sec. 4 of the main paper along with a visual representation in Fig. S1 for clarity. The graphs involved in the paper:

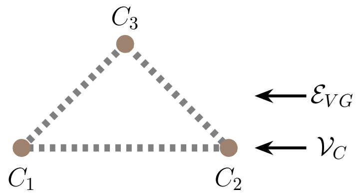
- $\mathcal{G}_{BA} = (\{\mathcal{V}_C \cup \mathcal{V}_P\}, \mathcal{E}_{BA})$: bipartite BA graph (Fig. S1a),
- $\mathcal{G}_{VG} = (\mathcal{V}_C, \mathcal{E}_{VG})$: viewgraph (Fig. S1b),

where

- $\mathcal{V}_C$: set of cameras (brown nodes in Figs. S1a and S1b),
- $\mathcal{V}_P$: set of 3D points (violet nodes in Fig. S1b),
- $\mathcal{E}_{BA}$: cam-pt edges in $\mathcal{G}_{BA}$ representing 2D observations such that $\mathcal{E}_{BA} \subseteq \mathcal{V}_C \times \mathcal{V}_P$ (black lines in Fig. S1a),
- $\mathcal{E}_{VG}$: cam-cam edges in $\mathcal{G}_{VG}$ representing matched 2D observations such that $\mathcal{E}_{VG} \subseteq \mathcal{V}_C \times \mathcal{V}_C$ (thick dashed gray lines in Fig. S1b).



(a) Bipartite BA graph ($\mathcal{G}_{BA}$).



(b) Viewgraph ($\mathcal{G}_{VG}$).

Figure S1. Notations of the two graphs used in the paper. Bipartite BA graph: $\mathcal{G}_{BA} = (\{\mathcal{V}_C \cup \mathcal{V}_P\}, \mathcal{E}_{BA})$. Viewgraph: $\mathcal{G}_{VG} = (\mathcal{V}_C, \mathcal{E}_{VG})$. See text for details.

B. Additional Details on the Proposed Method

In Sec. 4 of the main paper, we presented our proposed method. Here, we provide details on maintaining subgraphs with union-find data structure and on the computation complexity of GPRBA.

Subgraphs with union-find data structure: In Step 2 of our method, we noted that matched observations in each cam-cam edge in the viewgraph $\mathcal{G}_{VG}$ form subgraphs of the bipartite BA graph $\mathcal{G}_{BA}$ that are generically parallel rigid (GPR), through Type-2 interactions of 4-length loops in $\mathcal{G}_{BA}$. Then, we considered Type-1 interactions of 4-length loops in $\mathcal{G}_{BA}$ via common tracks between two cam-cam edges in $\mathcal{G}_{VG}$. To efficiently record which cam-cam edges in $\mathcal{G}_{VG}$ together form a GPR subgraph in $\mathcal{G}_{BA}$, we utilize the union-find data structure on cam-cam edges.

The union-find data structure stores a collection of disjoint sets to efficiently query if two elements belong to the same set. Every disjoint set is identified by a representative element. In our method, each disjoint set represents a set of cam-cam edges in $\mathcal{G}_{VG}$ that together form a GPR subgraph of $\mathcal{G}_{BA}$. The data structure can be represented as a set of tuples $\mathcal{UF}$. In each tuple $(e, r) \in \mathcal{UF}$, e is an element denoting a cam-cam edge in $\mathcal{E}_{VG}$ and r denotes the representative element of the disjoint set in which e belongs to.

Before starting Step 2 of our method, we initialize $\mathcal{UF}$ by considering every element in $\mathcal{UF}$ as a disjoint set, i.e. for every $(e, r) \in \mathcal{UF}$, $r \leftarrow e$. Let $(e_1, r_1), (e_2, r_2) \in \mathcal{UF}$. While executing Step 2 of our method, when two cam-cam edges in $\mathcal{G}_{VG}$, denoted by elements e_1 and e_2, are identified to belong to the same set, then $r_2 \leftarrow r_1$. This means the representative elements of e_1 and e_2 are assigned the same, and hence, they belong to the same disjoint set in $\mathcal{UF}$. After Step 2 of our method is completed, we extract the cam-pt edges in $\mathcal{G}_{BA}$ that correspond to the cam-cam edges in $\mathcal{G}_{VG}$ having the same representative element in $\mathcal{UF}$. Each of these disjoint sets of cam-pt edges forms a GPR subgraph in $\mathcal{G}_{BA}$, which are used in Step 3 of our method.

Computational complexity of GPRBA: Let $\mathcal{N}$ be the subgraphs obtained at the end of Step 2. Our method has an average case computational complexity of $O(z|\mathcal{E}_{VG}| + z|\mathcal{V}_P| + |\mathcal{N}|^2)$, with $O(z|\mathcal{E}_{VG}| + z|\mathcal{V}_P|)$ for Step 1, $O(|\mathcal{E}_{VG}|)$ for Step 2 and $O(|\mathcal{N}|^2)$ for Step 3. The value of z depends on the number of cam-cam edges and tracks having less than two observations. In practice, for a dataset, $z \ll |\mathcal{E}_{VG}|$ and $|\mathcal{N}|^2 \ll |\mathcal{E}_{VG}| \ll |\mathcal{V}_P|$.

C. Additional Results

In Sec. 5 of the main paper, we provided camera translation errors with retriangulation on IMC 2022 [34] and visual reconstruction results with our HOF and GPRBA using GLOMAP [64]. Here, we present the translation errors both without and with retriangulation on IMC 2022 [34] and visual results on other datasets. We use the following notations in this supplementary material, same as that of the main paper:

- $\#N_C$: Number of cameras reconstructed.
- $\#N_P$: Number of 3D points reconstructed (in 10^3).

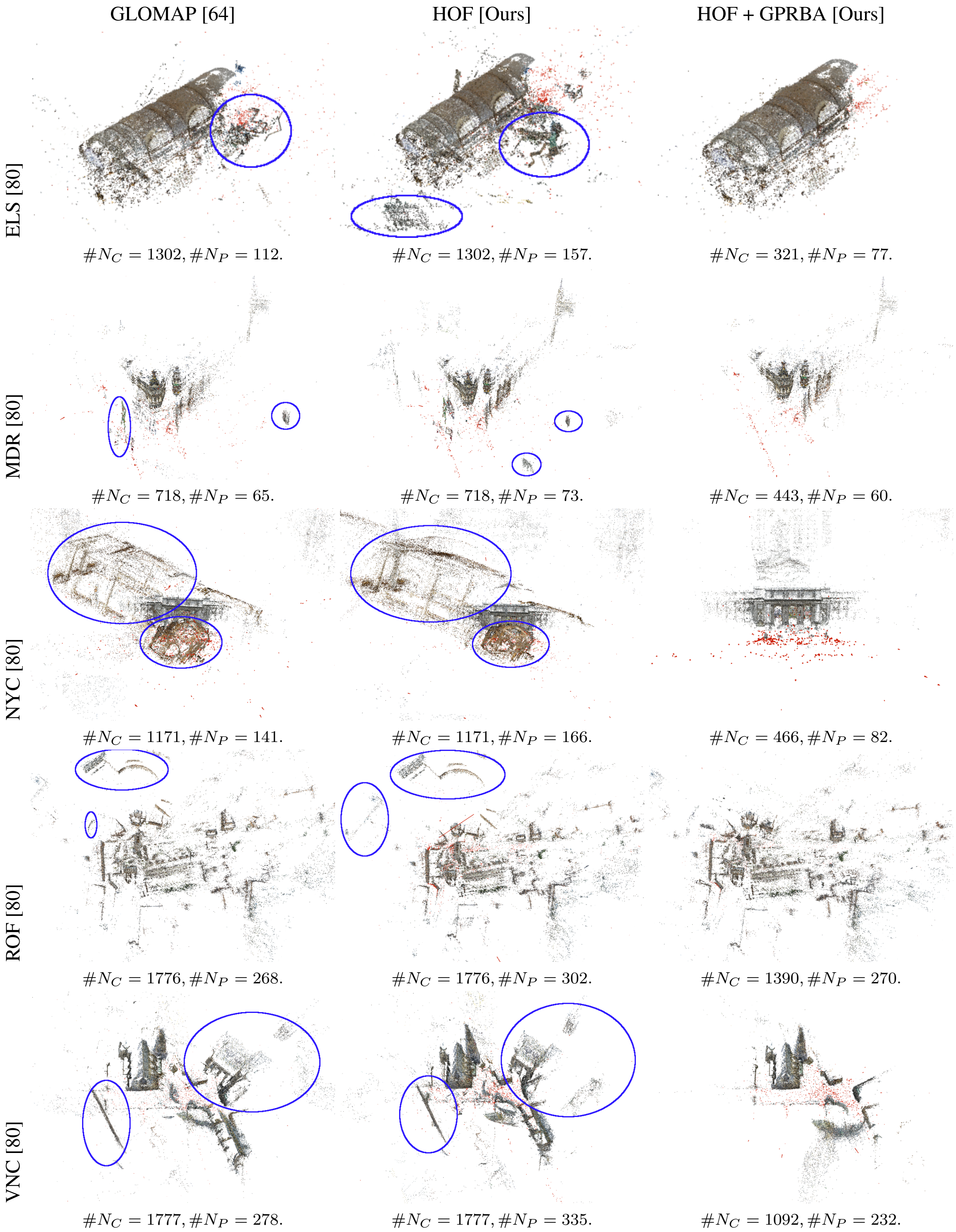
In Tab. S1, we provide reconstruction statistics both without and with retriangulation in GLOMAP [64]. We also provide camera translation errors after applying HOF and GPRBA individually along with HOF+GPRBA for completeness. However, we note that applying GPRBA individually without HOF does not guarantee a GPR graph, and is strongly discouraged. This is because hanging observations are not supported by any cam-cam edge in $\mathcal{G}_{VG}$, as discussed in practical considerations in Sec. 4. Thus, they are not accounted in GPRBA, which relies on cam-cam edges.

We observe that using GPRBA individually results in similar number of cams and more 3D points without retriangulation but similar number of 3D points with retriangulation on most datasets, compared to only HOF and HOF+GPRBA (Tab. 1 of the main paper). Overall, the median translation errors with retriangulation are less with HOF+GPRBA.

The retriangulation in GLOMAP is done in an incremental manner, which initializes with the output of without retriangulation phase. It also uses the matched observations in cam-cam edges that were not used in without retriangulation phase, due to which the impact of a GPR bipartite BA graph is difficult to assess exclusively from the results with retriangulation. So, we also provide the translation errors without retriangulation in Tab. S1. It can be seen that, without retriangulation, each of HOF and GPRBA improves over GLOMAP, while HOF+GPRBA performs overall best. This clearly shows that a GPR graph improves camera translations.

In Figs. S2 and S3, we show reconstructions with retriangulation in GLOMAP [64]. It can be seen that misplaced parts in the GLOMAP reconstructions are not removed by applying our HOF. Applying our HOF + GPRBA results in significantly clean reconstructions.

We also provide reconstruction results without retriangulation in Figs. S4 and S5. It can be seen that applying our HOF results in more reconstructed 3D points but has misplaced reconstruction parts. Similar to results with retriangulation, applying our HOF + GPRBA gives significantly clean reconstructions. We note that due to a small number of reconstructed 3D points by GLOMAP without retriangulation, misplaced parts are not clearly visible. As noted in the main paper, GLOMAP only retains observations with a low reprojection error, which removes many reconstructed 3D points. However, results with retriangulation clearly indicate that cameras belonging to misplaced parts are present in the reconstructions obtained without retriangulation, which can only be removed by applying our HOF + GPRBA.



ELS [80]: $\#N_C = 1302, \#N_P = 112.$ $\#N_C = 1302, \#N_P = 157.$ $\#N_C = 321, \#N_P = 77.$

MDR [80]: $\#N_C = 718, \#N_P = 65.$ $\#N_C = 718, \#N_P = 73.$ $\#N_C = 443, \#N_P = 60.$

NYC [80]: $\#N_C = 1171, \#N_P = 141.$ $\#N_C = 1171, \#N_P = 166.$ $\#N_C = 466, \#N_P = 82.$

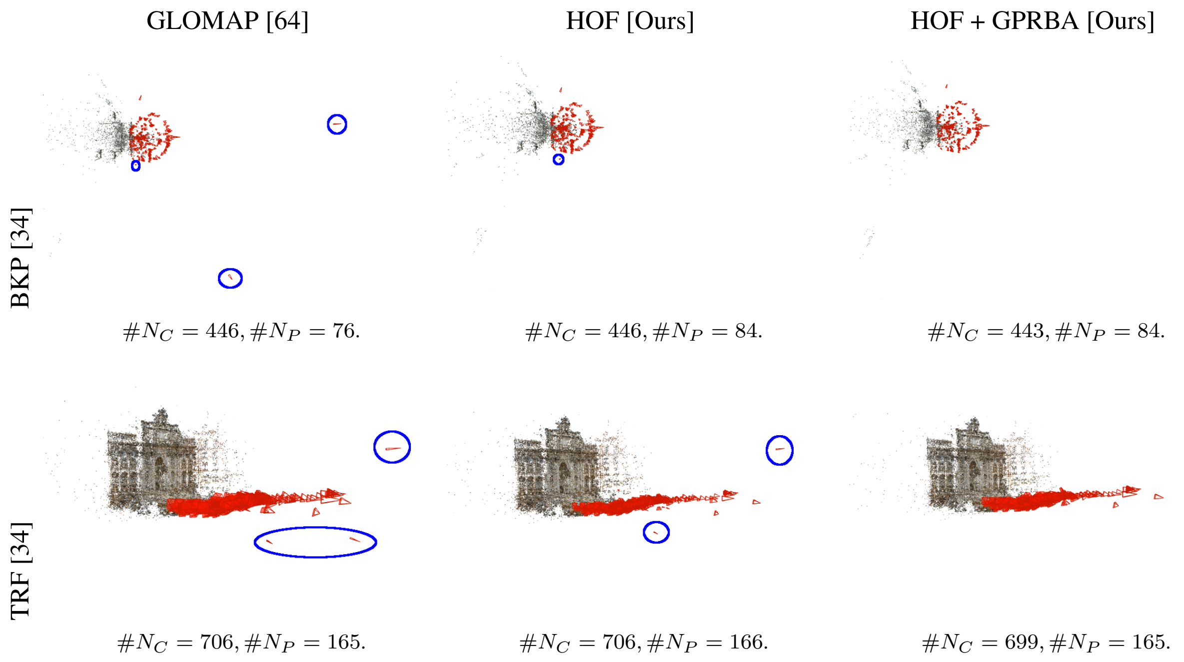
ROF [80]: $\#N_C = 1776, \#N_P = 268.$ $\#N_C = 1776, \#N_P = 302.$ $\#N_C = 1390, \#N_P = 270.$

VNC [80]: $\#N_C = 1777, \#N_P = 278.$ $\#N_C = 1777, \#N_P = 335.$ $\#N_C = 1092, \#N_P = 232.$

Figure S2. Reconstruction results with retriangulation in GLOMAP [64]. Applying our HOF does not remove misplaced parts (in blue), which are present in GLOMAP reconstructions. Our approach, HOF+GPRBA, designed to extract GPR subgraphs in BA, results in significantly clean reconstructions by removing misplaced cameras and 3D points.

Dataset	GPRBA (individual)			Median Translation Errors							
	# Cameras	# 3D Points (in 10^3)		w/o retriangulation (w/o retri.)				w/ retriangulation (w/ retri.)			
		w/o retri.	w/ retri.	GLO.	HOF	GPRBA	HOF+GPRBA	GLO.	HOF	GPRBA	HOF+GPRBA
BDG	347	18	30	1.04	0.61	**0.42**	0.52	**0.21**	0.26	0.28	0.26
BRM	176	16	26	0.72	0.35	0.38	**0.34**	0.16	0.15	0.15	**0.14**
BKP	445	36	81	6.18	0.28	2.39	**0.25**	0.69	0.21	0.33	**0.17**
COE	498	61	98	3.44	0.86	0.85	**0.81**	0.60	0.41	0.42	**0.40**
LMS	214	9	17	0.59	**0.05**	0.18	0.16	0.04	**0.03**	0.19	0.05
NDF	901	101	158	0.61	**0.14**	**0.14**	0.18	**0.04**	0.11	0.09	**0.04**
PNE	320	41	61	0.44	**0.13**	0.19	**0.13**	0.08	0.06	0.07	**0.05**
PSM	68	14	23	0.85	0.90	0.83	**0.30**	0.33	**0.20**	0.22	0.22
SCR	281	29	44	5.09	**0.12**	1.25	**0.12**	0.23	0.12	**0.11**	0.16
SAF	90	33	41	0.16	**0.06**	0.07	**0.06**	**0.07**	**0.07**	**0.07**	**0.07**
SPC	142	19	33	0.40	**0.24**	0.30	0.28	0.30	**0.16**	0.29	0.27
TNJ	217	20	28	2.26	0.83	**0.82**	0.84	0.74	0.29	0.29	**0.28**
TRF	700	117	165	0.06	0.06	**0.05**	**0.05**	0.05	**0.04**	0.09	0.08

Table S1. Reconstruction statistics on IMC 2022 [34] before and after applying our HOF and GPRBA methods. **Bold** and underlined camera translation errors are the best and second best camera translation errors in their respective category. A GPR bipartite BA graph with HOF+GPRBA results in overall best camera translations.



$\#N_C = 446, \#N_P = 76.$ $\#N_C = 446, \#N_P = 84.$ $\#N_C = 443, \#N_P = 84.$

$\#N_C = 706, \#N_P = 165.$ $\#N_C = 706, \#N_P = 166.$ $\#N_C = 699, \#N_P = 165.$

Figure S3. Reconstruction results with retriangulation in GLOMAP [64]. Similar observations can be made as in Fig. S2.

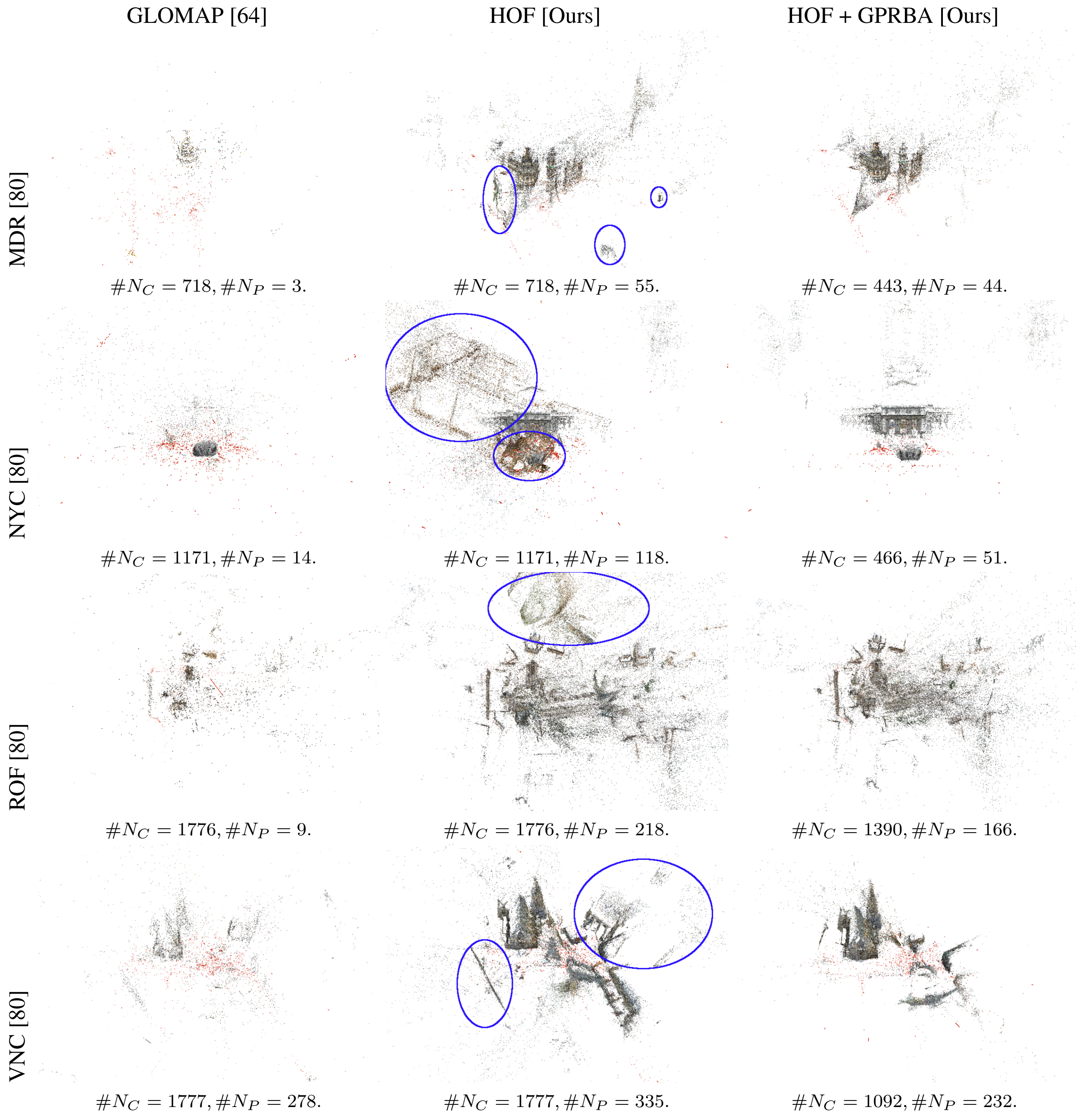
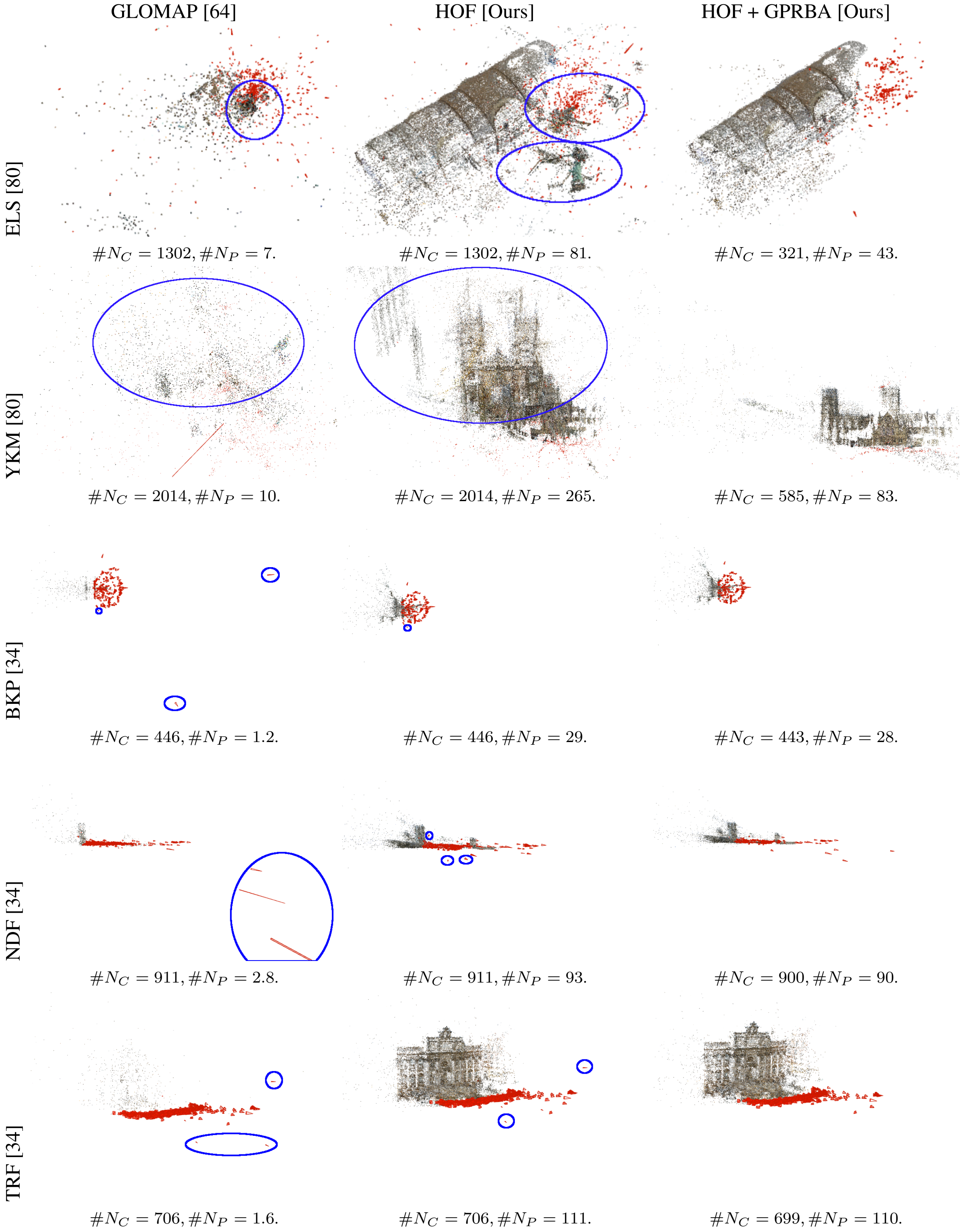


Figure S4. Reconstruction results without retriangulation in GLOMAP [64]. Applying our HOF increases reconstructed 3D points significantly but does not remove misplaced parts (blue ellipses) present in the reconstructions. Applying our HOF+GPRBA leads to significantly cleaner reconstructions by removing misplaced cameras and 3D points.



$\#N_C = 1302, \#N_P = 7.$ $\#N_C = 1302, \#N_P = 81.$ $\#N_C = 321, \#N_P = 43.$

$\#N_C = 2014, \#N_P = 10.$ $\#N_C = 2014, \#N_P = 265.$ $\#N_C = 585, \#N_P = 83.$

$\#N_C = 446, \#N_P = 1.2.$ $\#N_C = 446, \#N_P = 29.$ $\#N_C = 443, \#N_P = 28.$

$\#N_C = 911, \#N_P = 2.8.$ $\#N_C = 911, \#N_P = 93.$ $\#N_C = 900, \#N_P = 90.$

$\#N_C = 706, \#N_P = 1.6.$ $\#N_C = 706, \#N_P = 111.$ $\#N_C = 699, \#N_P = 110.$

Figure S5. Additional reconstruction results without retriangulation in GLOMAP [64]. Similar observations can be made as in Fig. S4.